

Université de Montréal

Évaluation de la compétence lexicale des modèles de langues

Par
Shuxu Li

Département de linguistique et de traduction, Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de maîtrise ès arts (M.A.) en linguistique

Août 2025

© Shuxu Li, 2025

Université de Montréal
Département de linguistique et de traduction, Faculté des arts et des sciences

Ce mémoire intitulé
Évaluation de la compétence lexicale des modèles de langues

Présenté par
Shuxu Li

A été évalué par un jury composé des personnes suivantes

Ayla Rigouts Terryn
Présidente-rapporteuse

François Lareau
Directeur de recherche

Antoine Venant
Codirecteur de recherche

Patrick Drouin
Membre du jury

Résumé

Les grands modèles de langues (LLM) manifestent une aisance remarquable dans l'usage du langage, mais l'ampleur et la profondeur de leurs connaissances linguistiques ne sont pas encore clairement délimitées. Dans ce mémoire, nous proposons un nouveau banc d'essai pour évaluer la compétence lexicale des LLM, en nous appuyant sur les fonctions lexicales (FL), un système dans le cadre de la théorie Sens-Texte (TST) qui modélise formellement la structure du lexique à travers des relations lexicales paradigmatiques et syntagmatiques.

Nous établissons une classification hiérarchique en regroupant des FL spécifiques en catégories plus générales et plus abstraites, afin d'examiner la capacité des modèles à distinguer des relations de spécificité variable. Dans le cadre expérimental, nous sélectionnons 82 FL (ou classes de FL) issues de cette hiérarchie, pour chacune desquelles nous construisons une tâche contrastive composée de 20 requêtes adressées aux LLM. Chaque requête, construite à partir des données des FL extraites du *Réseau lexical du français* (RL-fr), comprend : (i) une description textuelle de la FL ciblée, (ii) quelques exemples positifs (issus de la FL ciblée) et négatifs (issus de FL contrastives voisines), et (iii) une question polaire (*Oui/Non*) portant sur un nouvel exemple. Le modèle doit déterminer si ce nouvel exemple instancie ou non la FL ciblée en se fondant sur les deux premiers éléments de la requête.

Les résultats expérimentaux montrent que les tâches proposées constituent un défi notable pour les LLM. À travers les différentes FL, la performance varie significativement parmi les FL ciblées et décline systématiquement lorsque la finesse des distinctions s'accroît. Les modèles parviennent relativement bien à maîtriser les relations lexicales correspondant à des distinctions de partie du discours (PdD), mais rencontrent davantage de difficultés lorsqu'il s'agit de relations plus profondes, en particulier celles impliquant la structure actancielle. En outre, les réponses des modèles semblent en partie influencées par la similarité morphologique entre les paires de lexies, surtout lorsque ces indices sont renforcés par les exemples dans la requête.

Mots-clés: Grands modèles de langues, Lexicologie, Fonctions lexicales

Abstract

Large language models (LLMs) exhibit impressive fluency in language use, yet their underlying linguistic ability is not clearly circumscribed. In this study, we propose a new benchmark to assess LLMs’ lexical competence, grounded in the system of Lexical Functions (LFs)—a framework within Meaning–Text Theory (MTT) that formally models the structure of the lexicon through paradigmatic and syntagmatic lexical relations.

We construct a hierarchical classification of LFs by organizing specific functions into broader and more abstract categories, enabling a fine-grained evaluation of LLMs’ ability to distinguish among lexical relations of variable specificity. Our experimental framework targets 82 LFs (or LF classes) drawn from this hierarchy. For each LF target, we design a contrastive task consisting of 20 prompts presented to the models. Each prompt, built on LF instances from the dataset called *French Lexical Network* (LN-fr), includes: (i) a textual definition of the target LF, (ii) a few positive examples (from the target LF) and negative examples (from contrastive sibling LFs), and (iii) a binary (*Yes/No*) question on a new example. LLMs are asked to determine whether the new pair of words instantiates the target LF, given the prior definition and examples.

Experimental results show that the proposed tasks pose a substantial challenge to LLMs. Across different target LFs, model performance varies considerably and systematically declines as the distinctions become more fine-grained. The models handle lexical relations involving part-of-speech distinctions reasonably well, but struggle with deeper LFs, particularly those involving actantial structure. Moreover, model responses appear to be partly driven by morphological similarity between word pairs, especially when such cues are amplified by the prompt design.

Keywords: Large language models, Lexicology, Lexical functions

Table des matières

Résumé	3
Abstract	4
Table des matières	4
Liste des tableaux	7
Liste des figures	9
Liste des listings	11
Liste des sigles et abréviations	13
Remerciements	14
1 Introduction	15
1.1 Problématique	15
1.2 Travaux antérieurs	16
1.2.1 Tests analogiques pour l'évaluation sémantique des modèles de langues	16
1.2.2 Extensions récentes et méthodes alternatives des jeux de données analogiques	18
1.2.3 D'autres approches de l'évaluation de la capacité lexicale des LLM .	20
1.2.4 Limites des approches actuelles	21
1.3 Les fonctions lexicales comme modèle formel pour l'évaluation de la compétence lexicale	22
1.3.1 Le contexte théorique des fonctions lexicales	22
1.3.2 Les fonctions lexicales pour modéliser les relations lexicales	23

1.3.3	Adéquation des fonctions lexicales à notre tâche	23
1.4	Principes de notre proposition	24
1.4.1	Approche générale	24
1.4.2	Cadre expérimental	25
1.4.3	La contribution de la présente recherche	26
1.5	Organisation du mémoire	26
2	Classification des fonctions lexicales	28
2.1	Historique des classifications de Mel'čuk et collègues	28
2.1.1	Classification originale de Žolkovskij et Mel'čuk (1967)	28
2.1.2	Classification de Mel'čuk <i>et al.</i> (1995)	29
2.1.3	Classification de Mel'čuk et Polguère (2021)	32
2.2	Classification de Grimes	35
2.3	Classification de Ramos et Tutin	39
2.4	Classification de Jousse	42
2.4.1	Contexte des classifications précédentes de Jousse	42
2.4.2	Les classes sémantiques	43
2.4.3	Remarques sur cette classification	45
2.5	Classification de Lambrey	46
2.5.1	Problématique et démarche	46
2.5.2	Classification proposée pour les FL syntagmatiques	46
2.6	Classification du présent mémoire	50
2.6.1	Classifier les FL pour le projet	50
2.7	Synthèse des classifications susmentionnées	58
3	Méthodologie expérimentale	62
3.1	Définition de la tâche et aperçu de la méthode	62
3.2	Le jeu de données : RL-fr	63
3.2.1	Sélection des FL ciblées	63
3.2.2	Jeu de données préparé pour l'évaluation	66
3.3	Échantillonnage des exemples contrastifs fondé sur la hiérarchie des FL	68
3.3.1	Motivation pour l'échantillonnage contrastif	68
3.3.2	Stratégie de contraste fondée sur la hiérarchie	69
3.3.3	Pondération équilibrée pour l'échantillonnage	70
3.3.4	Génération du jeu de questions	71

3.4	Exécution des expériences	74
3.4.1	Génération des requêtes	74
3.4.2	Configuration expérimentale	76
3.5	Synthèse	77
4	Résultats et discussion	78
4.1	Performance à travers les LLM	78
4.1.1	Vue d'ensemble	78
4.1.2	Biais de la polarité des réponses	79
4.2	Effets des configurations expérimentales	80
4.2.1	Effets du nombre d'exemples (<i>k</i> -shot)	80
4.2.2	Effet de <i>kw_ctx</i> (contexte du mot-clé) et de <i>vl_pfm</i> (forme propositionnelle de la valeur)	81
4.3	Performance à travers les fonctions lexicales	83
4.3.1	Disparité de performance selon les FL	83
4.3.2	Structures hiérarchiques et performance par FL	85
4.3.3	Défis posés par les FL liées à la structure actancielle	89
4.3.4	Réflexion sur les limites de la hiérarchie des FL	90
4.4	Impact de la similarité morphologique entre les mots-clés et les valeurs des FL	91
4.4.1	Corrélation entre la similarité morphologique et les réponses des modèles	92
4.4.2	Biais induits par la similarité dans les exemples de la requête	95
4.5	Synthèse des résultats	97
5	Conclusion	99
5.1	Synthèse	99
5.2	Limites	101
5.3	Travaux futurs	102
	Bibliographie	103
A	Description des nœuds ciblés	109

Liste des tableaux

1.1	Exemples de fonctions lexicales	23
1.2	Aperçu de l’approche contrastive	25
2.1	Synthèse des classifications de FL	60
2.2	Synthèse des classifications de FL (suite)	61
3.1	Liste des FL simples remplacées par des fonctions complexes sémantiquement équivalentes.	65
3.2	Contenu du jeu d’instances filtrées des FL	68
3.3	Exemples extraits du fichier des instances de FL sélectionnées (le Keyword in Context (KWIC) original correspond à une fenêtre de 13 mots centrée sur le mot-clé, ici raccourcie pour des raisons de lisibilité)	68
3.4	Distribution déséquilibrée des fonctions lexicales (extrait du jeu de données).	71
3.5	Paramètres expérimentaux.	77
4.1	Exactitude (Acc), précision (Prec), rappel (Rec) et score F1 des trois modèles selon les différentes configurations. La précision mesure la fiabilité des réponses, le rappel évalue leur couverture, et le score F1 en donne une moyenne harmonique.	79
4.2	Effet positif de l’augmentation de k — extrait de Adv_1 sur QWEN	82
4.3	Effet négatif avec la présence du KWIC contexte du mot-clé—extrait de Magn sur QWEN	82
4.4	Exemples d’erreurs typiques observées dans les réponses de modèles (QWEN, $k = 10$, $kw_ctx=F$, $vl_pfm=T$)	85
4.5	Contraste de performance entre des FL avec indices d’actants (S_i) et avec indices de circonstants (S_{res} , S_{instr} , etc.).	89

4.6	Exemples d'erreurs révélatrices : les modèles prédisent à tort une correspondance avec la FL cible, en raison de similarités syntaxiques entre fonctions lexicales proches.	90
4.7	Scores d'exactitude des modèles sur une sélection de FL représentatives. . . .	94

Liste des figures

1.1	Exemples de réponses obtenues sur des questions lexicales, via l’interface en ligne de Mistral (https://chat.mistral.ai/chat) en juillet 2025.	16
2.1	FL inversible et FL de détail (Grimes, 1990, p. 360)	36
2.2	Fonctionnement du patron Modification, (Lambrey, 2016, p. 81)	47
2.3	Fonctionnement du patron Préposition, (Lambrey, 2016, p. 84)	47
2.4	Fonctionnement du patron NGovSem, (Lambrey, 2016, p. 85)	48
2.5	Fonctionnement général des indices pour les Verbes supports, (Lambrey, 2016, p. 88)	49
2.6	Fonctionnement général des indices pour les Verbes de réalisation, (Lambrey, 2016, p. 91)	49
3.1	Fragment du RL-fr autour de la lexie AMOUR	64
3.2	Structure hiérarchique des FL utilisée dans ce projet	67
3.3	Illustration d’échantillonnage contrastif structurel entre une fonction cible <i>A</i> et ses voisins hiérarchiques <i>B</i> et <i>C</i> . Les instances associées à <i>A</i> sont utilisées comme exemples positifs, tandis que celles de <i>B</i> et <i>C</i> , issus du même parent, servent de contre-exemples.	70
4.1	Matrices de confusion des trois modèles évalués. Chaque modèle a été interrogé sur un total de 98 400 questions. Nous observons, par exemple, que MISTRAL tend à répondre « Oui » de manière générale, indépendamment de la réponse correcte attendue (45 795 fois pour les cas où la réponse correcte est « oui », et 42 289 fois pour les cas où la réponse correcte est « non »). Les cas où la réponse attendue est « non » sont ainsi fortement sous-classifiés.	80
4.2	Tendances d’exactitude des trois modèles selon les configurations expérimentales	81

4.3	Exactitude des trois LLM sur différentes classes de fonctions lexicales.	84
4.4	Tendances de performance selon la profondeur hiérarchique des fonctions lexicales.	86
4.5	Exemple de dégradation d’exactitude selon la profondeur : branche de Op_{er}_i	87
4.6	Corrélation entre (i) la similarité morphologique de chaque paire de lexies (<i>mot-clé</i> et <i>valeur</i>) et (ii) la polarité de la réponse (<i>oui/non</i>).	93
4.7	Effet de la similarité des k-shot sur l’indice de corrélation des réponses	97

Liste des listings

3.1	Illustration conceptuelle de la tâche d'évaluation contrastive	69
3.2	Structure du fichier json contenant les questions générées pour les FL cibles.	73
3.3	Exemple de requête système	74
3.4	Exemple de requête utilisateur : S_0	75
3.5	Exemple de question posée à la suite d'une requête : S_0	76
4.1	Exemple de requête : verbe support	87
4.2	Exemple de requête : $Oper_i$	88
4.3	Exemple de requête : $Oper_2$	88
4.4	Exemple d'une requête : A_0	96

Liste des sigles et abréviations

FL	fonction lexicale
RSém	représentation sémantique
RSyntP	représentation syntaxique profonde
RSyntS	représentation syntaxique de surface
TAL	traitement automatique des langues
TST	théorie Sens-Texte
DiCo	<i>dictionnaire de combinatoire</i>
LAF	<i>Lexique actif du français</i>
ASyntP	actant syntaxique profond
Vsupp	verbe support
Vreal	verbe de réalisation
GAT	génération automatique de texte
PdD	partie du discours
RL-fr	<i>Réseau lexical du français</i>
KWIC	Keyword in Context
LLM	grand modèle de langues

Remerciements

Je tiens tout d'abord à exprimer ma plus profonde gratitude à mon directeur, François Lareau, pour ses conseils avisés dans le cadre de mon projet de recherche. Sa grande disponibilité, le temps qu'il a consacré à répondre à mes questions, ses relectures attentives de mes travaux ainsi que son soutien constant m'ont été d'une aide inestimable tout au long de mon parcours de maîtrise.

Je souhaite également remercier chaleureusement mon codirecteur, Antoine Venant, pour son aide précieuse dans la réalisation de mes expériences. Il a partagé sans réserve ses connaissances, répondu avec patience à mes questions et m'a toujours encouragé avec bienveillance.

Je remercie le gouvernement du Québec pour l'Exemption des droits de scolarité supplémentaires. Je tiens également à remercier le Conseil de recherches en sciences humaines du Canada (CRSH, subvention RNH02072) ainsi que le Fonds de recherche du Québec – Société et culture (FRQSC, subvention 366841) pour leur soutien financier à cette recherche.

Je n'oublie pas de remercier ma professeure au baccalauréat, Yifan Ma, pour son aide et sa recommandation au moment de la demande à ce programme de maîtrise.

J'ai eu la chance de croiser sur mon chemin de nombreux camarades et amis bienveillants. Parmi eux, je remercie tout particulièrement Shuyang Sun, avec qui j'ai suivi des cours et travaillé sur plusieurs projets, dans une ambiance toujours motivante et solidaire. Je remercie également Li Liu et Shiyu Li, mes amis rencontrés au département, pour leurs conseils précieux et leur généreux partage d'expériences, tant sur le plan des études que sur la vie étudiante à l'UdeM.

Enfin, mes pensées les plus sincères vont à ma famille. Je remercie de tout cœur ma mère et ma sœur pour leur soutien indéfectible et leurs encouragements constants. Je suis également profondément reconnaissant envers mon amour Jiahao Xie, pour sa présence attentive et son dévouement sans faille tout au long de cette aventure académique.

Chapitre 1

Introduction

1.1 Problématique

Les grands modèles de langues (LLM) actuels, comme notamment GPT-4 (OpenAI, 2023), LLaMA (Touvron *et al.*, 2023) et Qwen (Bai *et al.*, 2023), ne se contentent pas de produire des textes cohérents : ils sont également largement mobilisés dans diverses applications du traitement automatique des langues (TAL), telles que la traduction automatique, l'extraction d'information ou encore les questions-réponses, notamment dans des scénarios où les modèles sont guidés par des amorces (*prompts*) pour accomplir une tâche donnée. En outre, ils obtiennent généralement de bonnes performances sur ces tâches (Zhao *et al.*, 2023). Par conséquent, les recherches sur l'évaluation des LLM ont évolué, passant de critères tels que la grammaticalité et la cohérence à l'analyse de leur capacité de raisonnement, de leur exactitude factuelle, de leurs biais, ou d'autres propriétés extralinguistiques, comme le souligne Chang *et al.* (2024).

Cependant, des questions fondamentales demeurent quant à la nature et à la profondeur des connaissances linguistiques que ces modèles acquièrent, ainsi qu'à la manière dont ils accèdent à ces connaissances et les exploitent ou les partagent. Bien que ces LLM semblent pouvoir « employer » la langue avec fluidité, il reste à déterminer dans quelle mesure ils « comprennent » véritablement les structures linguistiques, et jusqu'à quel point ils sont capables de « raisonner » sur des objets linguistiques de façon abstraite.

Le lexique en constitue un exemple particulièrement significatif. Une véritable compréhension de la langue suppose en effet une maîtrise approfondie du lexique. Celui-ci ne se réduit pas à un simple inventaire de mots et de définitions, mais constitue un système

structuré dans lequel les **unités lexicales** (ou **lexies**) sont reliées par un ensemble de relations—telles que la synonymie, l’antonymie, les dérivations morphologiques, l’intensification, entre autres—que l’on retrouve dans toutes les langues. Dès lors, les LLM sont-ils eux aussi capables de maîtriser ces phénomènes lexicaux ? Derrière leurs sorties apparemment « flexibles » et « cohérentes », comme illustré dans la figure 1.1, existe-t-il réellement une compréhension interne du système lexical ? Ou bien ne font-ils que « simuler » cette structure à partir de cooccurrences statistiques et de régularités contextuelles ? Plus encore, peut-on, à l’instar des études linguistiques sur l’humain, évaluer leurs compétences lexicales à l’aide d’une approche guidée par la théorie ?

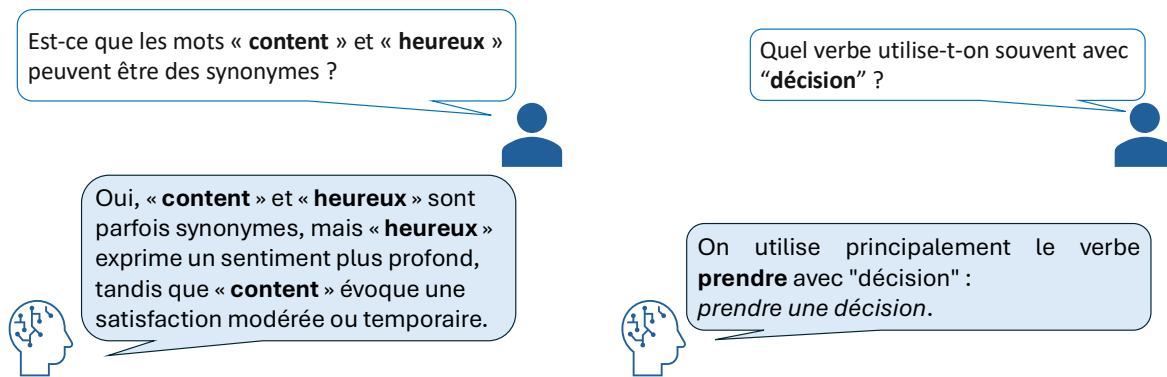


FIGURE 1.1 – Exemples de réponses obtenues sur des questions lexicales, via l’interface en ligne de Mistral (<https://chat.mistral.ai/chat>) en juillet 2025.

En se basant sur ces interrogations, notre présente étude propose d’adopter un cadre théorique existant en linguistique comme outil d’évaluation, afin de construire un test systématique en français permettant d’examiner si les LLM sont capables de reconnaître et de distinguer différents types de relations lexicales. L’objectif est ainsi de mesurer leur **compétence lexicale** au sens linguistique du terme.

1.2 Travaux antérieurs

1.2.1 Tests analogiques pour l’évaluation sémantique des modèles de langues

L’exploitation de différents types de relations lexicales pour évaluer la compétence linguistique constitue de longue date une approche attrayante. Elle est notamment au cœur de

nombreux jeux de test (*benchmarks*) analogiques populaires. Le SAT (Turney *et al.*, 2004) constitue un des premiers jeux de test analogiques. Il se compose de 374 questions à choix multiples (5 options) construites à partir de questions de l'examen SAT (*Scholastic Assessment Test*). Par exemple, une question peut présenter une paire telle que *chat : miaulement*, suivie d'un ensemble d'options : (a) *rat : détalier*, (b) *oiseau : bec*, (c) *chien : aboiement*, (d) *cheval : palefrenier*, (e) *lion : gratter*, dont la bonne réponse est (c), car la relation illustrée dans "*X : Y*" est que *X* désigne l'animal et *Y* le nom du son qu'il produit. Cette étude propose plusieurs nouvelles règles permettant de fusionner les distributions de probabilités produites par différents modules statistiques indépendants, évalués sur le jeu de test analogique. Les résultats montrent que cette approche fait passer la performance de chaque module, initialement proche de la référence aléatoire (20 %), à 31 %, 33 % et 43 % après fusion.

Mikolov *et al.* (2013) ont proposé le *Google analogy test set*, qui contient 8 869 paires de mots reliées par des relations sémantiques (comme *Athènes : Grec, frère : sœur*) et 10 675 paires reliées par des relations morphologiques (comme *cheval : chevaux, facile : facilement*). Ce jeu est utilisé pour évaluer deux nouvelles architectures de modèles d'apprentissage — CBOW (*Continuous Bag-of-Words*) et *Skip-gram*— entraînés sur un large corpus extrait du *Google News*, visant à représenter les mots dans un espace vectoriel de manière plus efficace. La tâche consiste à prédire des analogies vectorielles de la forme : $\vec{\text{roi}} - \vec{\text{homme}} + \vec{\text{femme}} \approx \vec{\text{reine}}$. Cette évaluation est particulièrement difficile, car une prédiction n'est considérée comme correcte que si le mot associé au vecteur prédit correspond exactement à la réponse attendue. Les modèles proposés par les auteurs surpassent largement les approches traditionnelles basées sur les réseaux de neurones — NNLM (*Feedforward Neural Net Language Model*) et RNNLM (*Recurrent Neural Net Language Model*), en particulier lorsque le volume de données d'entraînement est important (783 millions de mots) et que la dimension des vecteurs est suffisamment élevée (jusqu'à 600 dimensions). La capacité des systèmes de représentations vectorielles à reconnaître ce type de relations sémantiques contribue à améliorer la performance dans diverses tâches de traitement automatique des langues, telles que la traduction automatique ou l'extraction d'information.

Dans le prolongement des travaux exploitant les décalages vectoriels pour détecter les relations linguistiques à partir d'espaces de plongement (*embedding*), Gladkova *et al.* (2016) ont proposé le BATS (*Bigger Analogy Test Set*), un jeu de test équilibré composé de 99 200 analogies couvrant 40 types de relations sémantiques et morphologiques. Leurs expériences montrent que les relations dérivationnelles et lexicales restent difficiles à modéliser, avec des taux de réussite relativement faibles, même pour les meilleurs modèles (GloVe ou modèles basés sur SVD). L'étude met également en évidence que les performances ne dépendent pas

systématiquement de la dimension vectorielle ni de la taille des fenêtres de contexte, remettant en question certaines intuitions admises sur la modélisation des relations morphologiques et sémantiques.

Ces bancs d’essai analogiques sont devenus des références incontournables pour l’évaluation des représentations distributionnelles, et ont également été largement mobilisés pour examiner les capacités sémantiques des LLM de dernière génération.

1.2.2 Extensions récentes et méthodes alternatives des jeux de données analogiques

Ushio *et al.* (2021) évaluent ainsi les modèles à l’aide de benchmarks analogiques bien établis, en s’appuyant sur des amorces et les probabilités de complétion associées. Leur étude révèle notamment que les analogies lexicales du jeu BATS sont plus difficiles à traiter pour les modèles que les analogies morphologiques ou encyclopédiques.

E-KAR (pour *Explainable Knowledge-intensive Analogical Reasoning*) constitue une extension récente des jeux de données analogiques. Proposé par Chen *et al.* (2022), ce jeu de données a été construit à partir d’une sélection de questions extraites des concours de la fonction publique chinoise, couvrant des relations analogiques requérant des connaissances linguistiques, encyclopédiques et de sens commun. Il s’agit d’un jeu de test bilingue, comprenant 1 655 questions en chinois, dont 1 251 ont été traduites en anglais. L’une des tâches proposées suit le format classique des questions analogiques à choix multiple, avec des analogies binaires et ternaires. Contrairement aux bancs d’essai précédents, les auteurs distinguent deux conditions expérimentales : l’une est en format original où seules la question et les options sont fournies, celle-ci est définie en mode difficile, et l’autre en mode simple, dans laquelle chaque terme (question et options) est accompagné d’une explication en langue naturelle, validée manuellement, décrivant la nature de la relation analogique. Les résultats montrent que les modèles rencontrent toujours de grandes difficultés sur E-KAR. Même le meilleur modèle testé, ROBERTA-LARGE avec affinage, n’atteint qu’environ 50 % d’exactitude, alors qu’une performance humaine atteint jusqu’à 77,8 % (dans les cas difficiles) et 83,3 % (dans les cas faciles). Cependant, en accord avec les performances humaines, les modèles bénéficient nettement des explications : la présence d’une justification textuelle augmente l’exactitude d’environ 5 points de pourcentage. Particulièrement, les erreurs les plus fréquentes des modèles résident dans les relations de type *Extension* ; par exemple lorsque l’un des termes est un hyperonyme de l’autre (comme *Terre : planète*) ou lorsque l’un est une partie constitutive de l’autre (comme *volant : voiture*). En revanche, les modèles s’en sortent mieux sur

les relations linguistiques plus systématiques, telles que les paires verbe–objet (par exemple *transporter : marchandise*).

Yuan *et al.* (2024) ont proposé l’ANALOGYKB, un jeu de données analogiques à plus grande échelle. Il couvre non seulement des relations analogiques identiques telles que les antonymies (comme *haut : bas :: gauche : droite*), mais aussi des analogies plus complexes et abstraites (comme *Tim Cook : Apple :: Joe Biden : USA*), qui établissent des parallèles entre des paires appartenant à des catégories conceptuelles différentes, comme la relation de ‘dirigeant de l’entreprise’ et celle de ‘chef d’État’ dans l’exemple. Ce jeu rassemble plus d’un million d’instances analogiques, issues d’un filtrage sur *Wikidata* et *ConceptNet*, couvrant au total 943 types de relations analogiques.

D’une part, les modèles pré-entraînés sur ANALOGYKB tels que ROBERTA-LARGE et DEBERTA-v3, présentent une amélioration significative sur d’autres jeux de données analogiques existants dans des tâches de choix multiple, comme *Google analogy test set* (Mikolov *et al.*, 2013), *BATS* (Gladkova *et al.*, 2016). Ces gains de performance sont d’autant plus marqués que la taille du jeu d’entraînement augmente (par exemple avec T5-LARGE). L’analyse montre également que les relations analogues les plus complexes incluses dans ANALOGYKB jouent un rôle notable dans l’amélioration des capacités des modèles à reconnaître des analogies complexes (notamment chez ROBERTA-LARGE).

D’autre part, les modèles entraînés sur ANALOGYKB obtiennent également de meilleurs résultats dans des tâches de complétion analogique de type $A : B :: C : X$, où il s’agit de prédire le quatrième terme X . Selon Yuan *et al.*, cette amélioration est observée tant chez des modèles de taille moyenne (T5-LARGE) que chez les LLM comme INSTRUCTGPT-3 et CHATGPT, y compris dans des types de relations hors distribution, comme les relations scientifiques ou métaphoriques. Enfin, une évaluation humaine montre que l’exposition aux amorces construites à partir des exemples de ANALOGYKB améliore sensiblement la capacité d’explication des relations analogiques par les modèles comme INSTRUCTGPT-3 et CHATGPT¹, en renforçant leur aptitude à expliciter les liens conceptuels sous-jacents aux analogies.

De leur côté, Wijesiriwardene *et al.* (2023) ont proposé le banc d’essai ANALOGICAL, qui se concentre sur des données analogiques présentant une forte similarité sémantique. Ce benchmark regroupe 13 sous-jeux de données issus de corpus analogiques existants, réorganisés selon une taxonomie ascendante en six niveaux de complexité croissante : (i) des

1. La version exacte n’est pas précisée dans Yuan *et al.* (2024).

analogies entre mots ou paires de mots, à la manière des jeux classiques ; (ii) des correspondances entre un mot et une phrase équivalente ; (iii) des paires de phrases, où l’une résulte d’une alternance aléatoire de l’autre (suppression, masquage ou permutation de mots) ; (iv) des phrases et leur négation ; (v) des phrases et leur implication ; et (vi) des métaphores mises en parallèle avec leur reformulation explicite en langage naturel. Au fur de ces niveaux dans la taxonomie, les textes deviennent plus longs et les relations analogiques plus abstraites.

L’étude a évalué huit modèles (dont T5, LINKBERT, ROBERTA) à l’aide de diverses métriques fondées sur la distance vectorielle entre les éléments analogiques prédits. En général, une distance plus faible indique une meilleure performance, sauf pour la tâche de négation, où une plus grande distance est attendue entre une phrase et sa forme niée. Les résultats montrent que, hormis le niveau (iii) où les données sont relativement brutes, les performances des modèles diminuent nettement à mesure que la complexité des relations analogiques augmente. Cette tendance souligne les limites des modèles en ce qui concerne la capture des relations implicites ou abstraites dans des analogies linguistiques complexes.

Plus récemment, Petrov *et al.* (2025a,b,c) ont développé un nouveau banc d’essai appelé ALF (*Analogies lexicales du français*), basé sur des analogies, en adoptant une approche similaire à la nôtre, fondée sur l’exploitation d’occurrences de FL issues de *Réseau lexical du français* (RL-fr). Les auteurs ont sélectionné 25 FL courantes (21 paradigmatiques et 4 syntagmatiques) afin de construire un jeu de données présentant un niveau de complexité élevé, composé de 2 600 analogies lexicales fines. Ce jeu de données a été évalué sur plusieurs modèles, dont CHATGPT-4O-MINI, LLAMA 3.0-8B, LLAMA 3.1-8B et QWEN 2.5-14B. Les résultats montrent que les modèles de taille moyenne obtiennent de bonnes performances sur les relations morphologiques dérivationnelles, mais rencontrent davantage de difficultés avec les relations syntagmatiques ainsi qu’avec la distinction des rôles d’événement et de participant.

1.2.3 D’autres approches de l’évaluation de la capacité lexicale des LLM

À côté des bancs d’essai analogiques, d’autres approches ont également proposé des méthodes visant à évaluer plus directement la compétence lexicale des LLM. Ettinger (2020) a introduit une batterie de tests inspirée des protocoles de diagnostic en psycholinguistique, appliquée notamment à BERT. Cette évaluation inclut des tâches variées telles que l’inférence pragmatique ou de sens commun, la reconnaissance des rôles sémantiques, ainsi que la connaissance des événements. Les résultats montrent que le modèle démontre une bonne robustesse pour retrouver les hyperonymes de noms, mais il éprouve des difficultés notables

face aux inférences complexes et à la prédiction d'événements fondée sur les rôles sémantiques². En particulier, le modèle se révèle clairement insensible aux effets contextuels induits par la négation.

La capacité lexicale des LLM à traiter les expressions figées (dans le cadre de la phraséologie) constitue également un aspect important à évaluer. L'étude de Liu et Lareau (2024), par exemple, propose un jeu de données construit à partir des locutions extraites du Réseau Lexical du Français (RL-FR) afin d'évaluer la performance du modèle CAMEMBERT dans la prédiction de mots masqués au sein de locutions, en comparaison avec des expressions libres contenant des lexèmes simples. Les résultats montrent que : (1) CAMEMBERT prédit significativement mieux les unités lexicales faisant partie d'une locution que les lexèmes simples hors contexte idiomatique ; (2) le modèle n'est pas sensible aux variations du degré d'idiomaticité entre sous-types de locutions ; (3) il affiche une performance particulièrement élevée pour la prédiction des mots grammaticaux, indépendamment du caractère idiomatique ; (4) une corrélation positive est observée entre la longueur des locutions et la qualité des prédictions. Cette étude éclaire un aspect complémentaire de la compétence lexicale des LLM : la capacité à traiter les expressions figées issues du RL-fr. Elle montre notamment que la performance des modèles varie selon la nature phraséologique des unités, ce qui confirme l'importance de prendre en compte différents types de relations lexicales dans l'évaluation.

1.2.4 Limites des approches actuelles

Cependant, ces jeux de données analogiques souffrent souvent d'un manque de fondement théorique et d'une couverture inégale. Par exemple, BATS (Gladkova *et al.*, 2016), l'un des jeux les plus équilibrés, diversifiés et systématiquement construits à ce jour, couvre très peu de relations lexicales syntagmatiques—celles liées à la combinaison des lexies plutôt qu'à leur substitution—, et néglige de nombreux phénomènes relevant du sens plutôt que de la morphologie stricte (comme l'analogie *continuer : continuation : : dormir : sommeil*). Bien que ces ressources soient des outils précieux pour évaluer des capacités linguistiques et de raisonnement de haut niveau, elles s'éloignent de l'évaluation de la compétence lexicale à proprement parler, ce qui limite leur utilité pour les études purement linguistiques.

2. Par *rôles sémantiques*, Ettinger (2020) entend les fonctions que les entités remplissent dans un événement, telles que l'agent, le patient ou l'instrument. Ces rôles sont inférés à partir de la structure syntaxique de la phrase, et doivent être associés à des connaissances sur les interactions typiques entre les types d'entités impliquées. Le modèle est ainsi évalué sur sa capacité à relier la forme syntaxique à une interprétation sémantique fondée sur la connaissance des événements.

Alors que la maîtrise des relations paradigmatiques telles que la synonymie ou l'hyponymie est fréquemment testée dans ces travaux, d'autres types de connaissances, comme celles mobilisées dans les constructions à verbe support (par exemple *appliquer une pression*) ou liées à la structure actancielle (comme les paires du type *médecin–patient*), sont souvent négligées.

Notre travail vise ainsi à remédier à ce manquement en proposant un **banc d'essai centré exclusivement sur les relations lexicales**.

1.3 Les fonctions lexicales comme modèle formel pour l'évaluation de la compétence lexicale

1.3.1 Le contexte théorique des fonctions lexicales

Nous nous appuyons sur le cadre théorique de la TST, développée par Mel'čuk et ses collègues depuis les années 1960 (Mel'čuk, 1973, 1996, 2016; Mel'čuk *et al.*, 1995; Mel'čuk et Polguère, 2021). La TST propose une modélisation linguistique orientée vers la production de texte à partir du sens, en mettant l'accent sur le rôle central du **lexique** et de ses propriétés combinatoires.

Selon la TST, le passage de la représentation sémantique (RSém) à la représentation syntaxique profonde (RSyntP) implique des opérations de sélection lexicale, que ce soit par un locuteur humain ou un système informatique. Une première forme de sélection concerne le choix d'une unité lexicale en fonction du contenu sémantique que l'on souhaite exprimer. Par exemple, pour exprimer 'véhicule automobile à quatre roues conçu pour le transport individuel de quelques personnes...', on choisira la lexie VOITURE.

Une autre forme de sélection dépend, quant à elle, d'une unité lexicale déjà choisie. Ainsi, une fois le mot VOITURE sélectionné, si l'on souhaite désigner la personne qui la conduit, on utilisera CONDUCTEUR ou CHAUFFEUR. De même, pour exprimer son fonctionnement, on optera probablement pour le verbe ROULER.

Pour représenter de manière formelle la structure du lexique et les connexions entre les unités lexicales, la TST s'appuie sur un système de **FL**, qui représentent des relations paradigmatique ou syntagmatique stables et récurrentes entre unités lexicales—c'est-à-dire entre des mots pris dans un sens déterminé.

1.3.2 Les fonctions lexicales pour modéliser les relations lexicales

D’après Mel’čuk *et al.* (1995); Mel’čuk (1996), le terme de « fonction » est ici utilisé, comme indiqué précédemment, au sens mathématique du terme : $f(X) = Y$. Formellement, une FL f est une fonction qui associe à une expression lexicale donnée L — appelée l’argument de f — un ensemble $\{L_i\}$ d’expressions lexicales, qui constituent la valeur de f . Ces expressions lexicales expriment, en fonction de L , un sens spécifique associé à f :

$$f(L) = \{L_i\}.$$

Ainsi, chaque FL encode une relation sémantique et/ou syntaxique spécifique entre une unité lexicale (appelée dorénavant **mot-clé**) et un ensemble d’unités lexicales (sa **valeur**). Le tableau 1.1 illustre quelques-unes des FL les plus courantes :

Relation	FL	Exemple
Synonyme	Syn	Syn(VOITURE) = AUTOMOBILE
Antonyme	Anti	Anti(PETIT) = GRAND
Nominalisation	S ₀	S ₀ (PARTIR) = DÉPART
Intensificateur	Magn	Magn(BOIRE) = ‘COMME UN TROU’
Verbe Support	Oper ₁	Oper ₁ (DÉCISION) = PRENDRE

TABLEAU 1.1 – Exemples de fonctions lexicales

1.3.3 Adéquation des fonctions lexicales à notre tâche

Les fonctions lexicales constituent un cadre idéal pour l’évaluation fine de la compétence lexicale des LLM. En effet, elles offrent un **système structuré et riche** des relations lexicales paradigmatiques et syntagmatiques. Ce cadre présente plusieurs avantages :

- Il permet de tester aussi bien des relations maîtrisées par les LLM (comme la synonymie) que des phénomènes plus subtils, souvent négligés (comme les collocations à verbe support ou les relations sémantiques actanciennes) ;
- Il s’appuie sur une tradition lexicographique bien établie, offrant une classification **structurée** des relations lexicales ainsi qu’un ensemble de ressources d’**exemples contrôlés** ;
- Il rend possible une **analyse fine** des performances des modèles, selon les types et la complexité des relations lexicales impliquées.

En nous fondant sur ce cadre, nous visons à dépasser les limites des évaluations analogiques traditionnelles, en proposant un banc d’essai systématique et théoriquement fondé, afin d’explorer en profondeur la capacité des LLM à reconnaître et manipuler les relations lexicales qu’elles représentent.

1.4 Principes de notre proposition

1.4.1 Approche générale

Dans le cadre de notre évaluation de la compétence lexicale des LLM, nous reformulons la **question centrale** suivante : *dans quelle mesure les LLM peuvent-ils être sollicités, via des amorces, pour reconnaître si une paire d’unités lexicales correspond à un type donné de relation lexicale ?*

Afin de répondre à cette question, nous nous appuyons sur le formalisme des fonctions lexicales (FL) qui fournit un cadre structuré et théoriquement fondé pour modéliser une large variété de relations lexicales. Bien que notre expérimentation s’appuie sur des données en français, issues du *Réseau lexical du français* (RL-fr), le système présenté vise à être essentiellement indépendant de toute langue : les FL reposent sur des phénomènes lexicaux récurrents observés dans de nombreuses langues, et constituent ainsi une base d’évaluation linguistiquement neutre.

Nous établissons d’abord une classification hiérarchique de FL tout en les regroupant à différents niveaux selon leurs propriétés sémantiques et syntaxique ; une présentation détaillée de cette hiérarchie est fournie au chapitre 2. Nous sélectionnons un sous-ensemble pertinent de nœuds de FL (ou groupe de FL) à partir de cette hiérarchie pour exercer l’évaluation, dont chacun pourrait être *ciblé* à formuler les amorces dans notre protocole d’évaluation.

Par exemple, à un niveau de granularité grossier, nous évaluons si les LLM sont capables de reconnaître un adjectif dérivé sémantiquement à partir d’une lexie donnée (comme DÉTRUIRE → DESTRUCTIF vs DÉTRUIRE → DESTRUCTION). À un niveau plus fin, nous examinons leur capacité à distinguer les dérivations adjectivales sémantiquement neutres (comme DÉTRUIRE → DESTRUCTIF) de celles impliquant un glissement de sens plus marqué (comme DÉTRUIRE → DESTRUCTIBLE).

Pour chaque FL ciblée, nous associons un ensemble de fonctions *contrastives* partageant certaines propriétés avec la cible mais s’en distinguant par d’autres. Par exemple, certaines

relations lexicales appartiennent toutes à la catégorie de la dérivation adjectivale, mais présentent néanmoins des différences internes—comme illustré dans le tableau 1.2. Nous demandons alors aux modèles de langue d’identifier les paires de mots relevant de la fonction cible, et de rejeter celles qui relèvent des fonctions contrastives.

Relation ciblée	Niveau de granularité	Relation contrastive
DÉTRUIRE → DESTRUCTIF	Niveau grossier : contrastée avec une dérivation d’une autre catégorie grammaticale (ex. une nominalisation)	DÉTRUIRE → DESTRUCTION
DÉTRUIRE → DESTRUCTIF	Niveau fin : contrastée avec une autre dérivation adjectivale, mais qui exprime une propriété différente, comme ‘ce qui peut être détruit’ plutôt que ‘ce qui détruit’	DÉTRUIRE → DESTRUCTIBLE

TABLEAU 1.2 – Aperçu de l’approche contrastive

1.4.2 Cadre expérimental

Notre plateforme expérimentale repose principalement sur une approche par *amorçage*, dans laquelle chaque *amorçe* comprend une description linguistique de la FL ciblée, ainsi que des exemples contrastifs positifs et négatifs.

Afin de générer automatiquement des paires lexicales correspondant à différentes FL, nous avons exploité une ressource lexicographique en français, le *Réseau lexical du français* (RL-fr) (Lux-Pogodalla et Polguère, 2011; ATILF, 2024), à partir de laquelle nous avons extrait et sélectionné les paires pertinentes selon nos critères. Ces paires jouent le rôle d’exemples d’apprentissage dans les amorces. Pour chaque amorçe ainsi construite, un nouvel exemple est soumis au modèle, accompagné d’une question visant à déterminer s’il relève, lui aussi, de la FL cible ou non, en complément des exemples déjà fournis.

Nous avons produit un jeu de données de test composé de ces amorces structurées, couvrant plusieurs dizaines de FL, et l’avons soumis à divers modèles de langage afin de mener une évaluation systématique à grande échelle. Les détails du cadre expérimental sont présentés au chapitre 3.

1.4.3 La contribution de la présente recherche

Comme nous l’avons présenté dans la §1.2, plusieurs travaux ont déjà exploré la capacité lexicale des LLM. Parmi les plus récents, l’étude de l’ALF (Petrov *et al.*, 2025a,b,c) propose un jeu de données analogiques françaises construites à partir d’instances de FL issues du *RL-fr*. Ce cadre ouvre sans doute des perspectives nouvelles pour l’évaluation fine de la capacité lexicale des LLM.

Notre approche s’en distingue toutefois à plusieurs égards. Plutôt que de traiter les relations lexicales comme des paires isolées, nous les organisons en une structure de catégories hiérarchique, regroupant des relations spécifiques au sein de classes plus générales, afin d’analyser la capacité des modèles à opérer des distinctions à différents niveaux de spécificité. Par ailleurs, au lieu de demander directement aux modèles de résoudre une équation analogique donnée (tâche à réponse ouverte), nous recourons à des questions fermées de type *Oui/Non*, enrichies de contextes plus élaborés. Si cette reformulation rend sans doute la tâche moins difficile, elle permet en revanche de contourner certaines limites propres au format analogique brut : en particulier, le manque d’information explicite fourni au modèle, ou le risque de faux négatifs lors de l’évaluation. Ce format nous autorise également à intégrer, dans les amorces, des indices relatifs au sens des mots ou aux rôles sémantiques, et ainsi à explorer un éventail plus large et plus équilibré de relations lexicales.

En somme, nous proposons tout d’abord une classification hiérarchique et structurée des FL, qui organise les FL cibles en fonction de distinctions spécifiques entre relations lexicales apparentées mais contrastives à différents niveaux d’abstraction. Nous introduisons ensuite un jeu d’évaluation fondé sur des questions polarisées visant à tester la capacité lexicale des LLM à reconnaître ces contrastes de manière fine. Plusieurs modèles récents à poids ouverts sont évalués sur ce benchmark, avec une attention particulière portée à l’effet de différentes configurations d’*amorçage*. Enfin, nous examinons l’influence de certains indices de surface sur le comportement des LLM testés.

1.5 Organisation du mémoire

Ce mémoire est organisé de la manière suivante. Le **chapitre 2** couvre le développement de une hiérarchie de classification des fonctions lexicales (FL), fondé sur la théorie Sens-Texte issue des études en lexicographie, et ajusté pour répondre aux exigences de notre

évaluation. Le **chapitre 3** détaille notre protocole expérimental, y compris les données employées, les méthodes de création des contrastes, la génération des amorces, ainsi que les modalités précises pour leur production et exécution. Le **chapitre 4** se concentre sur l'exposition et l'analyse des résultats obtenus, avec une double approche : d'une part, les performances des modèles linguistiques évalués, et d'autre part, les caractéristiques des FL examinées. Ce chapitre explore aussi comment ces résultats enrichissent notre compréhension des compétences des LLM à représenter et manier des connaissances lexicales structurées. Enfin, nous concluons dans le **chapitre 5** avec une synthèse et des pistes de travaux futurs.

Chapitre 2

Classification des fonctions lexicales

Dans le but d'évaluer la capacité lexicale des LLM à partir des données de fonctions lexicales (FL) dans le cadre de la théorie Sens-Texte (TST), nous avons besoin d'un modèle structuré permettant de classifier ces FL selon leur diversité de granularité. En nous appuyant sur les travaux antérieurs consacrés aux FL (Žolkovskij et Mel'čuk, 1967, 1970; Mel'čuk *et al.*, 1995; Mel'čuk et Polguère, 2021; Ramos et Tutin, 1996; Grimes, 1990; Lambrey, 2016; Jousse, 2010) qui fournissent des références fondamentales, nous proposons notre propre système de classification, spécifiquement adapté à notre tâche d'évaluation.

2.1 Historique des classifications de Mel'čuk et collègues

2.1.1 Classification originale de Žolkovskij et Mel'čuk (1967)

Žolkovskij et Mel'čuk (1967) présentent une première division des fonctions lexicales, les divisant en deux catégories—les *Substituts* et les *Paramètres sémantiques* :

les Substituts—mots ou locutions susceptibles d'être employés dans le texte à la place du mot-clé, cette substitution n'exigeant que des modifications purement syntaxiques visant à sauvegarder les sens original.

les Paramètres sémantiques—mots ou locution susceptibles d'être employés avec le mot-clé pour exprimer d'une façon idiomatique certains sens ('très', 'faire', 'créer', etc.).

(Žolkovskij et Mel'čuk, 1967, p. 4)

Dans le travail de Žolkovskij et Mel'čuk (1970), les fonctions comme Syn , $Conv_{ij}$, S_0 ,

A_0 , V_0 et Adv_0 sont classées parmi les *Substituts*, tandis que toutes les autres FL sont simplement regroupées sous la classe des *Paramètres sémantique*. Cela constitue les bases principales de la classification ultérieure en fonctions syntagmatiques et paradigmatisques.

2.1.2 Classification de Mel'čuk *et al.* (1995)

Mel'čuk *et al.* (1995) considèrent les fonctions lexicales comme une composante essentielle de la rédaction d'un dictionnaire, car elles permettent de fournir aux apprenants et aux utilisateurs de la langue des moyens d'expression à la fois riches et variés. Afin de mettre en lumière cette contribution des fonctions lexicales, l'ouvrage en question a établi l'inventaire suivant des FL :

1. FL paradigmatisques :

(a) FL de Base :

- i. Synonyme : Syn
- ii. Conversif : $Conv_{ij}$
- iii. Antonyme : $Anti$

(b) FL apparentées aux premières :

- i. Contrastif : $Contr$
- ii. Épithète pléonastique : $Epit$
- iii. Générique : $Gener$
- iv. Figuratif : $Figur$

(c) FL de dérivation syntaxique :

- i. Nominalisation : S_0
- ii. Verbalisation : V_0
- iii. Adjectivisation : A_0
- iv. Adverbialisation : Adv_0

(d) FL nominales :

- i. Dérivés sémantiques nominaux actanciels : S_i
- ii. Dérivés sémantiques nominaux circonstanciels :
 - Nom d'instrument : S_{instr}
 - Nom de moyen : S_{med}
 - Nom de manière : S_{mod}

- Nom de lieu : S_{loc}
- Nom de résultat : S_{res}

- iii. Singulatif : $Sing$
- iv. Collectif : $Mult$
- v. Nom de chef : Cap
- vi. Nom d'équipe : $Equip$
- vii. Nom de démarrage : $Germ$
- viii. Nom du point culminant : $Culm$

(e) FL adjectivales :

- i. Dérivés sémantiques adjectivaux actanciels : A_i
- ii. Dérivés sémantiques adjectivaux potentiels : $Able_i$
- iii. Dérivés sémantiques adjectivaux virtuels : $Qual_i$

2. FL syntagmatiques :

(a) FL adjectivales :

- i. Intensificateur : $Magn$
- ii. Comparatifs :
 - $Minus$
 - $Plus$
- iii. Confirmateur : Ver
- iv. Laudatif : Bon
- v. Péjoratif : $Pejor$
- vi. Positif : Pos ¹

(b) FL adverbiales :

- i. Dérivés sémantiques adverbiaux : Adv_i
- ii. Préposition instrumentale : $Instr$
- iii. Préposition locative :
 - LOC_{in}
 - LOC_{ad}
 - LOC_{ab}

1. Les fonctions telles que $Pejor$ et Pos sont en fait des fonctions historiques qui ne font plus partie du système des FL standard. Cependant, il est toujours pertinent de les mentionner ici pour illustrer l'évolution de ces concepts.

- iv. Consécutif : Propt
- (c) FL verbales :
 - i. Pred
 - ii. Verbes supports :
 - A. Oper_i
 - B. Func_i
 - C. Labor_{ij}
 - iii. Verbes de réalisation :
 - A. Real_i
 - B. Fact_i
 - C. Labreal_{ij}
 - iv. Verbes phasiques :
 - A. Incep
 - B. Cont
 - C. Fin
 - v. Verbes causatifs :
 - A. Caus
 - B. Perm
 - C. Liqu
 - vi. Verbe d'implication : Involv
 - vii. Verbe de manifestation : Manif
 - viii. Verbe de préparation : Prepar
 - ix. Verbe d'état proche : Prox
 - x. Verbe de dégradation : Degrad
 - xi. Verbe de son typique : Son
 - xii. Expression impérative : Imper
 - xiii. Verbe résultatif : Result_i
 - xiv. Expression de fonctionnement difficile : Obstr
 - xv. Expression de fonctionnement excessif : Excess
 - xvi. Expression d'arrêt de fonctionnement : Stop
 - xvii. Expression de symptôme d'un état : Sympt_{ijk}

2.1.3 Classification de Mel'čuk et Polguère (2021)

Après quelques décennies d'analyses et de discussions, Mel'čuk et Polguère (2021) ont réussi à organiser et à clarifier la littérature abondante et complexe sur les FL afin de produire une version actualisée du système des FL. Ce travail présente les FL de manière plus claire et plus structurée que les travaux antérieurs, rendant ainsi le cadre plus cohérent et complet.

Voici l'inventaire détaillé de cette étude, qui se base aussi sur la dichotomie entre relations paradigmatiques et syntagmatiques :

1. FL paradigmatiques :

(a) Trois piliers du système de FL :

- i. Synonyme : *Syn*
- ii. Antonyme : *Anti*²
- iii. Conversif : *Conv_{ij}*

(b) FL apparentées à *Syn* et *Anti* :

- i. Terme générique : *Gener*
- ii. Terme figuratif : *Figur*
- iii. Terme contrastif : *Contr*

(c) Dérivations structurales :

- i. Nominalisation : *S₀*
- ii. Verbalisation : *V₀*

2. Le sens de la fonction *Anti* est raffiné dans cet article, en fonction de la négation aux différents niveaux de la définition lexicographique des lexies. Autrement dit, *Anti* pourrait être considéré comme une généralisation des quatre types d'antonymes :

- Antonymes contradictoires (opposés par une négation au plus haut niveau), par exemple, *Anti*(PRÉSENT) = ABSENT :
'*x* est absent' = '*x* est **non** présent'
- Antonymes contraires (opposés par une négation enchâssée sous le prédicat central), par exemple, FERMER est un antonyme contraire de OUVRIR :
'*x* ouvre *y*' = '*x* cause que *y* est ouvert'
'*x* ferme *y*' = '*x* cause que *y* n'est **pas** ouvert'
- Antonymes scalaires (opposés par les composantes 'plus' ~ 'moins'), par exemple, LOURD et LÉGER sont des antonymes scalaires :
'*x* est lourd' = '*x* a un poids plus grand que le poids normal pour les *x*'
'*x* est léger' = '*x* a un poids moins grand que le poids normal pour les *x*'
- Antonymes de positionnement (positions opposées géométriques ou autres), par exemple, GAUCHE et DROITE sont des antonymes de positionnement :
'*x* est à gauche' = '*x* est du côté qui est celui du coeur'
'*x* est à droite' = '*x* est du côté qui n'est pas du coeur'.

- iii. Adjectivisation : A_0
 - iv. Adverbialisation : Adv_0
 - v. Clausativisation : $Claus$
- (d) Dérivation de prédicativisation :
- i. $Pred$
- (e) Dérivations des noms d'actants et de circonstants :
- i. Nom typique de l'actant syntaxique profond (ASyntP) i de L : S_i
 - ii. Équipe qui fait fonctionner L : $Equip$
 - iii. Chef de L : Cap
 - iv. Nom typique de circonstant d'instrument : S_{instr}
 - v. Nom typique de circonstant de moyen : S_{med}
 - vi. Nom typique de circonstant de manière : S_{mod}
 - vii. Nom typique de circonstant de lieu : S_{loc}
 - viii. Nom typique de circonstant de résultat : S_{res}
- (f) Dérivations des adjectifs actanciels :
- i. Qualificatif adjectival typique de l'ASyntP i de L : A_i
 - ii. Qualificatif adjectival typique de l'ASyntP i potentiel de L : $Able_i$
 - iii. Qualificatif adjectival typique de l'ASyntP i éventuel de L : $Qual_i$
- (g) Dérivation des adverbes actanciels :
- Adverbe signifiant 'en étant i de L ' : Adv_i
- (h) Dérivations de type singulatif, collectif, et autres significations proches de la flexion :
- i. Singulatif : $Sing$
 - ii. Collectif (pour les verbes : multiplicatif) : $Mult$
 - iii. Impératif : $Imper$
 - iv. Perfectif : $Perf$
 - v. Imperfectif : $Imperf$
 - vi. Résultatif : $Result_i$
2. FL syntagmatiques :
- (a) Collocatifs nominaux désignant des «états» de L :

- i. Origine : Germ
 - ii. Culminatif : Culm
- (b) Collocatifs modificateurs :
- i. Épithète pléonastique à valeur stylistique : Epit
 - ii. Épithète pléonastique à valeur informative de précision : Redun
 - iii. Intensificateur Magn
 - iv. Évaluateur d'adéquation : Ver
 - v. Évaluateur laudatif : Bon³
 - vi. Comparatif positif : Plus
 - vii. Comparatif négatif : Minus
- (c) Collocatifs prépositionnels :
- i. Préposition locative de positionnement : LOC_{in}
 - ii. Préposition locative de destination : LOC_{ad}
 - iii. Préposition locative de provenance : LOC_{ab}
 - iv. Préposition instrumentale : Instr
 - v. Préposition causale : Propt
- (d) Collocatif verbe copule :
- Verbe qui signifie 'être' : Copul
- (e) Collocatifs verbes supports :
- i. Func_i
 - ii. Oper_i
 - iii. Labor_{ij}
- (f) Collocatifs verbes de réalisation :
- i. Fact_i
 - ii. Real_i
 - iii. Labreal_{ij}
 - iv. Verbe collocatif proche de 'préparer' : Prepar
- (g) Collocatifs verbes phasiques et assimilables :

3. Selon Mel'čuk et Polguère (2021), la fonction Degrad est équivalente à IncepPredAntiBon, alors qu'elle avait été raccourcie comme une FL simple dans Mel'čuk *et al.* (1995) avec le sens 'se dégrader ou devenir pire', le même cas pour Liqu=CausFin, Perm=NonCausFin, etc.

- i. Verbe collocatif qui signifie ‘commencer’ : *Incep*
 - ii. Verbe collocatif qui signifie ‘continuer’ : *Cont*
 - iii. Verbe collocatif qui signifie ‘cesser’ : *Fin*
 - iv. Verbe collocatif—non-phasique—qui signifie ‘être sur le point [de L]’ : *Prox*
- (h) Collocatifs verbes de mode de fonctionnement :
- i. Verbe collocatif qui signifie ‘ne pas fonctionner comme il faut’ : *Obstr*
 - ii. Verbe collocatif qui signifie ‘s’interrompre’ : *Stop*
 - iii. Verbe collocatif qui signifie ‘fonctionner de façon excessive’ : *Excess*
- (i) Collocatifs verbes de causation :
- i. Verbe collocatif qui signifie ‘causer’ : *Caus*
 - ii. Verbe collocatif qui signifie ‘faire cesser’ : *Liqu*
 - iii. Verbe collocatif qui signifie ‘ne pas faire cesser, permettre’ : *Perm*
- (j) Autres collocatifs verbaux :
- i. Verbe collocatif qui signifie ‘produire un son typique’ : *Son*
 - ii. Verbe collocatif qui signifie ‘se manifester dans qqch’ : *Manif*
 - iii. Verbe collocatif qui signifie ‘agir sur qqch’ : *Involv*
 - iv. Verbe collocatif qui signifie ‘montrer des symptômes [de L] dans un élément de son corps, **littéralement ou métaphoriquement**’ : *Sympt_{ijk}*

2.2 Classification de Grimes

La méthode de Mel’čuk et de ses collègues pour classifier les FL en les divisant en catégories paradigmatiques et syntagmatiques, a été largement adoptée. Toutefois, des incohérences ont été relevées par Grimes (1990) : certaines FL paradigmatiques permettent la cooccurrence de la valeur et du mot-clé (comme *Gener* : *Gener*(CULPABILITÉ) = SENTIMENT [de ~]), tandis que des valeurs fusionnées peuvent apparaître parmi les FL syntagmatiques (comme *Magn* : *Magn*(DÉFAITE) = DÉBÂCLE). Particulièrement, la fonction *Gener*, selon Mel’čuk, est limitée aux lexies en collocation syntaxiquement, alors que toutes les autres relations générique–spécifique sont traitées comme des « types » de relations encyclopédiques, au lieu d’être incluses comme FL. Grimes le désapprouve en justifiant que cette approche a négligé le fort parallélisme entre ces formes et l’information encyclopédique. Selon ses notes dans son article (Grimes, 1990, p. 358), il pourrait être avantageux pour la gestion de l’information d’englober tous ces genres de relations dans le mécanisme des FL.

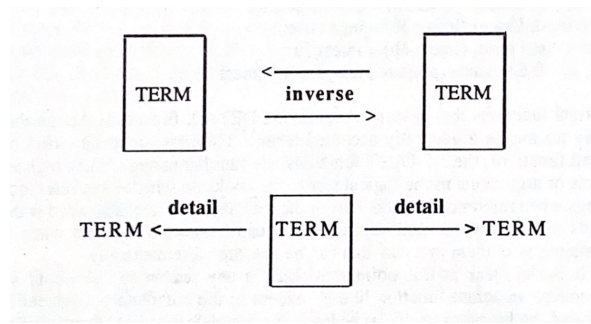


FIGURE 2.1 – FL inversible et FL de détail (Grimes, 1990, p. 360)

Grimes reconnaît que caractériser le comportement des FL comme paradigmatic ou syntagmatic relève d'une tradition linguistique de longue date, mais cette perspective ne correspond pas toujours de manière parfaitement cohérente aux faits observés. En revanche, il propose une nouvelle façon de classer les FL en introduisant la notion de fonction inverse (*inverse function*). Selon lui, les FL peuvent être réparties en 2 types distincts : les fonctions **inversibles** et les fonctions de **détail**.

D'une part, une fonction est dite inversible lorsqu'il existe une autre FL qui inverse le mot-clé et la valeur de la première. Par exemple, $A_0(\text{MONDE}) = \text{MONDIAL}$, et à l'inverse, $S_0(\text{MONDIAL}) = \text{MONDE}$. Ainsi, S_0 et A_0 sont toutes deux des FL inversibles, et l'une est la fonction inverse de l'autre. D'autre part, pour les FL pour lesquelles on ne peut pas trouver une autre FL qui l'inverse, Grimes les classe comme les FL de détail. Par exemple les FL de collocations verbales, $Op_{er_1}(\text{SOMMEIL}) = \text{AVOIR}$, mais il n'existe pas de FL prenant AVOIR comme mot-clé et retournant SOMMEIL comme valeur. La distinction de ces deux classes est représentée dans la figure 2.1.

En se basant sur cela, Grimes propose la classification ci-dessous :

1. FL inversibles (celles qui ont leur inverse dans le système des FL)
 - (a) FL de catégorie (dérivations syntaxiques selon Mel'čuk)
 - i. N_0 (S_0 de Mel'čuk), V_0 , A_0 et Adv_0
 - ii. $Pred$
 - iii. $Copul$
 - (b) FL inversibles simples (celles qui sont directement jumelées avec leur inverse)
 - i. Syn_c et Syn_s
 - ii. Antecedent et Consequent ($Result_i$ de Mel'čuk et son inverse)
 - iii. $Liqu$ et son inverse $State$

- iv. Sympt_{ijk} et son inverse Manifest
 - v. Cap et son inverse Equip
 - vi. Sing et son inverse Mult
- (c) FL inverses symétriques (celles dont l'inverse est la même FL)
- i. Syn (exact)
 - ii. Conv_{ij}
 - iii. Anti
- (d) FL inverses groupées (*Clustered inverse*), comportant deux fonctions différentes qui semblent partager le même inverse. Elles doivent faire l'objet d'une analyse plus approfondie pour déterminer si la notion de « même » s'applique réellement. Il s'agit de la fonction causatif Caus avec son Event (événement qui est causé), versus la fonction très similaire Perm avec son Event.
- (e) FL inverses hiérarchiques
- i. Gener et Specific, Grimes en présente des exemples en anglais :

Gener(HAMMER) = HAND TOOL, Gener(PLIER) = HAND TOOL, Gener(WRENCH) = HAND TOOL, alors que Specific(HAND TOOL) = HAMMER, WRENCH, PLIER. C'est le même cas pour les paires de FL comme Whole et Part, Sequence et Phase qui sont introduites par Grimes.⁴

2. FL de détail

- (a) FL de détail actancielles
- i. S_i
 - ii. A_i
 - iii. Able_i
 - iv. Qual_i
 - v. Adv_i
- (b) FL Périphériques
- i. LOC_{ab} , LOC_{ad} et LOC_{in}
 - ii. S_{loc} , S_{instr} , S_{mod} , S_{med} et S_{res}

4. Grimes (1990) estime que la proposition de Mel'čuk de restreindre la fonction Gener aux seules relations lexicales visait à éviter que le lexique ait à gérer un trop grand nombre de relations générique. Or, selon lui, cette limitation est injustifiée : la fonction Gener devrait au contraire s'étendre à toutes les relations de type générique, y compris celles de nature encyclopédique, même si leur nombre est considérable.

(c) FL Aspectuelle

i. INTERNE : celles qui se situent à l'intérieur d'une période de temps définie.

A. Incep

B. Cont

C. Centr

D. Degrad

E. Fin

ii. EXTERNE : celles qui concernent les éléments qui précèdent immédiatement l'événement désigné par le mot-clé ou qui en découlent directement.

A. Prep (Prepar de Mel'čuk)

B. Prox

C. Antecedent et Consequent (Result_i de Mel'čuk)⁵

(d) FL de Degré

i. Superlatif (Magn de Mel'čuk)

ii. Bon

iii. Ver

iv. Increase et Decrease (Plus et Minus de Mel'čuk)

(e) FL Factives : (des FL de collocations verbales)

i. Func_i Oper_i et Labor_{ij}

ii. Fact_i Real_i et Labreal_{ij}

(f) FL diverses

i. Figur

ii. Involv

iii. Son

L'étude de Grimes se distingue par une approche originale, qui remet en question la distinction traditionnelle entre fonctions paradigmatiques et syntagmatiques. Bien que certaines parmi les FL mentionnées ne figurent pas dans la liste des FL Standards, son objectif global est de montrer comment les FL sont inter-connectées au niveau du sens. La classification des fonctions inverses repose sur une logique systématique consiste à regrouper les relation

5. Grimes argumente que c'est une autre facette de cette paire de FL qui se prolonge au-delà de la période de temps occupée par l'événement désigné par le mot-clé.

lexicales qui sont mutuellement inverse, telles que *Cap* vs *Equip*, *Mult* vs *Sing*. Cette approche pourrait être appliquée au sein du système des FL standards existant pour regrouper d'autres relations similaires, et cela a directement inspiré notre propre proposition de classification dans le cadre du présent projet.

2.3 Classification de Ramos et Tutin

Ramos et Tutin (1996) ont également apporté des contributions significatives aux recherches sur les FL. Elles ont réalisé une analyse constructive de la classification des FL. S'appuyant sur les travaux antérieurs (Žolkovskij et Mel'čuk, 1970, 1967; Mel'čuk *et al.*, 1995, etc.), elles ont choisi de conserver la classification fondamentale entre fonctions paradigmatiques et syntagmatiques, mais elles ont proposé des critères distinctifs qui reflètent leurs propres perspectives sur ces deux types de relations lexicales.

Selon elles, la valeur d'une FL paradigmatique et son mot-clé doivent partager une composante sémantique commune. De plus, ces deux éléments ne se combinent dans la même phrase que si leur cooccurrence est déterminée par leur sens respectif. Toutefois, pour une FL syntagmatique, la composante sémantique commune n'est pas nécessaire mais son mot-clé et sa valeur doivent être en cooccurrence conformément à certaines restrictions.

Voici la classification de Ramos et Tutin :

1. FL paradigmatiques

(a) Relations substitutives

- i. *Syn*
- ii. *Anti*⁶
- iii. *Gener_p*⁷
- iv. *Conv_{kij}*
- v. *Contr*

(b) Dérivations syntaxiques

- i. *S₀*
- ii. *V₀*
- iii. *A₀*

6. Dans cette sous-classe, une nouvelle fonction *Anti_{comp}* pour les complémentaires est introduite par Ramos et Tutin, par exemple : **Anti_{comp}**(ALIVE) = DEAD en anglais.

7. Ramos et Tutin ont introduit une autre fonction *Gener_p* pour les termes génériques qui ne se combinent jamais avec les mots-clés.

iv. Adv₀

(c) Dérivations actancielles

i. S_i

ii. A_i

iii. Adv_i

(d) Dérivations circonstancielles

i. S_{loc}

ii. S_{instr}

iii. S_{med}

iv. S_{mod}

v. S_{res}

(e) Qualificatifs pour les actants potentiels

i. Able_i

ii. Qual_i

(f) Commandes typiques

i. Imper

2. FL syntagmatiques

(a) Verbes supports

i. Func_i

ii. Oper_i

iii. Labor_{ij}

(b) Verbes de réalisation

i. Fact_i

ii. Real_i

iii. Labreal_{ij}

(c) Prédicateur

i. Pred

(d) Verbes de phase

i. Incep

ii. Cont

- iii. Fin
- (e) Verbes causatifs
 - i. Caus
 - ii. Liqu
 - iii. Perm
- (f) Verbes aspectuels
 - i. Prox
 - ii. Perf
 - iii. Result_i
- (g) Verbes introduisant une nouvelle situation
 - i. Involv
 - ii. Manif
 - iii. Sympt_{ijk}
- (h) Verbes au sens ‘négative’
 - i. Degrad
 - ii. Excess
 - iii. Obstr
 - iv. Nocer
- (i) Verbes “mélangés”
 - i. Prepar
 - ii. Son
- (j) Modificateurs adjectivaux et adverbiaux
 - i. Magn
 - ii. Bon
 - iii. Ver
 - iv. Pos_i
 - v. Epit
 - vi. Plus
 - vii. Minus
 - viii. Pejor

(k) Prépositions liées au mot-clé

- i. LOC_{in} , LOC_{ab} et LOC_{ad}
- ii. Instr
- iii. Propt

(l) Noms quasi-synonymes

- i. Gener
- ii. Figur

(m) Noms quasi-méronymes

- i. Sing
- ii. Mult
- iii. Cap
- iv. Equip
- v. Centr
- vi. Culm
- vii. Germ

2.4 Classification de Jousse

2.4.1 Contexte des classifications précédentes de Jousse

En général, Jousse (2010) présente la modélisation d'un système des relations lexicales basée sur les FL dans le cadre de la TST. Avant de proposer sa propre structuration, Jousse a présenté et critiqué les recherches précédentes sur le même sujet, et ces dernières dans lesquelles une forte divergence ont été observée même si elles sont fondées sur une base identique—la distinction entre les relations paradigmatiques et syntagmatiques. Elle déduit que c'est soit que les auteurs n'avaient pas perçus de la même manière les concepts de relations paradigmatiques et syntagmatiques, soit que les critères pour la réparation des FL n'avaient pas été respectés.

Jousse (2010) met en évidence un certain nombre de remarques dans les travaux précédents, qu'elle illustre par les exemples suivants : (i) Žolkovskij et Mel'čuk (1970, p. 10) ont classé les FL dans deux groupes—Substitutives (Syn , $Conv_{ij}$, $Anti$, A_0 , V_0 , Adv_0 , S_0) et Paramètres sémantiques (toutes les autres)—sans aucune sous-classe; (ii) Certains travaux

reprennent la distinction paradigmatiques/syntagmatiques, mais l’appliquent de manière discutable. Ce sont Steele et Meyer qui classent certaines fonctions typiquement syntagmatiques, comme *Magn* ou *Ver*, parmi les fonctions paradigmatiques ; (iii) Une classe du type « Verbes divers » devrait être évitée dans une classification rigoureuse, car elle sert à récupérer les FL qui n’ont pas trouvé de classes adéquates. C’est notamment le cas dans les travaux de Ramos et Tutin (1996) ou de Grimes (1990).

2.4.2 Les classes sémantiques

Vu toutes les recherches proposées ci-dessus, Jousse a tenté d’établir un nouveau système de FL en les classant selon quatre perspectives : classification sémantique, combinatoire, de partie du discours et communicationnelle. C’est la perspective sémantique qui nous intéresse le plus pour ce mémoire. Inspiré par le modèle de définition analytique pour définir les unités lexicales, Jousse a essayé de décomposer les FL en composantes sémantiques, et elle relève quatre cas de figures :

- FL sémantiquement vides, p. ex. : $S_0(\text{CRAINdre}) = \text{CRAINTE}$.
- FL à composante sémantique unique, p. ex. : *Magn* encode uniquement un sens ‘intensité’.
- FL à composantes sémantiques multiples, p. ex. : *AntiBon* encode ‘jugement’ et ‘négativité’.
- Plusieurs FL pour une même composante sémantique, p. ex. : ‘utilisation’ pour Fact_i , Real_i et Labreal_{ij} .

La classification proposée par Jousse repose sur les composantes sémantiques : elle vise à regrouper les FL qui partagent des composantes communes, de façon à permettre une double entrée. D’une part, il s’agit de pouvoir accéder à un ensemble aussi large que possible de FL à partir d’une composante sémantique donnée ; d’autre part, lorsqu’une FL possède plusieurs composantes, elle devient accessible à partir de chacune d’elles. Le travail n’est pas néanmoins de dresser un relevé exhaustif ou un ensemble des composantes de FL, mais de les identifier de façon souple et explicite. Pour ce faire, les gloses de vulgarisation pour les FL dans les travaux du *dictionnaire de combinatoire* (DiCo) (Polguère, 2000) et du *Lexique actif du français* (LAF) (Mel’čuk et Polguère, 2007) ont été empruntés, et les composantes similaires sont regroupées dans une même rubrique ou classe sémantique. Dix classes sémantiques ont été créées dans sa thèse ; en voici une énumération incomplète :

1. ÉQUIVALENCE
 - (a) Lexies Similaires : *Syn*, (*QSyn*, *Cf*, =)

- (b) **Conversif** : $Conv_{ij}$
 - (c) **Sens vide** : A_0, V_0, Adv_0, S_0
2. OPPOSITION
- (a) **Anti, Non, Contr**
 - (b) $Conv_{ij}$
 - (c) **Fem, Masc**
3. ÉLÉMENT/ENSEMBLE
- (a) **Générique/hyponyme** : $Gener, Hypo$, 'Qui sert pour toutes les portes d'un bâtiment' (CLÉI.1) = PASSE, PASSE-PARTOUT⁸
 - (b) **Parties constitutives** : $Mero, Hollo, Compos$
 - (c) **Individu/animaux** : 'Petit de'
 - (d) **élément/ensemble** : $Sing, Mult, Cap, Equip$
4. PARTICIPANTS
- (a) **Liées aux actants** : $S_i, A_i, Able_i, Qual_i$
 - (b) $Cap, Equip$
5. QUALIFICATIFS
- (a) **Jugement** : $Ver, AntiVer, Bon, AntiBon$
 - (b) **Intensité** : $Magn, AntiMagn, CausPredPlus, CausPredMinus, Culm$
 - (c) **Forme** : 'Qui descend jusqu'aux mollets de X' (JUPE) = À MI-MOLLET
6. PHASE/ASPECT
- (a) $Prepar, Incep, Centr, Fin$
 - (b) $S_{res}, De\ nouveauOper_1(AMI\ 1.1) = REDEVENIR [ART \sim]$
 - (c) $Magn^{temp}$
7. MANIÈRE
- (a) $S_{mod}, S_{med}, S_{instr}$
 - (b) $Instr, Adv_i, Adv_iReal_1, etc.$
8. CAUSE
- (a) $Propt$

8. Jousse (2010) a aussi notamment exploré des régularités sémantiques parmi un ensemble de relations lexicales représentées par des FL non standard.

(b) 'Dont la raison est l'amour' (CRIME_{1b}) = PASSIONNEL

9. LOCALISATION

(a) S_{loc}

(b) PreparReal₁, Culm, Loc_{in}, etc

10. ACTION/ÉVÉNEMENT

(a) Création : CausFunc₀, etc

(b) Manifestation : Symp_{t_{ijk}}, Manif, etc

(c) Réalisation/fonctionnement : Real, Fact, Involv, Son

(d) Augmentation : CausPredPlus

(e) Diminution : CausPredMinus

(f) Destruction/disparition : LiquOper₁, FinFunc₀

(g) Tentative : Essayer de Oper₁₂(VICTOIRE₁) = DISPUTER [art ~ à N=Y]

2.4.3 Remarques sur cette classification

Dans le cadre de ce mémoire, nous avons tenté d'établir une structure des FL en suivant des approches classiques, telles que celle de Mel'čuk et Polguère (2021), qui proposent des distributions hiérarchiques des FL en distinguant les relations paradigmatiques et syntagmatiques. Cependant, il reste encore des FL difficiles à regrouper avec les autres. Ce travail de Jousse a offert un bon modèle pour mieux organiser les FL. Par exemple, la FL *Prepar* pourrait être classée dans le groupe des verbes de phase, comme son analyse le suggère, et les FL fondamentales (*Syn*, *Anti*, *S₀*, etc.) sont classées de manière plus logique et intuitive. Cependant, les avantages de cette classification se limitent au rôle des composantes sémantiques, tandis que sa structure interne de classification n'est pas entièrement applicable à notre travail.

Dans le cadre d'une évaluation automatique de la compétence lexicale des LLM, il est important de disposer d'un ensemble de relations lexicales à la fois fréquentes et cohérentes. À ce titre, les FL standards et non standards devraient être classées selon des critères distincts et bien définis. Or, le travail de Jousse propose une classification unifiée, englobant aussi bien des FL standards (simples ou complexes) que non standards. Cette approche ne reflète pas adéquatement les distinctions importantes entre ces catégories. Deuxièmement, les relations paradigmatiques et syntagmatiques constituent deux types fondamentaux de relations lexicales, et il est crucial de maintenir cette distinction lors de la construction d'un système

de FL, comme cela est établi dans les approches classiques. Troisièmement, selon l'analyse de Jousse, certaines FL, en raison de leurs multiples composantes sémantiques, apparaissent inévitablement dans plusieurs classes. Cependant, cela peut entraîner des confusions lors de l'évaluation des relations entre les différentes FL, ce qui contredit l'objectif initial du système des FL qui vise à simplifier l'apprentissage des langues. Enfin, bien que Jousse ait évité d'utiliser des catégories floues comme VERBES DIVERS dans son propre travail, il reste difficile de contourner le fait que sa classe ACTION/ÉVÉNEMENT englobe encore un grand nombre de FL verbales, ce qui révèle ainsi la nécessité de créer des sous-catégories plus spécifiques.

2.5 Classification de Lambrey

2.5.1 Problématique et démarche

Lambrey (2016) s'intéresse aux collocations en génération automatique de texte (GAT) et elle remarque que celles-ci n'ont pas été traitées de manière exhaustive en réalisation linguistique. Elle montre qu'il est difficile de modéliser leur fonctionnement en englobant tous leurs comportements dans un programme informatique. D'autre part, la création et l'organisation de ressources linguistiques en vue de la réalisation de textes est une tâche fastidieuse et coûteuse, mais nécessaire lorsque l'on souhaite traiter des phénomènes linguistiques précis. Dans ce contexte, Lambrey a réalisé une intégration exhaustive des collocations dans un réalisateur profond de texte multilingue.

2.5.2 Classification proposée pour les FL syntagmatiques

Elle propose à cette fin une classification en fonction des patrons syntaxiques suivants :

1. Modification (inversant la relation de dominance de RSém vers RSyntP) : ces fonctions établissent une relation attributive entre la base et le collocatif au sein de la RSyntP. Le collocatif joue le rôle de modificateur adjectival ou adverbial, vis-à-vis de la base. Ces FL sont donc classées dans un seul schéma, nommé Modification. Chacune est représentée par une relation attributive de dominance inversée entre la base et le collocatif dans la RSyntP de la phrase, comme illustré à la figure 2.2.
 - (a) Magn
 - (b) Bon
 - (c) Ver

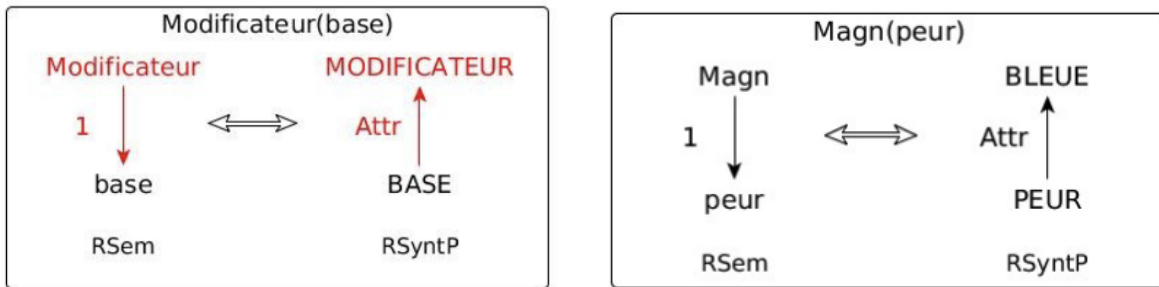


FIGURE 2.2 – Fonctionnement du patron Modification, (Lambrey, 2016, p. 81)

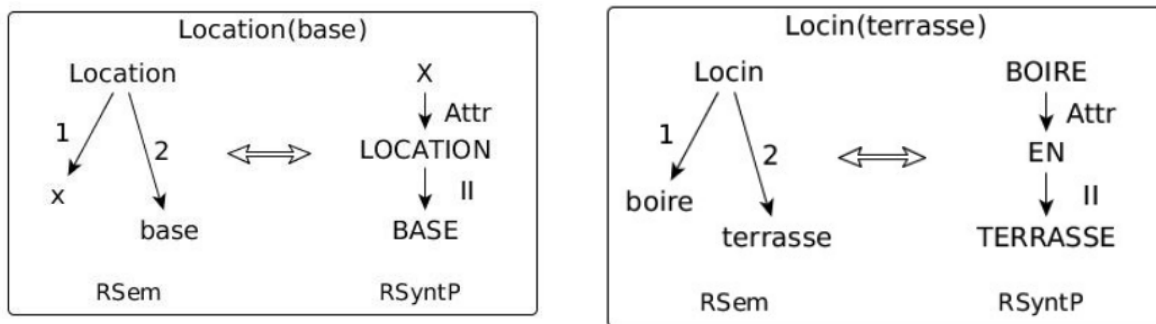


FIGURE 2.3 – Fonctionnement du patron Préposition, (Lambrey, 2016, p. 84)

(d) Pos

2. Préposition : Les FL qui sont prédicatives et possèdent deux actants en RSém, dont l'actant 1 concerne une action ou une personne, tandis que l'actant 2 représente la base de la collocation et est donc l'entité reliée à la base. La Figure 2.3 montre un exemple de patron syntaxique pour les prépositions exprimant le sens de 'localisation', cela peut aussi s'appliquer à d'autres sens associé aux fonctions Instr ou Propt.

(a) LOC_{in}

(b) LOC_{ad}

(c) LOC_{ab}

(d) Instr

(e) Propt

3. Nom gouverneur sémantique regroupe des fonctions nominales qui divergent sémantiquement, mais leur représentation en RSém et leur correspondance en RSyntP sont identiques. Dans chaque cas, la RSém intègre un nœud prédicatif ayant la base comme actant 1. Ces FL introduisent ensuite le collocatif nominal en RSyntP, ce dernier régissant la lexicalisation de la base avec la relation I (complément du nom en syntaxe

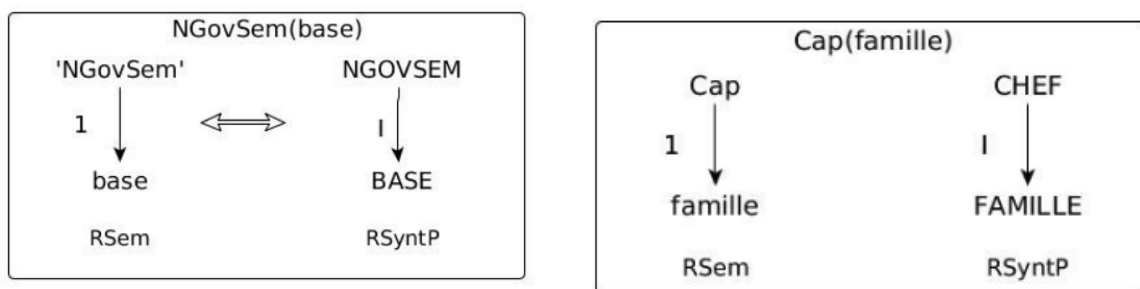


FIGURE 2.4 – Fonctionnement du patron NGovSem, (Lambrey, 2016, p. 85)

de surface). Ce groupe de FL est illustré dans le patron ci-dessous 2.4.

- (a) Culm
 - (b) Sing
 - (c) Mult
 - (d) Germ
 - (e) Centr
 - (f) Cap
 - (g) Equip
4. Verbes supports : les verbes supports sont des fonctions sémantiquement vides qui représentent les collocations dont le collocatif verbal est sélectionné en fonction de la base L. La figure 2.5 montre le patron sémantico-syntaxique en prenant $Oper_i$ comme exemple. En RSém, le prédicat ou quasi-prédicat ' x ' et son i -ème actant ' a ' sont régi par la relation i . Cette base est lexicalisé comme X occupant la position de Π -ème (I pour $Func_i$, III pour $Labor_{ij}$) actant du verbe support (Vsupp) en RSyntP.
- (a) $Func_i$
 - (b) $Oper_i$
 - (c) $Labor_{ij}$
5. Verbes de réalisation
- (a) $Fact_i$
 - (b) $Real_i$
 - (c) $Labreal_{ij}$

Selon Lambrey, le patron RSém–RSyntP dans la figure 2.5 peut également s'appliquer à représenter d'autres FL de collocations verbales comme les verbe de réalisation (Vreal). En effet, dans tous ces cas, le prédicat sémantique figurant dans la

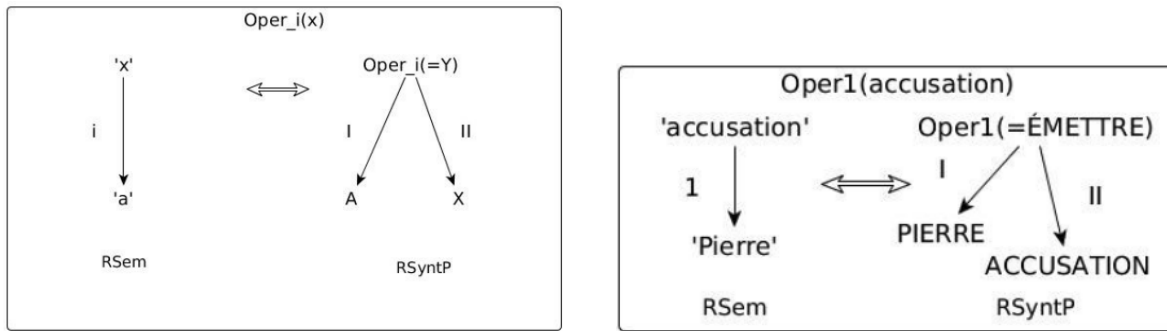


FIGURE 2.5 – Fonctionnement général des indices pour les Verbes supports, (Lambrey, 2016, p. 88)

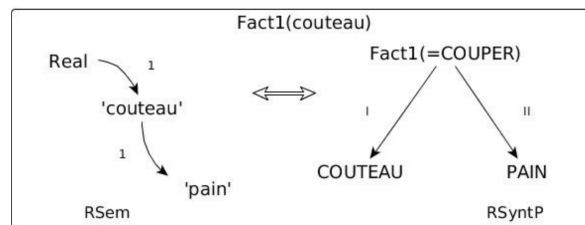


FIGURE 2.6 – Fonctionnement général des indices pour les Verbes de réalisation, (Lambrey, 2016, p. 91)

RSém devient un actant syntaxique du verbe dans la RSyntP, tandis que les arguments sémantiques de ce prédicat peuvent être réalisés comme d'autres actants du verbe. Comme le montre la figure 2.6, ce patron reste valide même lorsque la sémantique du verbe change. Par exemple, si l'on remplace ' x ' par une valeur sémantique concrète comme 'couteau', tout en y ajoutant une composante de sens telle que 'Real', la structure de la RSyntP à droite reste conforme au schéma décrit précédemment.

6. Autres verbes sémantiquement pleins

- (a) Manif
- (b) Obstr
- (c) Excess
- (d) Son
- (e) Symp_{ijk}
- (f) Degrad
- (g) Stop

7. Combinaisons de FL Verbales

- (a) Verbes de Phase, Prepar, Prox (qui pourraient se combiner avec d'autres FL Verbales)
 - i. Incep
 - ii. Cont
 - iii. Fin
 - iv. Prox
 - v. Prepar⁹
- (b) Verbes causatifs
 - i. Caus
 - ii. Liqu
 - iii. Perm

8. Dérivation Adjectivale

- (a) A_i
- (b) A_i avec les FL Verbales

9. Les FL non classées

- (a) Epit
- (b) Figur
- (c) Involv

2.6 Classification du présent mémoire

2.6.1 Classifier les FL pour le projet

2.6.1.1 Limites des classifications existantes

À la lumière des observations ci-dessus, nous constatons que les classifications existantes des FL demeurent insuffisamment structurées et manquent de granularité. Par exemple, $S_0(\text{PRODUIRE})$ renvoie à l'activité abstraite désignée par le verbe lui-même, produisant ainsi la forme nominale PRODUCTION , et représente donc une dérivation sans ajout de sens. En revanche, $S_1(\text{PRODUIRE})$ donne PRODUCTEUR_N , qui désigne l'agent de l'activité—c'est-à-dire

9. Peut être regroupé ici, non pas en raison de la propriété combinatoire partagée avec les verbes de phase, mais en raison de sa proximité sémantique, car Mel'čuk a modifié l'application de *Prepar*.

le premier argument du prédicat ‘produire’. De manière similaire, $S_2(\text{PRODUIRE})$ produit **PRODUIT**, qui réfère au résultat de l’activité—soit le second argument du prédicat.

Cela illustre deux niveaux de distinction sémantique : bien que S_1 et S_2 soient toutes deux des dérivations orientées vers les arguments, et donc sémantiquement proches, elles diffèrent quant au rôle actancielle qu’elles représentent. S_0 , en revanche, est plus distincte, puisqu’elle encode l’événement lui-même sans aucune augmentation sémantique. La diversité des distinctions sémantiques observées entre FL souligne la nécessité d’un modèle de classification à plusieurs niveaux de granularité. Dans cette recherche, nous nous attachons à évaluer dans quelle mesure les LLM sont sensibles à ces distinctions fines, ce qui requiert un schéma de classification explicite pour modéliser systématiquement les FL.

2.6.1.2 Objectifs de classification dans notre approche

Notre approche consiste à cibler des connaissances lexicales spécifiques tout en nous appuyant sur les annotations pertinentes du RL-fr pour générer automatiquement des exemples de questions et de réponses mobilisant ces connaissances. En premier lieu, il nous faut déterminer et prioriser les connaissances des relations lexicales à évaluer. Nous avons donc besoin d’une classification des FL bien structurée qui nous permette de cibler une classe spécifique tout en insérant des réponses incorrectes provenant d’autres FL de la même classe ou d’une classe proche.

Un modèle de FL structuré doit refléter leurs caractéristiques sémantiques et syntaxiques, car ce sont les liens sous-jacents entre ces caractéristiques qui constituent les aspects essentielles à évaluer. À cet effet, nous restons davantage enclin à suivre la dichotomie de classes paradigmatique et syntagmatique, qui représentent deux grandes catégories principales de relations lexicales. Comme Ramos et Tutin (1996) le soulignent, pour les FL paradigmatiques, les valeurs et les mots-clés partagent nécessairement des composantes sémantiques communes. En effet, les deux éléments ne forment généralement pas un syntagme, alors que certaines valeurs ne peuvent se combiner avec le mot-clé que dans des relations de cooccurrence lexicale libre, c’est-à-dire des cooccurrences déterminées sémantiquement par les significations respectives du mot-clé et de la valeur. En revanche, dans le cas des FL syntagmatiques, la valeur et le mot-clé ne partagent pas forcément de composantes sémantiques communes, mais ils sont combinés dans un syntagme pour exprimer un sens précis, dans des conditions de cooccurrence lexicale restreinte.

Notre approche est de d’abord classifier les FL selon les PdD des paires de lexies qu’elles

relient. Chaque classe est subdivisée en fonction des caractéristiques sémantiques des relations qu'elles représentent, puis affinée en tenant compte des patrons syntaxiques et des différences au niveau actanciel. Nous proposons donc la classification suivante :

2.6.1.3 Une classification hiérarchique des FL

FL paradigmatiques

1. Substitutive : classe inspirée de la classification de Ramos et Tutin. C'est une classe de FL dont le mot-clé et la valeur appartiennent à la même catégorie syntaxique, et sont syntaxiquement interchangeables.

(a) Sens similaire : nous regroupons ici ces deux FL qui désignent l'équivalence sémantique entre la valeur et le mot-clé.

i. *Syn* retourne le synonyme de L ;

ii. *Gener* désigne l'hyponyme de L sans modifier le sens. La valeur de cette fonction est sémantiquement plus général que le mot-clé, ce qui rend difficile de distinguer une divergence entre eux (ex : ALLER et son *Gener* DÉPLACER peuvent exprimer le même sens), il serait donc pertinent de regrouper cette fonction ici ;

(b) Sens opposé :

i. *Anti* renvoie l'antonyme sémantiquement opposé à L ;

ii. *Contr* est une opposition qui n'est pas nécessairement strictement sémantique, mais pourrait être un contraste plutôt conceptuel ou paralinguistique. Elle se confond avec *Anti* dans certains cas, par exemple :

$Contr(SORTIE) = Anti(SORTIE) = ENTRÉE$

$Contr(DÉFENDRE) = ATTAQUER$

$Anti(DÉFENDRE) = CAPITULER, ABANDONNER.$

iii. $Conv_{ij}$ représente en fait une équivalence sémantique, mais elle est souvent associée à *Anti* par une opposition de position des actants, même si elle n'ajoute aucune sémantème. Il est donc pertinent d'évaluer les modèles de langue à les distinguer, par exemple :

$Anti(SE SOUVENIR) = OUBLIER$

$Conv_{21}(SE SOUVENIR) = REVENIR$

REMARQUE : $Conv_{ij}$ peut également être intégrée à la classe des FL *Substitives à sens similaire*, dans la mesure où le mot-clé et la valeur partagent

un sens globalement équivalent, à l'exception d'une inversion dans l'assignation des arguments sémantiques. Cependant, cette fonction peut prêter à confusion avec des FL de type *Anti*, notamment d'un point de vue intuitif, car la transformation opérée—bien qu'impliquant une permutation argumentative plutôt qu'une opposition lexicale—peut être perçue comme une forme de « contraire ». Par exemple, *VICTOIRE* et *DÉFAITE* constituent la FL *Conv₂₁* au lieu de *Anti*. C'est pourquoi nous classons cette FL parmi les sens opposés.

2. Dérivation nominale : cette classe regroupe les FL qui retournent la valeur nominale. Elle peuvent être sémantiquement vides ou non.

(a) Sens vide : S_0 , qui nominalise *L*. Il s'agit d'une dérivation purement syntaxique, sans aucune modification du sens ou de la position des actants ;

(b) Ajout de sens :

i. Dérivation nominale actancielle : S_i , qui désigne la dérivation nominale actancielle sans aucun sémantème ajouté, mais elle pourrait être perçue comme si le sens était modifié, par exemple *VENDEUR*, soit $S_1(\text{VENDRE})$, n'est pas ressenti comme le même sens que *VENTE*, qui est $S_0(\text{VENDRE})$, car il existe en réalité une modification de dominance communicative dans leur définition¹⁰. Cette explication s'applique également aux dérivations adjectivales et adverbiales ci-dessous.

ii. Dérivation nominale circonstancielle : dans cette sous classe il est regroupé le FL qui retourne le nom des circonstants typiques du mot-clé, soit :

A. S_{instr} désigne l'instrument typique de *L*, comme $S_{\text{instr}}(\text{BOIRE}) = \text{VERRE}$.

B. S_{med} désigne la méthode typique de *L*, comme $S_{\text{med}}(\text{COLLER}) = \text{COLLE}$.

C. S_{mod} désigne la mode ou le moyen typique de *L*, comme $S_{\text{mod}}(\text{VIVRE}) = \text{「MODE DE VIE」}$.

D. S_{loc} désigne le lieu typique de *L*, comme $S_{\text{loc}}(\text{FUMER}) = \text{FUMOIR}$.

E. S_{res} désigne le résultat typique de *L*, comme $S_{\text{res}}(\text{NUIRE}) = \text{DOMMAGE}$.

3. Dérivation adjectivale : cette classe de FL désigne les adjectifs dérivés à partir du mots-clé, soit sémantiquement vide ou avec ajout de sens.

(a) Sens vide : A_0 , qui adjectivise *L*, comme $A_0(\text{COMPARER}) = \text{COMPARATIF}$.

10. La définition de *VENTE* est : 'l'action par laquelle la personne *x* vend la chose *y* à *z* pour un prix *w*'. En revanche, *VENDEUR* se définit comme 'la personne *x* qui vend *y* à ...'. Dans ces deux cas, la dominance communicative entre l'actant *x* et le prédicat 'vendre' est différente, dans la définition de *VENTE*, 'vendre' domine communicativement le *x*, alors que pour *VENDEUR*, la dominance est à l'envers

(b) Ajout de sens :

i. A_i : comme les S_i ci-dessus, cette FL caractérise l'actant i du mot-clé prédicatif en inversant la dominance communicative, comme $A_1(\text{DOUTER}) = \text{DUBITATIF}$.

ii. Qualificatifs de L :

A. $Able_i$, qui caractérise l'actant i du sens 'peut être L', comme $Able_2(\text{LIRE}) = \text{LISIBLE}$.

B. $Qual_i$, qui caractérise l'actant i du sens 'a tendance d'être L', comme $Qual_2(\text{TROMPER}) = \text{NAIF}$.

4. Dérivation adverbiale : deux FL sont regroupée ici ; l'une est vide de sens et l'autre modifie la dominance communicative

(a) Dérivation syntaxique : Adv_0 , comme $Adv_0(\text{VOIR}) = \text{VISUELLEMENT}$.

(b) Dérivation actancielle : Adv_i , comme $Adv_1(\text{ADMIRER}) = \text{ADMIRATIVEMENT}$.

REMARQUE : D'après Mel'čuk et Polguère (2021, p. 45), la fonction Adv_i avait été historiquement classée parmi les syntagmatiques (de façon erronée d'après eux), du fait que la valeur est souvent introduite par une préposition régissant un nom L, ce qui pourrait être considéré comme une collocation. Par exemple :

$Adv_1(\text{ÉLÉGANCE}) = \text{AVEC} \sim$

Il ne faut pas confondre avec le FL syntagmatiques prépositionnelles comme $Instr$ et $Propt$.

5. Dérivation verbale : cette classe regroupe les FL paradigmatiques verbales.

(a) Sens vide

i. V_0 , qui est une dérivation syntaxique, comme $V_0(\text{PRÉSENTATION}) = \text{PRÉSENTER}$.

ii. $Pred$ qui transforme L en prédicat syntaxique ¹¹, comme $Pred(\text{ALCOOLIQUE}) = \text{BOIRE}$.

(b) Ajout de sens :

i. $Result_i$ ajoute le sens 'résulter nécessairement', comme $Result_1(\text{ACQUÉRIR}) = \text{POSSÉDER}$.

ii. $Perf$ ajoute le sens 'perfectif' ou 'aboutissement', comme $Perf(\text{AGONISER}) = \text{MOURIR}$.

11. $Pred$ fonctionne de façon « autonome » dans ce cas, bien qu'elle soit avant tout destinée à construire des FL complexes—cf. $IncepPredPlus$.

6. Dérivation clausale :

- (a) *Claus* est la clausativisation sans ajout de sens, comme *Claus*(FÉLICITER) = Félicitations !
- (b) *Imper* est une clausativisation avec ajout du sens ‘impératif’, comme *Imper*(VENIR) = Ici !

FL syntagmatiques Ce grand groupe de FL est classifié en suivant globalement l’approche de Lambrey, qui regroupe d’abord les FL en fonction de leur patron syntaxique, puis par sens.

1. **Modificateur** : cette classe correspond principalement à celle proposée par Lambrey (cf. §2.5). Les FL de cette classe ont le même patron syntaxique qu’à la figure 2.2.

(a) **Modificateur vide de sens** : deux FL de modification qui n’expriment aucun ajout de sens :

- i. *Epit* retourne l’épithète pléonastique de L, comme *Epit*(DEVISER) = GAIE-MENT.
- ii. *Redun* retourne l’épithète redondante de L, dans le but d’en préciser l’acception, comme *Redun*(APPEL) = TÉLÉPHONIQUE.

(b) **Modificateur ajoutant du sens** : des FL qui modifient sémantiquement L ; ces trois FL expriment un sens positif¹².

- i. *Magn* dénote l’intensification de L, comme *Magn*(PEUR) = BLEUE.
- ii. *Ver* dénote le fait que L est tel qu’il devrait être, comme *Ver*(SOURIRE) = FRANC.
- iii. *Bon* dénote le jugement positif du locuteur par rapport à L, comme *Bon*(CHOIX) = HEUREUX.

2. Collocation_prépositionnelle

(a) *LOC_{ab}*/*LOC_{ad}*/*LOC_{in}* désignent respectivement les prépositions introduisant L comme le lieu de positionnement, de provenance et de destination, comme *LOC_{in}*(FRANCE) = EN [~].

(b) *Instr* désigne les prépositions introduisant L comme l’instrument, comme *Instr*(AVION) = PAR [~].

12. La raison pour laquelle nous spécifions « positif » ici est que, dans RL-fr, les opposés de ces trois FL dans le sens négatif (*AntiMagn*/*AntiVer*/*AntiBon*) sont également considérés comme FL standards simples. De plus, étant donné leur occurrence assez importante dans le jeu de données, nous pourrions aussi insérer une sous-classe intermédiaire pour classer ces deux genre de modificateurs opposés, ce qui faciliterait une évaluation plus approfondie.

- (c) *Propt* désigne les prépositions introduisant L comme la cause, comme *Propt*(INTÉRÊT) = PAR [~].
3. Nom gouverneur : cette classe englobe les FL qui désignent un gouverneur syntaxique nominal d'un mot-clé nominal¹³. De plus, nous proposons d'inclure la fonction *Figur* dans cette classe car sa valeur est également un gouverneur syntaxique de L, même si elle n'ajoute pas de sens réel ; nous nous éloignons ici de Lambrey.
- (a) Sens vide : *Figur* n'ajoute pas de sens mais retourne un gouverneur nominal métaphorique ou stylistique.
- (b) Ajout de sens :
- i. Nombre du nom :
 - A. *Cap* et *Sing* ont le sens 'individu' ou 'unité', comme *Cap*(ROYAUME) = ROI, *Sing*(RIZ) = GRAIN.
 - B. *Equip* et *Mult* ont le sens 'groupe', comme *Equip*(NAVIRE) = ÉQUIPAGE, *Mult*(CHIEN) = MEUTE de [~s]
 - ii. État du nom :
 - A. *Culm* désigne le « point culminant / central » de L, comme *Germ*(FLEUR) = BOUTON [de (ART) ~]
 - B. *Germ* désigne l'état initial de L, comme *Culm*(JOIE) = COMBLE [de ART ~].
4. Collocation verbale :
- (a) Sens vide :
- i. *Copul* est une copule signifiant 'être', qui prend L comme complément en le supportant syntaxiquement, comme *Copul*(EXEMPLE) = ÊTRE, SERVIR [d'~]
 - ii. Les verbes supports servent aussi à supporter syntaxiquement leur mot-clé pour construire une structure syntaxique valide, sans ajout de sens.¹⁴

13. Pour Mel'čuk et Polguère (2021), l'application de *Sing* et de *Mult* ne se limite pas aux noms, mais aussi à certains verbes :

Sing(BATTRE) = FRAPPER

AntiMagn+Mult(SAUTER) = SAUTILLER

Cependant, d'après le jeu de données du RL-fr, aucune instance verbale n'est présente pour cette application ; nous nous concentrons donc ici uniquement sur les mots-clés nominaux.

14. Selon Mel'čuk et Polguère (2021), la différence entre la fonction *Copul* et les verbes supports réside dans le fait que *Copul* est équivalent au sens 'être' et son mot-clé peut-être un nom/adjectif/adverbe, tandis que les verbes supports prennent souvent une lexie nominale comme mot-clé. Plus précisément, *Copul* constitue le pendant de *Oper₁* pour les adjectifs et les noms quasi-prédicatifs assimilables (tels que MALADE_N, SURDOUÉE_N, etc.).

- A. Func_i , comme $\text{Func}_0(\text{TONNERRE}) = \text{GRONDER}$.
- B. Oper_i , comme $\text{Oper}_1(\text{MÉFAIT}) = \text{PERPÉTRER}$.
- C. Labor_{ij} , comme $\text{Labor}_{12}(\text{HÉRITAGE}) = \text{OBTENIR, RECEVOIR}$.

(b) Sens plein :

i. Ajout du sens ‘fonctionner’ ou ‘utiliser’

A. Fonctionnement normal (verbes de réalisation)

- Real_i , comme $\text{Real}_1(\text{PROMESSE}) = \text{TENIR}$.
- Labreal_{ij} , comme $\text{Labreal}_{12}(\text{micro-ondes}) = \text{PASSER}$.
- Fact_i , comme $\text{Fact}_0(\text{RÊVE}) = \text{SE RÉALISER}$.

B. Fonctionnement anormal

- Obstr ‘fonctionner avec difficulté’, comme $\text{Obstr}(\text{MÉMOIRE}) = \text{FLANCHER}$.
- Stop ‘arrêter de façon anormale’, comme $\text{Stop}(\text{MATCH}) = \text{S’INTERROMPRE}$.
- Excess ‘fonctionner de façon excessive’, $\text{Excess}(\text{PENSÉES}) = \text{SE BOUSCULER}$.
- Degrad ‘se dégrader’¹⁵, comme $\text{Degrad}(\text{LAIT}) = \text{TOURNER}$.

ii. Verbes de phase

- A. Incep ‘commencer’, comme $\text{Incep}(\text{DORMIR}) = \text{S’ENDORMIR}$.
- B. Cont ‘continuer’, comme $\text{ContOper}_1(\text{SILENCE}) = \text{GARDER}$.
- C. Fin ‘finir’, comme $\text{Fin}(\text{DORMIR}) = \text{SE RÉVEILLER}$.
- D. Prepar ‘préparer’, comme $\text{Prepar}(\text{COUTEAU}) = \text{AIGUISER}$.

REMARQUE : Les FL listées ci-dessus peuvent fonctionner de façon autonome, tout en étant fréquemment combinées avec d’autres FL verbales pour former des FL complexes. Dans ce dernier cas, nous incluons également Prox ‘sur le point de’, qui indique aussi une phase ; elle s’associe souvent avec les Vsupp :

$$\text{ProxFunc}_0(\text{TEMPÊTE}) = \text{MENACER}.$$

iii. Verbes causatifs :

- A. Caus ‘causer’, comme $\text{CausFunc}_0(\text{LÉSION}) = \text{ENTR’AÑER}$.

15. D’après Mel’čuk et Polguère, Degrad est le raccourci de la FL complexe IncepPredAntiBon , elle est regroupée ici du fait de sa signification négative comme Obstr , Stop et Excess .

B. Perm ‘ne pas faire cesser’, comme PermOper₁(abandon) = LAISSER.

C. Liqu ‘faire cesser’, comme Liqu(DORMIR) = ÉVEILLER.

REMARQUE : Tout comme les verbes de phase, bien que ces FL causatives puissent être utilisées seules, elles sont davantage employées dans des FL complexes (CausPredPlus, PermFact₀, etc.) dans le RL-fr.

iv. Autres verbes sémantiquement pleins :

A. Involv signifie ‘agir sur qqch’, comme Involv(FOULE) = DÉFERLER.

B. Manif signifie ‘se manifester dans qqch’, comme Manif(FATIGUE) = SE LIRE.

C. Son signifie ‘produire un son typique’, comme Son(FLAMME) = CRÉPITER.

2.7 Synthèse des classifications susmentionnées

Les classifications présentées dans les recherches antérieures offrent divers points de vue aux FL et encouragent une réflexion plus approfondie sur la structuration. Mel’čuk et ses collègues (Mel’čuk *et al.*, 1995; Ramos et Tutin, 1996; Mel’čuk et Polguère, 2021, etc.) se sont principalement focalisés sur l’analyse des relations lexicales représentées par les FL, en s’appuyant sur la dichotomie paradigmatiques et syntagmatiques. Cependant, Jousse et Grimes ont pris le parti de rejeter cette classification classique, chacun développant sa propre approche dans des directions distinctes. De son côté, Grimes a examiné les FL sous un angle fonctionnel, en explorant la possibilité de transformation entre elles. Selon lui, certaines FL peuvent être converties en d’autres en inversant simplement la position de la valeur et de l’argument, tandis que d’autres FL ne le permettent pas. Jousse, quant à elle, a opté pour une approche strictement sémantique, classant les FL selon le type de relation sémantique à exprimer. Ce qui est intéressant, c’est que Lambrey (2016) adopte une approche opposée à celle de Jousse. Elle privilégie l’étude des manifestations des FL dans la représentation syntaxique de surface (RSyntS), regroupant les FL en fonction de leur réalisation syntaxique commune tout en prenant en compte leurs différences sémantiques. Cette démarche est dictée par ses objectifs spécifiques dans le cadre de la GAT, avec une attention particulière portée aux FL syntagmatiques qui forment les collocations.

D’un autre côté, en examinant la structure interne de leurs classifications, nous pouvons observer un nombre significatif de points communs bien que les cadres extérieurs diffèrent. Par exemple, les sous-classe de modificateurs et de certaines fonctions verbales tendent à

rester généralement similaires. Cela pourrait s'expliquer par le fait qu'elles partagent dans leur interne un haut degré de similitude dans les schémas sémantiques et syntaxiques.

Dans ce contexte, nous proposons une classification spécifiquement conçue pour notre projet d'évaluation. Nous suivons l'idée d'une distinction entre FL paradigmatiques et syntagmatiques, puis classons les FL en fonction des parties du discours représentées par les FL, et ces FL sont ensuite regroupées selon leurs propriétés distinctives sur les plans sémantique et syntaxique, ce qui nous permet de construire une hiérarchie à plusieurs niveaux. La structure finale, utilisée pour l'évaluation, est présentée sous forme graphique à la figure 3.2 au chapitre 3. Toutes ces classifications seront rassemblées avec notre propre classification dans le tableau 2.1 à titre de référence.

Meľčuk (2021)	Ramos & Tutin (1996)	Lambrey (2016)	Nous
SYNTAGMATIQUES			
Modif Magn Ver Bon Degrad Epit Redun Plus Minus Prép Loc _{ad/ab/in} Instr Propt Vsupp Func _i Oper _i Labor _{ij} Vreal Fact _i Real _i Labreal _{ij} Prepar V. de phase Incep Cont Fin Prox V. causatifs Caus Perm Liqu V. de fonctionnement Obstr Stop Excess V. de situation Son Manif In- volv Sympt _{ijk} État de N. Germ Culm Copule Copul	Modif Magn Ver Bon Epit Plus Minus Pos _i Pejor Prép Loc _{ad/ab/in} Instr Propt Vsupp Func _i Oper _i Labor _{ij} Vreal Fact _i Real _i Labreal _{ij} V. de phase Incep Cont Fin V. causatifs Caus Perm Liqu Prédicativiser Pred V. divers PreparSon V. négatifs Degrad Obstr Ex- cess Nocer V. d'aspect Prox Perf Result _i V. de situation Manif Involv Sympt _{ijk} Quasi-Mero Culm Sing Mult Germ Centr	Modif Magn Ver Bon Pos Prép Loc _{ad/ab/in} Instr Propt Vsupp Func _i Oper _i Labor _{ij} Vreal Fact _i Real _i Labreal _{ij} V. de phase Incep Cont Fin V. causatifs Caus Perm Liqu Autres V. Obstr Stop Excess Son Degrad Manif Sympt _{ijk} FL non-classées Epit Figur Involv N. gouvern. sém. Culm Sing Mult Germ Centr Cap Equip Dériv. A. A _i	Modif Plus Minus Epit Redun Magn Ver Bon Prép Loc _{ad/ab/in} Instr Propt N. gouvern. Figur Culm Germ Cap Sing Mult Equip Copule Copul Vsupp Func _i Oper _i Labor _{ij} Vreal Fact _i Real _i Labreal _{ij} V. de fonction anormale Obstr Excess Stop Degrad V. de phase Incep Cont Fin Prepar V. causatifs Caus Perm Liqu V. sens plein divers Involv Manif Son
PARADIGMATIQUES			
Trois bases Syn Conv _{ij} Anti Trois apparentés Gener Figur Contr Dériv. syntaxique S ₀ A ₀ Adv ₀ V ₀ Claus Prédicativiser Pred Dériv. N. S _i Equip Cap S _{instr} S _{mod} S _{med} S _{loc} S _{res} Dériv. A. A _i Able _i Qual _i Dériv. Adv Adv _i Dériv. variée Sing Mult Imper Imperf Result _i	Substitutives Syn Anti Conv _{ij} Gener _p Contr Quasi-synonymes Gener Figur Dériv. syntaxique S ₀ A ₀ V ₀ Adv ₀ Dériv. actants S _i A _i Adv _i Dériv. circonstants S _{instr} S _{mod} S _{med} S _{loc} S _{res} Qualif. actants Able _i Qual _i Commande typique Imper	Équivalent Syn Gener Opposé Anti Contr Conv _{ij} Dériv. N. S ₀ S _i S _{instr} S _{mod} S _{med} S _{loc} S _{res} Dériv. A. A ₀ A _i Able _i Qual _i Dériv. V. V ₀ Pred Perf Result _i Dériv. Adv. Adv ₀ Adv _i Dériv. Claus. Claus Imper	

TABLEAU 2.1 – Synthèse des classifications de FL

Grimes (1990)	Jousse (2010)
<u>FL de détail</u>	<u>CLASSES SÉMANTIQUES</u>
<p>Degré Magn Bon Ver Plus Minus Périphérique LOC_{ad/ab/in} Factive Func_i Oper_i Labor_{ij} Fact_i Real_i Labreal_{ij} Aspect interne Incepar Cont Centr Fin Degrad Aspect externe Prepar Prox Andecedent Consequence Actenciel S_i A_i Able_i Qual_i Adv_i Divers Figur Involv Son</p>	<p>Équivalence Syn Conv_{ij} S₀ V₀ A₀ Adv₀ Opposition Anti Non Contr Conv_{ij} (Fem vs Masc) Élément/Ensemble Gener Hypo Mero Hollo Compos Sing Mult Cap Equip Phase/Aspect Prepar Incepar Centr Fin S_{res} Magn^{temp} Qualificatif Ver AntiVer Bon AntiBon Magn Culm Plus Participants S_i A_i Able_i Qual_i Adv_i Cap Equip Cause Propt etc. Manière S_{med} S_{mod} S_{instr} Instr Adv Localisation S_{loc} Culm LOC_{ad/ab/in} Action/Événement CausFunc₀ Sympt_{ijk} Manif Real_i Fact_i Involv Son CausPredPlus CausPredMinus LiquOper₁ FinFunc₀</p>
<u>FL inversible</u>	
<p>Catégorie S₀ A₀ V₀ Adv₀ PredCopol Inverse simple Syn_c/Syn_s Liqu/State Sympt_{ijk}/Manif Cap/Equip Sing/Mult Inverse symétrique Syn Anti Conv_{ij} Inverse regroupé Caus/Event Perm/Event Inverse hiérarchique Gener/Specific Whole/Part Sequence/Phase</p>	

TABLEAU 2.2 – Synthèse des classifications de FL (suite)

Chapitre 3

Méthodologie expérimentale

3.1 Définition de la tâche et aperçu de la méthode

La présente étude vise à examiner dans quelle mesure les LLM sont capables de reconnaître différents types de relations lexicales. Plus précisément, nous cherchons à savoir si, lorsqu'ils sont sollicités par une requête (*prompt*) appropriée, les modèles peuvent juger si une paire de lexies française correspond à la fonction lexicale ciblée donnée.

Pour ce faire, nous adoptons une stratégie d'évaluation fondée sur les requêtes (*prompt-based*), s'inscrivant dans la lignée du paradigme des *instructions naturelles* (*Natural Instructions*; voir Mishra *et al.*, 2022; Wang *et al.*, 2022). Ce paradigme consiste à transformer une tâche linguistique en une consigne en langage naturel, afin que le modèle réponde à la question posée à partir de cette requête.

Afin de tester la capacité des LLM à distinguer des FL plus ou moins éloignées, nous adoptons une approche **contrastive**. Pour chaque **FL cible**, nous définissons un ensemble de **FL contrastives** partageant certaines propriétés formelles (par exemple, des constructions dérivationnelles similaires), mais différant par des traits sémantiques ou syntaxiques clés (comme un changement de sens plus marqué, ou un rôle actanciel différent). Ces contrastes nous permettent de contrôler le niveau de finesse requis pour le raisonnement du modèle.

Chaque requête utilisée dans notre protocole expérimental comporte plusieurs composantes. Cela commence par une **description** formelle de la FL cible, fondée sur une définition théorique issue de la littérature (accompagnée des références pertinentes), que nous enrichissons par une reformulation plus accessible afin d'en faciliter l'interprétation par les modèles. Elle inclut ensuite plusieurs exemples illustrant des paires de lexies conformes à cette FL—**exemples positifs**, ainsi que des exemples issus de FL contrastives, c'est-à-dire partageant

certaines propriétés avec la FL cible tout en s’en distinguant sur un autre plan—**exemples négatifs**. Enfin, la requête se termine par une **question** adressée au LLM portant sur une nouvelle paire de lexies, à laquelle il doit répondre en indiquant si cette paire instancie ou non la FL cible.

Cette stratégie expérimentale constitue la base de notre démarche. Pour la rendre possible, il nous faut d’abord disposer d’une classification hiérarchique structurée des fonctions lexicales, ce qui a été établi au chapitre 2. Le présent chapitre est quant à lui consacré à la présentation des ressources lexicales mobilisées, au traitement des données, à la préparation des questions polaires, à la génération des requêtes, ainsi qu’à l’organisation concrète des expériences.

3.2 Le jeu de données : RL-fr

Le rôle central que joue la lexicologie dans la **théorie Sens-Texte (TST)** a favorisé le développement de nombreuses ressources lexicographiques fondées sur ce cadre théorique, telles que Mel’čuk *et al.* (1995); Apresjan (2000); Mel’čuk *et al.* (1999); Mangeot (2000); Polguère (2014); Alonso Ramos (2015); L’Homme *et al.* (2009); Barrios Rodríguez (2024). Parmi celles-ci, le **Réseau lexical du français (RL-fr)** (Lux-Pogodalla et Polguère, 2011; ATILF, 2024) se distingue comme un réseau lexical de grande envergure.

Construit selon les principes de la lexicologie explicative et combinatoire (Mel’čuk *et al.*, 1995), le RL-fr constitue un *système lexical* au sens de Polguère (2014) : une modélisation du lexique français sous forme de graphe. Les nœuds correspondent aux unités lexicales (ou *lexies*), et les arêtes représentent des liens paradigmatiques ou syntagmatiques, standardisés à l’aide du système des fonctions lexicales de la TST, comme illustré à la figure 3.1.

Dans le cadre de cette étude, nous utilisons la version 3.1 du RL-fr, qui comprend environ 30 000 unités lexicales couvrant près de 19 000 vocables du français. Outre les formes propositionnelles et les exemples d’usage, cette ressource contient plus de 66 000 instances annotées de fonctions lexicales, constituant un réseau riche de relations lexicales paradigmatiques et syntagmatiques. Cette ressource constitue ainsi une base solide pour la construction d’un jeu d’évaluation fondé sur les FL.

3.2.1 Sélection des FL ciblées

L’un des aspects les plus pertinents du RL-fr pour notre étude est qu’il fournit des instances annotées de nombreuses fonctions lexicales, chacune comportant des informations

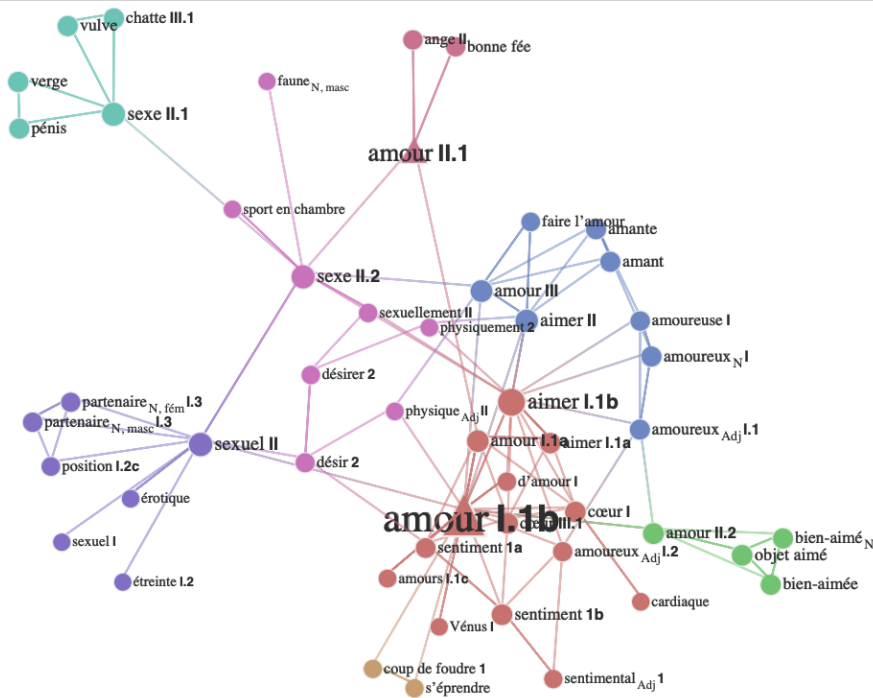


FIGURE 3.1 – Fragment du RL-fr autour de la lexie AMOUR

comme le nom de la fonction, un mot-clé et une valeur associée. Toutefois, compte tenu de l'ampleur du jeu de données, toutes les FL répertoriées ne sont pas exploitables dans le cadre de notre recherche. Une phase de filtrage, que nous détaillons ci-dessous, est donc nécessaire afin de ne conserver que les FL pertinentes pour nos objectifs.

3.2.1.1 Seuil minimal d'occurrences

La distribution des FL dans le RL-fr est fortement déséquilibrée : certaines fonctions sont instanciées par des centaines d'exemples, tandis que d'autres n'apparaissent que quelques fois. Nous avons fixé un seuil minimal de **30** occurrences par FL, afin d'assurer une représentativité suffisante des données tout en excluant les fonctions trop rares pour permettre un échantillonnage fiable. Cela garantit que chaque FL cible possède des exemples suffisants pour permettre la construction de requêtes robustes et variées.

Par ailleurs, afin de garantir la qualité linguistique des requêtes et la cohérence de l'ensemble du protocole expérimental, nous exigeons que les lexies sélectionnées—qu'il s'agisse des mots-clés ou des valeurs—soient accompagnées d'informations linguistiques pertinentes, telles que la forme propositionnelle et au moins un exemple d'usage de la lexie. En conséquence, toutes les unités lexicales dépourvues de ces informations, ainsi que les instances de

FL simple (moins de 30)	FL complexe substituée	Exemple
Incep	IncepFunc ₀ , IncepOper ₁	IncepFunc ₀ (AUBE)=BLANCHIR, IncepOper ₁ (SOMMEIL)=TOMBER
Fin	FinFunc ₀	FinFunc ₀ (NUAGE)=FUIR
Liqu	LiquFunc ₀	LiquFunc ₀ (ÉTANG)=VIDER
Caus	CausFunc ₀	CausFunc ₀ (MEUBLE)=FABRIQUER
Equip	MultS ₁	MultS ₁ (CONCERT) = ORCHESTRE

TABLEAU 3.1 – Liste des FL simples remplacées par des fonctions complexes sémantiquement équivalentes.

FL qui en dépendent, sont automatiquement exclues de notre sélection.

3.2.1.2 Fonctions lexicales standard et simples

Notre recherche se concentre sur les **FL standard simples** pour des résultats plus cohérents. Nous excluons de notre sélection les fonctions complexes et les configurations de FL, qui combinent plusieurs opérations ou qui ne sont pas représentées de manière unifiée dans les ressources existantes. Ce choix permet de maintenir une cohérence formelle dans la construction des questions polaires, en ciblant des relations identifiables et interprétables de façon stable.

Dans la pratique, certaines exceptions à ce principe ont été faites. Il existe en effet des FL dont la composante sémantique est particulièrement pertinente, mais pour lesquelles la FL simple ne dépasse pas le seuil de **30** occurrences requis dans RL-fr. Pour ces cas particuliers, nous avons choisi d’inclure des FL complexes équivalentes ou étroitement apparentées sur le plan sémantique à notre échantillonnage. Par exemple, la FL de verbes de phase *Incep*, qui exprime le sens ‘début’, est trop peu représentée dans le jeu de données. En revanche, la fonction complexe *IncepFunc₀*, formée par combinaison avec un verbe support sans contenu sémantique, offre un nombre suffisant d’occurrences. Nous avons donc approximé ces deux fonctions comme équivalentes pour l’échantillonnage, ce qui permet d’évaluer une plus grande diversité de relations sémantiques. L’ensemble des FL faisant l’objet de ce traitement approximatif est présenté dans le tableau 3.1.

3.2.1.3 Filtrage des variantes et sélection des formes canoniques

Bien que le RL-fr offre des informations détaillées sur les fonctions lexicales, sa structuration ne correspond pas toujours parfaitement à la hiérarchie que nous avons établie dans notre

cadre d'évaluation. En particulier, certaines « familles » de FL, telles que les synonymes, regroupent plusieurs variantes sémantiques, comme Syn (synonyme exact), Syn_c (synonyme avec sens moins riche), Syn_s (synonyme avec sens plus riche), etc. Cela peut entraîner des ambiguïtés, voire des recouvrements avec d'autres FL comme $Gener$. Par exemple, on observe dans le RL-fr l'instance suivante : $Syn_c(LONGUEUR) = GRANDEUR$, tandis que l'hyponyme correspondant est annoté comme : $Gener(LONGUEUR) = GRANDEUR$.

Ces cas soulignent le risque de confusion dans la construction des requêtes. Pour garantir la clarté et la précision des formulations, nous avons fait le choix de ne conserver, pour chaque famille de FL, que la forme la plus canonique—c'est-à-dire les FL identifiées par leur nom standard, sans indices additionnels.

Ce filtrage permet de mieux aligner les données du RL-fr avec notre structure hiérarchique de référence, tout en assurant l'univocité des relations testées dans les requêtes.

3.2.1.4 Complétion manuelle des cas partiels

Certaines FL, telles que A_i et Adv_i , représentent des cas particuliers dans le système des FL. En effet, ces fonctions ne produisent pas uniquement des relations paradigmatiques (par exemple, $A_1(CONDUIRE) = CONDUCTEUR_{ADJ}$, $A_2(CONNAÎTRE) = CONNU$), mais peuvent également donner lieu à des formes syntagmatiques. Ces dernières se manifestent souvent sous la forme d'un groupe prépositionnel comprenant une préposition suivie du mot-clé lui-même, comme dans $A_1(COLÈRE) = \text{'EN COLÈRE'}$ ou $Adv_1(CAPRICE) = \text{'PAR CAPRICE'}$. Or, dans la ressource **RL-fr**, les instances correspondant à ces cas syntagmatiques sont parfois incomplètes : elles ne mentionnent que la préposition, sans rappeler explicitement le mot-clé attendu à la suite. Pour remédier à cela et garantir la clarté des requêtes générées, nous avons procédé à une complétion manuelle de ces expressions, en réintégrant le mot-clé manquant dans les valeurs des FL concernées.

3.2.2 Jeu de données préparé pour l'évaluation

Hiérarchie filtrée des FL À partir de la structuration initiale des FL en deux grandes catégories—*paradigmatiques* et *syntagmatiques*—, nous avons retenu un total de 82 nœuds pertinents, comprenant à la fois des nœuds intermédiaires (catégories générales) et des nœuds terminaux (les sous-classes les plus spécifiques). Une vue simplifiée de cette hiérarchie est présentée dans la figure 3.2. Afin de faciliter l'échantillonnage contrastif dans les expériences, cette structure a été enregistrée dans un fichier JSON exploité par notre pipeline expérimental.

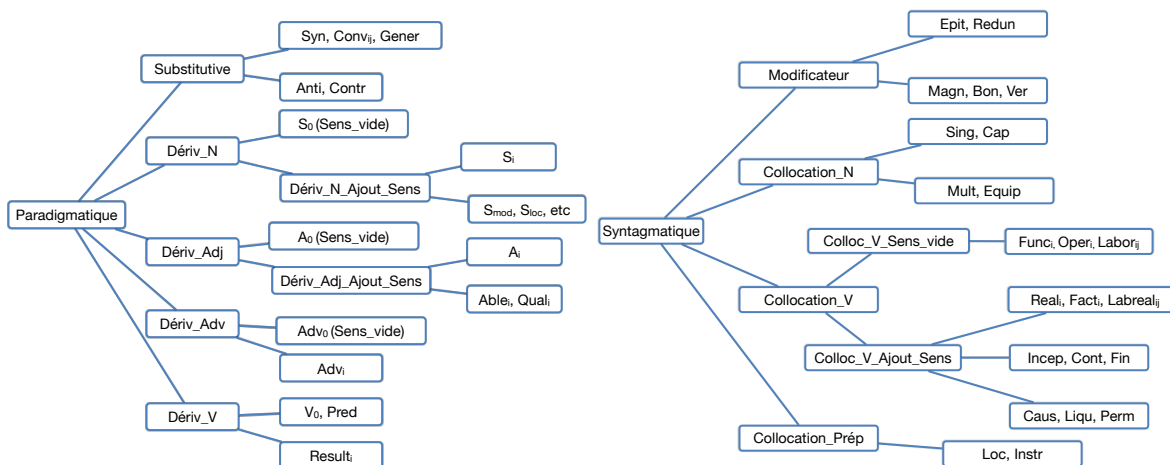


FIGURE 3.2 – Structure hiérarchique des FL utilisée dans ce projet

Descriptions formelle des FL Conformément aux recherches des FL dans la TST (Mel’čuk *et al.*, 1995; Mel’čuk, 1996; Mel’čuk et Polguère, 2021), nous avons rédigé une description formelle pour chacun des nœuds sélectionnés dans la structure des FL. Ces descriptions constituent une composante essentielle de la requête : elles permettent d’expliquer aux LLM la nature de la relation lexicale représentée par la FL cible. Ces descriptions sont aussi sauvegardées dans un fichier CSV, et leur contenu détaillé est présenté en annexe (voir l’annexe A).

Instances de FL sélectionnées À la suite des opérations de sélection et de normalisation décrites précédemment, nous avons constitué un jeu de données structuré, propre à servir de base pour les expériences à venir. Ce fichier, au format CSV, contient plus de 50 000 instances de fonctions lexicales issues du RL-fr. Chaque instance correspond à une relation FL annotée, accompagnée de métadonnées linguistiques nécessaires à la génération des requêtes. Le tableau 3.2 présente les informations pertinentes, tandis que le tableau 3.3 en illustre quelques exemples.

Colonne	Contenu
lf_id	Identifiant de la fonction lexicale
lf_name	Nom de la fonction lexicale
keyword	Mot-clé (argument) de la FL
value	Valeur associée à la FL
kw_propform	Forme propositionnelle du mot-clé
vl_propform	Forme propositionnelle de la valeur
kw_context	Exemple d’usage du mot-clé, au format <i>KWIC</i>

TABLEAU 3.2 – Contenu du jeu d’instances filtrées des FL

lf_id	lf_name	keyword	value	kw_propform	vl_propform	kw_context
ls:fr:lf:10	Anti	ACCEPTER	REFUSER	\$1 ~ \$2	\$1 ~ \$2	... Je ne peux pas accepter cette situation ...
ls:fr:lf:2	Syn	ÉGALEMENT	AUSSI	~	~	... cette dernière seront également partagés entre les deux
ls:fr:lf:126	Magn	COULEUR	VIVE	~ \$2 (de \$1)	\$1 qui est ~	... une chemise de la même couleur mon caleçon rose ...
ls:fr:lf:168	Oper ₂	DOUCHE	PRENDRE	~ de \$1	\$1 ~ \$2	... pendant sa douche . Il mettait souvent la radio
ls:fr:lf:28	A ₀	AUDITION	AUDITIF	~ de \$1	\$1 ~	... une visite médicale de sécurité (audition réflexes...)

TABLEAU 3.3 – Exemples extraits du fichier des instances de FL sélectionnées (le KWIC original correspond à une fenêtre de 13 mots centrée sur le mot-clé, ici raccourcie pour des raisons de lisibilité)

3.3 Échantillonnage des exemples contrastifs fondé sur la hiérarchie des FL

3.3.1 Motivation pour l’échantillonnage contrastif

En nous basant sur le paradigme des *Natural Instructions* (Mishra *et al.*, 2022; Wang *et al.*, 2022), qui facilite l’interaction avec les modèles via des questions-réponses structurées en requête, agrémentées de démonstrations avec peu d’exemples contrastifs, nous adoptons une approche similaire. En particulier, pour évaluer la finesse de la reconnaissance des FL par les LLM, nous avons opté pour une méthode d’échantillonnage contrastif, fournissant à la fois des exemples positifs et négatifs pour chaque FL cible. Le listing 3.1 illustre l’idée

générale de la tâche.

Listing 3.1 – Illustration conceptuelle de la tâche d'évaluation contrastive

Syn est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie désignant son synonyme exact ou approximatif.

Voici des exemples positifs de cette fonction:

nombre -> chiffre, pied -> arpon, planquer -> dissimuler

Voici des exemples négatifs de cette fonction:

sprinter -> athlète, ovale -> rond, intéresser -> s'intéresser

QUESTION:

tenir -> maintenir

Est-ce que la paire de mots ci-dessus constitue aussi un exemple de cette classe de fonction lexicale? Réponds

uniquement par '**Oui**' ou par '**Non**'

3.3.2 Stratégie de contraste fondée sur la hiérarchie

Nous exploitons la hiérarchie des fonctions lexicales pour guider la construction des exemples contrastifs. Le processus commence par la sélection progressive des nœuds à différents niveaux de granularité, depuis les catégories les plus abstraites (par exemple, *Dérivation_nominale*, *Collocation_verbale*) jusqu'aux nœuds terminaux spécifiques comme *Oper₁*, *Syn*, *A₂*, *Func₀*, etc.

Chaque nœud sélectionné est considéré comme une FL cible. Les exemples positifs sont alors prélevés parmi les paires de lexies annotées sous ce nœud dans le jeu de données, y compris celles appartenant à des sous-classes. Pour la construction des exemples négatifs, nous adoptons une stratégie contrastive fondée sur la proximité hiérarchique : les contre-exemples sont systématiquement extraits des nœuds frères, c'est-à-dire les autres classes partageant le même parent que la FL cible. Ce choix permet de construire des contrastes lexicaux fins : dans la mesure où les FL utilisées comme exemples négatifs se distinguent de la fonction ciblée tout en partageant avec elle certains traits sémantiques ou syntaxiques généraux. Cela rend la tâche plus exigeante et plus révélatrice des capacités du modèle à identifier la relation précise en jeu. Cette approche d'échantillonnage est illustré dans la figure 3.3.

Concrètement, si le nœud de *Substitutive* dans notre hiérarchie (voir la figure 3.2) est sélectionné comme cible, tous les nœuds frères (p. ex *Dériv_N*, *Dériv_Adj*) sont considérés comme des contrastes.

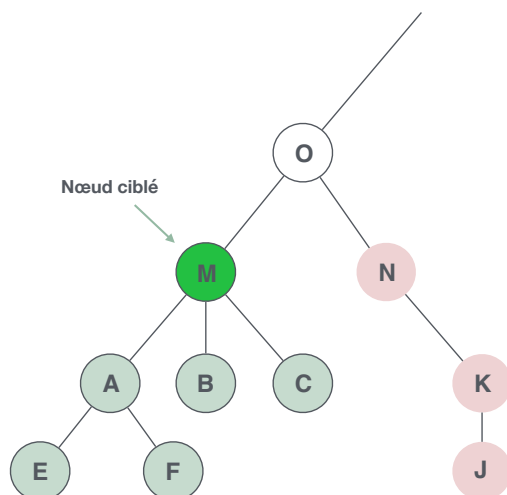


FIGURE 3.3 – Illustration d’échantillonnage contrastif structurel entre une fonction cible *A* et ses voisins hiérarchiques *B* et *C*. Les instances associées à *A* sont utilisées comme exemples positifs, tandis que celles de *B* et *C*, issus du même parent, servent de contre-exemples.

3.3.3 Pondération équilibrée pour l’échantillonnage

Afin de garantir une couverture représentative lors de l’échantillonnage aléatoire, il est essentiel d’assurer une répartition équilibrée des instances entre les différentes catégories. Prenons l’exemple illustré dans la hiérarchie présentée en figure 3.2. Si la fonction lexicale cible est *Substitutive*, ses contrastes pertinents seraient alors les catégories de différentes dérivations syntaxiques, comme *Dérivation_nominale*, etc. Une première méthode d’échantillonnage consisterait à regrouper tous les exemples issus des nœuds terminaux associés à *Substitutive*, à savoir *Syn*, *Gener*, *Anti*, *Conv_{ij}* et *Contr*, puis à en tirer un sous-ensemble aléatoire servant d’exemples positifs.

Cependant, cette méthode soulève une difficulté majeure liée à la distribution inégale des exemples dans notre corpus. Comme le montre le tableau 3.4, certaines fonctions lexicales sont associées à des milliers d’exemples, tandis que d’autres n’en comptent que quelques dizaines. Un tel échantillonnage naïf risquerait donc de sur-représenter les FL les plus fréquentes, compromettant la diversité des contrastes et introduisant un biais dans les prédictions du modèle.

Dans le cas de la catégorie *Substitutive*, un tirage aléatoire non pondéré conduirait le plus souvent à privilégier des fonctions très fréquentes telles que *Syn* ou *Gener*, au détriment des fonctions moins représentées. Cette approche compromettrait la représentativité de la catégorie globale, en masquant sa diversité interne.

Fonction lexicale	Occurrences
Syn	4801
Gener	3118
S ₀	2187
Anti	1395
A ₀	1139
Oper ₁	858
Contr	677
Conv ₂₁	151
Func ₀	49

TABLEAU 3.4 – Distribution déséquilibrée des fonctions lexicales (extrait du jeu de données).

Afin d’éviter ce déséquilibre, nous introduisons un mécanisme de **pondération équilibrée** lors de la sélection des exemples, tant pour les positifs que pour les négatifs. Plus précisément, nous appliquons une pondération inversement proportionnelle à la taille de chaque sous-catégorie dans l’arborescence. Ainsi, les nœuds moins fréquents ont une probabilité accrue d’être choisis, ce qui permet de mieux répartir les exemples à travers l’espace des FL.

3.3.4 Génération du jeu de questions

Comme présenté précédemment dans §3.1, chaque requête se compose de la description de la FL, d’exemples positifs et négatifs, ainsi qu’une question à la fin. La description est un texte fixe rédigé manuellement pour chaque nœud cible, tandis que les exemples positifs et négatifs issus de l’échantillonnage contrastif servent de démonstrations *k-shot* dans la requête. Ces exemples permettent d’illustrer concrètement les types de mot-clés et de valeurs associés à la FL cible, tout en indiquant clairement quels types de paires de lexies n’en relèvent pas. Ce contraste contribue à tracer les frontières sémantiques de la FL à partir de sa définition théorique.

La dernière partie de la requête consiste à montrer un nouveau exemple—distinct des démonstrations *k-shot*—et à interroger les LLM si cet exemple correspond, lui-même aussi, à la FL cible. Afin d’éviter que les réponses du modèle ne soient biaisées intrinsèquement par des préférences de réponse, nous générons un nombre égal de questions positives (où on attend un « oui ») et négatives (où on attend un « non »).

Lors de la préparation expérimentale, tous les exemples nécessaires à la construction des requêtes—à la fois les démonstrations *k-shot* et les exemples servant à la question—sont préalablement agencés de manière structurée et enregistrés dans un fichier JSON. Ce fichier

peut être généré de manière déterministe à l'aide d'une graine aléatoire spécifiée, afin de fixer les questions utilisées pour l'évaluation et ainsi garantir la reproductibilité des expériences. La structure des fichiers contenant les questions est illustrée dans le listing 3.2.

Nous préparons au minimum 20 questions par nœud cible de la hiérarchie des FL. Chaque question comprend 5 exemples positifs candidats, 5 exemples négatifs candidats, ainsi qu'un exemple à évaluer. Cette configuration nous permet également de tester différentes valeurs de *k-shot* tout en gardant un contrôle strict sur les variables expérimentales.

Listing 3.2 – Structure du fichier json contenant les questions générées pour les FL cibles.

```

{...
  "Syn": {
    ...
    {
      "target": "Syn",
      "Q_ID": 10,
      "expected": "Oui",
      "exemple_question": {
        "lf_id": "ls:fr:lf:2",
        "lf_name": "Syn",
        "keyword": "ballon rond",
        "value": "football",
        "kw_propform": "~ pratiqué par les $1",
        "vl_propform": "~ pratiqué par les $1",
        "kw_context": "... de se comporter brillamment. Tout le
          monde se passionnait pour le [[ballon rond]]. ..."
      },
      "ex_pos_prompt": [
        ...
        {
          "lf_id": "ls:fr:lf:2",
          "lf_name": "Syn",
          "keyword": "coup de téléphone",
          "value": "coup de fil",
          ...
        },
        ...
      ],
      "exemples_negatifs_prompt": [
        ...
        {
          "lf_id": "ls:fr:lf:20",
          "lf_name": "Gener",
          "keyword": "sandale",
          "value": "chaussure",
          ...
        },
        ...
      ]
    },
    ...
  },
  'Gener': {...}
  ...
}

```

3.4 Exécution des expériences

3.4.1 Génération des requêtes

Afin de permettre aux LLM de bien interpréter et exploiter les informations fournies dans nos requêtes, nous avons veillé à concevoir des requêtes aussi claires et structurées que possible. Plus précisément, nous distinguons deux types de requêtes : la *requête système* et la *requête utilisateur*.

La *requête système* illustré dans le listing 3.3 sert à définir explicitement la tâche attendue des modèles, ainsi que les consignes de réponse, en présupposant leur familiarité de la TST. La *requête utilisateur*, quant à elle, contient les éléments principaux déjà évoqués : la description formelle de la FL cible, les exemples positifs et négatifs (*k-shot*), ainsi que la question posée au modèle. Cette structure est illustrée dans les listings 3.4–3.5.

Listing 3.3 – Exemple de requête système

```
Tu es un linguiste expert des fonctions lexicales de la théorie
Sens-Texte. Je vais te fournir la définition d'une famille de
fonctions lexicales, suivie d'exemples positifs et négatifs.
Ensuite, je te soumettrai une paire de lexies. Tu devras
répondre par "***Oui**" si cette paire correspond au lien
lexical modélisé par cette famille de fonctions lexicales, ou
répondre par "***Non**" dans le cas contraire. Pour chaque
mot-clé, je te fournirai un contexte au format KWIC (Key Word
In Context) qui te permettra de cerner l'acception qui nous
intéresse, ainsi que la forme propositionnelle de la lexie (qui
indique sa nature prédicative). Dans la forme propositionnelle,
les premier, deuxième, troisième, etc. arguments sont
représentés par "$1", "$2", "$3", etc.
Tu ne dois répondre que par "***Oui**" ou "***Non**", sans
explication supplémentaire.
```

Listing 3.4 – Exemple de requête utilisateur : S₀

S₀ est une fonction lexicale de dérivation nominale vide de sens qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée en dénotant le sens du mot-clé lui-même, sans ajouter de sens.

Voici des exemples positifs de cette classe:

produire -> production

Exemple d'utilisation du mot-clé: ... La ferme [[produit]] des légumes bio et locaux. ...

Forme propositionnelle du mot-clé: \$1 ~ \$2 pour \$3

Forme propositionnelle de la valeur: ~ de \$2 par \$1

Réponse: **Oui**

comprendre -> compréhension

Exemple d'utilisation du mot-clé: ... attendre les découvertes de Pasteur pour [[comprendre]] la cause de l'infection et tirer ...

Forme propositionnelle du mot-clé: \$1 ~ \$2

Forme propositionnelle de la valeur: ~ par \$1 de \$2

Réponse: **Oui**

...

Voici des exemples négatifs de cette classe:

écarter -> écart

Exemple d'utilisation du mot-clé: ...: "Mon père" et d'[[avoir écarté]] une chaise de la table pour ...

Forme propositionnelle du mot-clé: \$1 ~ \$2 de \$3

Forme propositionnelle de la valeur: ~ entre \$1 et \$2 de valeur \$3

Réponse: **Non**

gommer -> gomme

Exemple d'utilisation du mot-clé: ... plume il les note pour mieux [[gommer]] rayer barrer et définir les limites ...

Forme propositionnelle du mot-clé: \$1 ~ \$2 avec \$3

Forme propositionnelle de la valeur: ~ qui sert à \$1

Réponse: **Non**

...

Listing 3.5 – Exemple de question posée à la suite d’une requête : S_0

```
QUESTION:
préhistorique -> préhistoire
Exemple d'utilisation du mot-clé: ... Or les premiers
fouilleurs de gisements [[préhistoriques]] -- même les plus
célèbres -- ...
Forme propositionnelle du mot-clé: $1 ~
Forme propositionnelle de la valeur: ~

Est-ce que la paire de mots ci-dessus constituent aussi un
exemple de cette classe de fonction lexicale? Réponds
uniquement par '**Oui**' ou par '**Non**'.
```

3.4.2 Configuration expérimentale

Dans cette étude, nous avons sélectionné trois LLM récents et performants disponibles via la bibliothèque `transformers` sur la plateforme Hugging Face (Wolf *et al.*, 2020) : Qwen2.5-14B-Instruct-1M (ci-après, QWEN), LLaMA-3.1-8B-xInstruct (ci-après, LLAMA) et Mistral-7B-Instruct-v0.3 (ci-après, MISTRAL). Tous ces modèles sont utilisés avec leurs paramètres par défaut. L’ensemble des questions associées à chaque fonction lexicale cible est exécuté séparément sur chacun de ces trois modèles.

Nous avons défini trois paramètres expérimentaux principaux. Le premier est la valeur de *k-shot* : elle détermine le nombre d’exemples inclus dans la requête. Par exemple, un paramètre $k = 6$ signifie que chaque requête comprend trois exemples positifs et trois exemples négatifs pour la FL cible. Dans chaque exemple, nous fournissons par défaut la *forme propositionnelle* du mot-clé, afin de rendre explicite son information morpho-syntaxique et d’en faciliter la désambiguïsation. Ensuite, deux autres paramètres sont manipulés de manière contrôlée : (1) la présence ou non d’un exemple d’usage du mot-clé sous forme de KWIC, visant à préciser le sens ; (2) la présence ou non de la forme propositionnelle de la valeur. Ces deux dimensions constituent des variables expérimentales supplémentaires. L’ensemble des conditions expérimentales est résumé dans le tableau 3.5.

De plus, chaque question dans une même requête a été posée **cinq** fois à chaque modèle en utilisant des graines aléatoires (*random seeds*) distinctes, afin de garantir à la fois la reproductibilité et l’observation de la variance des réponses du modèle.

Les expérimentations préliminaires ont montré que les modèles respectaient globalement les consignes de format imposées dans le prompt, en produisant presque toujours une réponse débutant par « oui » ou « non », comme demandé. Seules quelques exceptions mineures ont été

Paramètre	Description
k	Nombre d'exemples fournis dans la requête (<i>prompt</i>) ($k \in \{2, 6, 10\}$).
<code>kw-ctx</code>	Indique si le mot-clé est accompagné d'un contexte d'usage au format KWIC (booléen).
<code>vl-pfm</code>	Indique si la valeur est fournie avec sa forme propositionnelle (booléen).

TABLEAU 3.5 – Paramètres expérimentaux.

observées, notamment des variations dans la casse ou l'omission des astérisques autour des mots-clés (p. ex. `Oui` au lieu de `**Oui**`). Par souci de robustesse, nous avons néanmoins intégré un module de code pour normaliser les réponses avant les analyses.

3.5 Synthèse

Dans le but d'évaluer la compétence lexicale des LLM à travers les FL définies par la TST, nous avons adopté le paradigme des *instruction naturelles*. Celui-ci consiste à interroger les modèles au moyen de requêtes structurées contenant des exemples contrastifs.

Pour cela, nous construisons une structure hiérarchique des FL (voir chapitre 2), et choisissons le RL-fr comme source d'exemples. Après un processus de filtrage, nous avons élaboré une série de questions pour les FL cibles sélectionnées dans la hiérarchie, puis généré les requêtes correspondantes, qui ont été testées sur trois LLM. Nous avons mis en place plusieurs conditions expérimentales portant sur le type et la quantité d'informations fournies dans les requêtes, telles que le nombre d'exemples, la présence de la forme propositionnelle des lexies, ou encore l'ajout d'un exemple d'usage en contexte. Ces variations visent à examiner dans quelle mesure les LLM tirent parti de ces informations.

Les données, le code et les résultats bruts de nos expériences sont mis à disposition dans le dépôt : <https://github.com/shuxu-li/evallfs-m.a.thesis-shuxuli>. Les résultats expérimentaux pour chaque modèle et les analyses sont présentés dans le chapitre suivant.

Chapitre 4

Résultats et discussion

Nous avons évalué la performance des trois modèles sélectionnés sur un ensemble de 82 FL cibles, selon la méthodologie exposée au chapitre 3. Le présent chapitre est consacré à l'analyse et à la discussion des résultats obtenus.

Chaque question de notre jeu d'évaluation est préalablement annotée¹ avec une réponse correcte; l'ensemble des analyses de performance repose donc sur la comparaison systématique entre les réponses générées par les modèles et ces réponses de référence. Nous commençons par une analyse globale de l'exactitude (*accuracy*) et du score F1 des modèles testés. Nous examinons ensuite leur performance en fonction des différentes configurations expérimentales. Dans un second temps, nous adoptons une perspective centrée sur les FL afin d'explorer les variations de performance selon la catégorie de FL. Enfin, nous discutons des facteurs susceptibles d'expliquer ces disparités, en mobilisant plusieurs angles d'interprétation.

4.1 Performance à travers les LLM

4.1.1 Vue d'ensemble

Les résultats expérimentaux révèlent que la performance globale des trois LLM testés demeure relativement modeste. Il convient de rappeler que notre tâche repose sur des questions à polarité binaire, pour lesquelles un système répondant au hasard obtiendrait en moyenne un taux de bonne classification de 50 %. Comme l'indique le tableau 4.1, les modèles MISTRAL

1. Comme chaque paire de lexies utilisée dans les questions est échantillonnée soit parmi les exemples positifs, soit parmi les exemples négatifs, la réponse correcte est ainsi systématiquement déterminée au moment du tirage.

Model	kw-ctx	vl-pfm	$k = 2$				$k = 6$				$k = 10$			
			Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
QWEN	F	F	63.8	65.8	63.8	62.7	67.6	69.7	67.6	66.7	70.1	71.8	70.1	69.5
	F	T	64.1	66.1	64.1	63.0	67.8	69.5	67.8	67.1	70.5	71.9	70.5	70.0
	T	F	60.1	65.7	60.1	56.3	64.1	69.0	64.1	61.5	66.0	69.4	66.0	64.4
	T	T	61.0	66.2	61.0	57.6	64.7	68.6	64.7	62.7	66.4	69.0	66.4	65.1
LLAMA	F	F	56.9	62.7	56.9	51.4	58.7	63.6	58.7	54.7	60.5	65.2	60.5	57.1
	F	T	55.9	65.2	55.9	48.4	57.1	62.3	57.1	51.9	58.1	63.8	58.1	53.2
	T	F	55.3	62.8	55.3	48.2	57.3	61.3	57.3	53.0	57.0	62.4	57.0	51.8
	T	T	53.8	65.5	53.8	44.1	55.5	62.7	55.5	48.5	54.7	61.9	54.7	47.1
MISTRAL	F	F	53.0	56.7	53.0	45.1	53.6	59.2	53.6	44.8	53.2	58.4	53.2	44.0
	F	T	53.6	59.2	53.6	46.5	56.4	61.9	56.4	50.0	55.8	60.9	55.8	49.3
	T	F	50.5	54.3	50.5	36.8	51.2	57.1	51.2	37.7	52.0	57.8	52.0	40.3
	T	T	51.6	54.3	51.6	41.0	52.9	58.5	52.9	42.9	52.8	61.4	52.8	41.7

TABLEAU 4.1 – Exactitude (Acc), précision (Prec), rappel (Rec) et score F1 des trois modèles selon les différentes configurations. La précision mesure la fiabilité des réponses, le rappel évalue leur couverture, et le score F1 en donne une moyenne harmonique.

et LLAMA ne surpassent cette référence aléatoire que de façon marginale. Même pour le modèle le plus efficace, QWEN, l’exactitude atteint seulement environ 70 %, ce qui suggère que les relations lexicales impliquées dans cette tâche représentent un défi considérable pour ces LLM.

4.1.2 Biais de la polarité des réponses

Étant donné que notre ensemble d’évaluation est strictement équilibré avec un nombre égal des questions positives et négatives, toute asymétrie dans la distribution des prédictions peut révéler un biais systématique dans les sorties des LLM. Dans la figure 4.1, les rangées indiquent les étiquettes de référence, alors que les colonnes correspondent aux prédictions des modèles. Les différences observées entre faux positifs et faux négatifs mettent en évidence des biais systématiques dans les réponses. LLAMA et QWEN montrent une nette tendance à prédire « Non », tandis que MISTRAL a plutôt tendance à sur-prédire « Oui ». Ces comportements suggèrent l’existence d’heuristiques de réponse ou de biais inductifs distincts, acquis durant l’entraînement, qui peuvent influencer la prise de décision lexicale dans la tâche binaire.

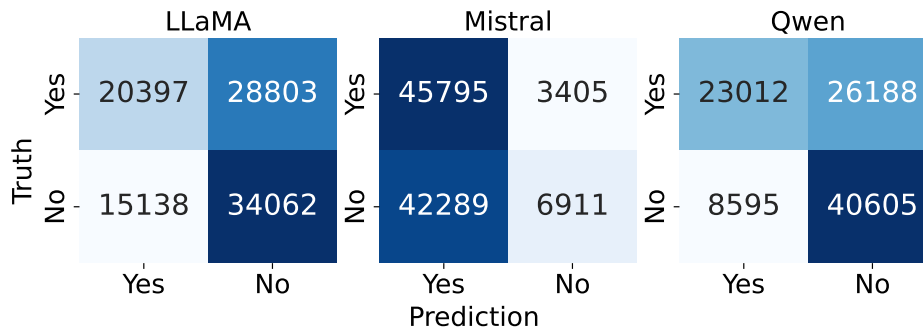


FIGURE 4.1 – Matrices de confusion des trois modèles évalués. Chaque modèle a été interrogé sur un total de 98 400 questions. Nous observons, par exemple, que MISTRAL tend à répondre « Oui » de manière générale, indépendamment de la réponse correcte attendue (45 795 fois pour les cas où la réponse correcte est « oui », et 42 289 fois pour les cas où la réponse correcte est « non »). Les cas où la réponse attendue est « non » sont ainsi fortement sous-classifiés.

4.2 Effets des configurations expérimentales

Dans le cadre de la configuration expérimentale présentée au chapitre précédent, nous avons testé chaque modèle sous trois paramètres différents (cf. tableau 3.5). Les résultats correspondants sont résumés dans le tableau 4.1. Les trois modèles évalués présentent à la fois des points communs et des divergences en fonction des conditions expérimentales. QWEN se distingue par une performance systématiquement supérieure à celle des autres modèles, suivi de LLAMA, tandis que MISTRAL affiche une exactitude globalement plus faible. Afin d’illustrer plus clairement l’impact des différents paramètres sur la performance, nous avons recours dans la figure 4.2 à un diagramme linéaire comparant les trois modèles sur chacun des paramètres. La présente section est consacrée à l’analyse détaillée de ces tendances.

4.2.1 Effets du nombre d’exemples (k -shot)

Comme le montre la figure 4.2, les trois modèles réagissent, à des degrés divers, aux variations du paramètre k -shot. Le modèle le plus sensible à ce facteur est QWEN, suivi de LLAMA, en particulier lorsque le contexte `kw_ctx` n’est pas fourni. Quant à MISTRAL, bien que sa courbe de performance en fonction de k soit plus plate, on observe néanmoins une tendance générale : plus le nombre d’exemples fournis augmente, mieux le modèle parvient à saisir la nature de la fonction lexicale cible.

Nous extrayons ici un exemple concret issu de l’historique expérimental afin d’illustrer comment l’augmentation du nombre d’exemples k -shot peut améliorer les réponses du

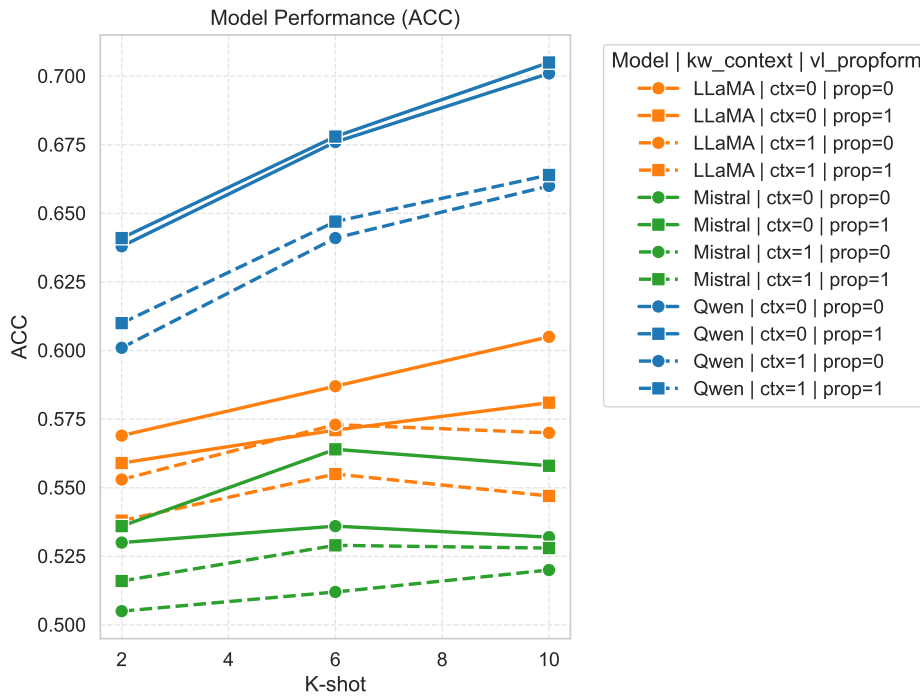


FIGURE 4.2 – Tendances d’exactitude des trois modèles selon les configurations expérimentales

modèle. Le tableau 4.2 présente un cas spécifique portant sur la FL cible Adv_1^2 , où le modèle QWEN adopte une décision plus correcte lorsqu’il dispose d’un plus grand nombre d’exemples. Cela suggère que le modèle est capable d’exploiter les régularités illustrées dans les démonstrations pour mieux identifier les paires conformes ou non à la FL cible.

4.2.2 Effet de `kw_ctx` (contexte du mot-clé) et de `vl_pfm` (forme propositionnelle de la valeur)

Ces deux paramètres ont été introduits afin d’évaluer leur rôle potentiel comme indices linguistiques permettant d’enrichir la compréhension du modèle vis-à-vis de la FL cible à partir du contexte sémantique des paires lexicales. Cependant, la figure 4.2 suggère qu’aucun des trois modèles testés ne semble réellement bénéficier de la présence de `kw_ctx`; au contraire, son ajout entraîne parfois une dégradation de performance. En revanche, le paramètre `vl_pfm` a un impact modérément positif sur QWEN et un impact significativement positif sur MISTRAL, alors qu’il n’apporte rien à LLAMA.

2. Ici Adv_1 est en contraste avec Adv_0 .

	$k = 2$	$k = 6$
Exemples positifs	COURAGE → COURAGEUSEMENT	COURAGE → COURAGEUSEMENT, FIERTÉ → FIÈREMENT, FIÈVRE → FÉBRILEMENT
Exemples négatifs	MÈRE → MATERNELLEMENT	MÈRE → MATERNELLEMENT, CORDIALITÉ → CORDIALEMENT, CONSTANCE → CONSTAMMENT
Exemple en question	SYMPTOMATIQUE → SYMPTOMATIQUEMENT	SYMPTOMATIQUE → SYMPTOMATIQUEMENT
Bonne réponse	Non	Non
Réponse de LLM	Oui	Non

TABLEAU 4.2 – Effet positif de l’augmentation de k — extrait de Adv_1 sur QWEN

Il convient de souligner que l’absence d’amélioration liée à certains éléments de la requête, comme `kw_ctx`, ne signifie pas que ce type d’information est inutile pour le repérage des relations lexicales. Cela indique plutôt que, dans leur forme actuelle, les modèles peinent à exploiter efficacement ces indices linguistiques dans le cadre de la tâche d’identification des FL. Comme le montre le tableau 4.3, si le modèle QWEN ne traite pas l’information four-

	<code>kw_ctx = True</code>	<code>kw_ctx = False</code>
Exemples positifs	HAINÉ → MORTELLE <code>kw_ctx</code> : ... se muer en une haine échappant à tout contrôle ...	HAINÉ → MORTELLE,
Exemples négatifs	AIR → RESPIRABLE <code>kw_ctx</code> : ... l’ air frais est aspiré ...	AIR → RESPIRABLE
Exemple en question	ANALOGIE → PROFOND <code>kw_ctx</code> : ... les formes de la folie l’ analogie est constante parce que	ANALOGIE → PROFOND
Bonne réponse	Oui	Oui
Réponse du LLM	Non	Oui

TABLEAU 4.3 – Effet négatif avec la présence du KWIC contexte du mot-clé—extrait de Magn sur QWEN

nie par `kw_ctx` comme un indice destiné à faciliter la désambiguïsation, mais plutôt comme un simple ajout à la requête—voire comme un élément perturbateur altérant sa compréhension de la fonction lexicale cible—il n’est guère surprenant que cette information ait un effet négatif.

En résumé, les LLM sont capables d’acquérir davantage de connaissances lexicales en

s'appuyant sur un plus grand nombre d'exemples de FL. Toutefois, ils n'exploitent pas pleinement les informations sémantique qui sont implicites dans le contexte d'usage ou dans la forme propositionnelle des lexies.

Dans les sections suivantes, nous concentrons notre analyse par rapport aux FL, avec la configuration la plus performante de chaque modèle (indiquée en gras dans le tableau 4.1), afin de minimiser les effets de confusion liés à la variation simultanée de plusieurs variables.

4.3 Performance à travers les fonctions lexicales

Dans la mesure où l'objectif central de cette étude est d'évaluer la capacité lexicale des LLM à travers les FL, il est essentiel d'adopter une perspective d'analyse centrée sur les FL elles-mêmes. Cette section vise à extraire les différentes FL cibles de la hiérarchie développée au chapitre 2 afin d'observer et d'analyser les variations de performance des trois LLM en fonction des types de relations lexicales. Nous examinons ensuite les caractéristiques sémantiques et syntaxiques associées aux FL présentant des performances contrastées, telles qu'elles sont représentées dans la hiérarchie. Cette analyse nous permet d'interroger plus finement ce que révèlent ces écarts en termes de compétence lexicale des modèles.

4.3.1 Disparité de performance selon les FL

Nous avons d'abord classé les performances des LLM sur l'ensemble des FL cibles, de la plus élevée à la plus faible, afin d'observer les disparités globales. À partir de ce classement, nous avons sélectionné plusieurs FL issues respectivement des groupes paradigmatique et syntagmatique, dont leur exactitude sont présentés dans la figure 4.3. Comme le montre cette figure, les modèles affichent des écarts d'exactitude notables selon les différentes FL cibles.

Certaines FL semblent plus faciles à apprendre pour les modèles, en particulier lorsque la distinction repose essentiellement sur la PdD. Par exemple, la classe de FL paradigmatiques *Dériv_Adj* (dérivation adjectivale) désigne les FL où on obtient une valeur adjectivale dérivée du mot-clé, tout en préservant la sémantique d'origine—par exemple, *CRIME* → *CRI-MINEL_{ADJ}* ou *ENFER* → *DAMNÉ_{ADJ}*. Elle est contrastée dans notre tâche avec d'autres types de dérivations (nominales, adverbiales, etc.) utilisées comme contre-exemples. Les résultats montrent que, lorsqu'on demande aux modèles d'évaluer si une paire telle que *SOLEIL* → *SOLAIRE* relève du même schéma, ceux-ci répondent souvent avec une grande exactitude—*QWEN* atteignant même des scores parfaits sur cette fonction dans sa configuration optimale.

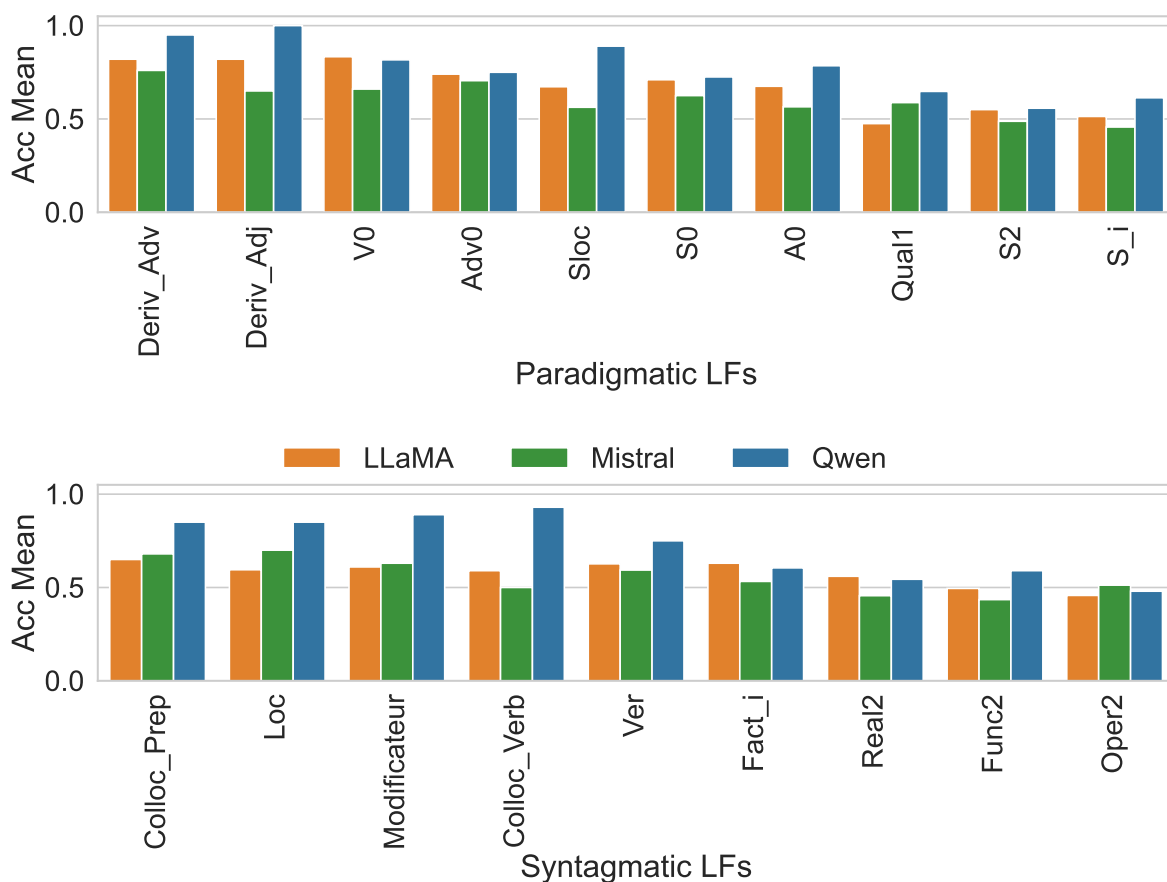


FIGURE 4.3 – Exactitude des trois LLM sur différentes classes de fonctions lexicales.

Un constat similaire peut être formulé du côté des FL syntagmatiques. Comme l’illustre la partie inférieure de la figure 4.3, les fonctions de collocations (verbales ou prépositionnelles) ainsi que les modificateurs (généralement adjectivaux ou adverbiaux), relèvent également de relations fondées sur la PdD. Il s’agit de cas où la FL retourne une valeur (préposition, verbe, adjectif ou adverbe) qui s’associe syntaxiquement au mot-clé. La bonne performance des modèles sur ces FL syntagmatiques vont dans le même sens que celles observées pour les FL paradigmatiques, suggérant que les relations de collocation basées sur la PdD sont également accessibles aux LLM.

À l’inverse, certaines FL se révèlent nettement plus complexes pour les modèles, notamment lorsqu’elles mobilisent des structures actanciennes sémantiques. Par exemple, la fonction Func_2 est définie comme une FL qui, à partir d’un mot-clé non verbal, fournit un verbe support permettant de construire une expression verbale sans altérer la signification du mot-clé. Dans cette construction, le mot-clé joue le rôle de sujet du verbe, et son argument sémantique §2 devient l’objet direct de celui-ci.

Dans notre tâche, Func_2 est mise en contraste avec d’autres fonctions apparentées telles que Func_0 ou Func_1 .³ Les exemples d’erreurs typiques présentés dans le tableau 4.4 illustrent les limites des LLM dans la prise en compte des distinctions actanciennes fines, en particulier pour des fonctions telles que Func_i . Les modèles peinent systématiquement à distinguer ces schémas sémantico-syntaxiques tellement fins, et leur performance tombe parfois en dessous du niveau attendu pour une réponse aléatoire.

FL cible	Exemple	Paire questionnée	Attendu	Réponse
S_2	RESSENTIR → SENSATION	HÔPITAL → PATIENT	Non	Oui
S_2	RESSENTIR → SENSATION	BATEAU → PASSAGER	Oui	Non
Func_2	MALADIE → ATTAQUER	TONNERRE → GRONDER	Non	Oui
Oper_2	NAUSÉE → DONNER	VISITE → FAIRE	Non	Oui
Oper_2	NAUSÉE → DONNER	NOUVELLE → ACCUEILLIR	Oui	Non

TABLEAU 4.4 – Exemples d’erreurs typiques observées dans les réponses de modèles (QWEN, $k = 10$, $\text{kw_ctx}=\text{F}$, $\text{vl_pfm}=\text{T}$)

Cette disparité suggère que les LLM n’ont pas une maîtrise homogène de l’ensemble des fonctions lexicales : leur sensibilité varie selon le type de relation ciblée. Dans la section suivante, nous examinons dans quelle mesure ces écarts de performance peuvent être corrélés à notre organisation hiérarchique des FL (cf. chapitre 2).

4.3.2 Structures hiérarchiques et performance par FL

Afin d’approfondir notre compréhension des disparités observées dans la section précédente, nous avons regroupé l’ensemble des FL en fonction de leur profondeur dans la hiérarchie, puis analysé la variation de performance des LLM à différents niveaux d’abstraction.

Dans la figure 4.4, chaque courbe représente la performance moyenne d’un modèle sur les FL cibles situées à un certain niveau de profondeur dans la hiérarchie (*Depth*), mesurée en termes d’exactitude (*Acc*). Les modèles présentent effectivement des écarts de performance systématiques selon le niveau de granularité des FL. Pour QWEN et LLAMA, les FL situées en profondeur—correspondant à des classifications plus spécifiques—s’avèrent plus difficiles à classifier, avec un déclin particulièrement marqué pour QWEN. MISTRAL semble relativement moins sensible à la profondeur dans les niveaux supérieurs de la hiérarchie, c’est en partie parce que sa performance globale est déjà très faible (à peine au-dessus du hasard), ce

3. Voir Mel’čuk et Polguère (2021); Mel’čuk (1996) pour une présentation détaillée de ces fonctions.

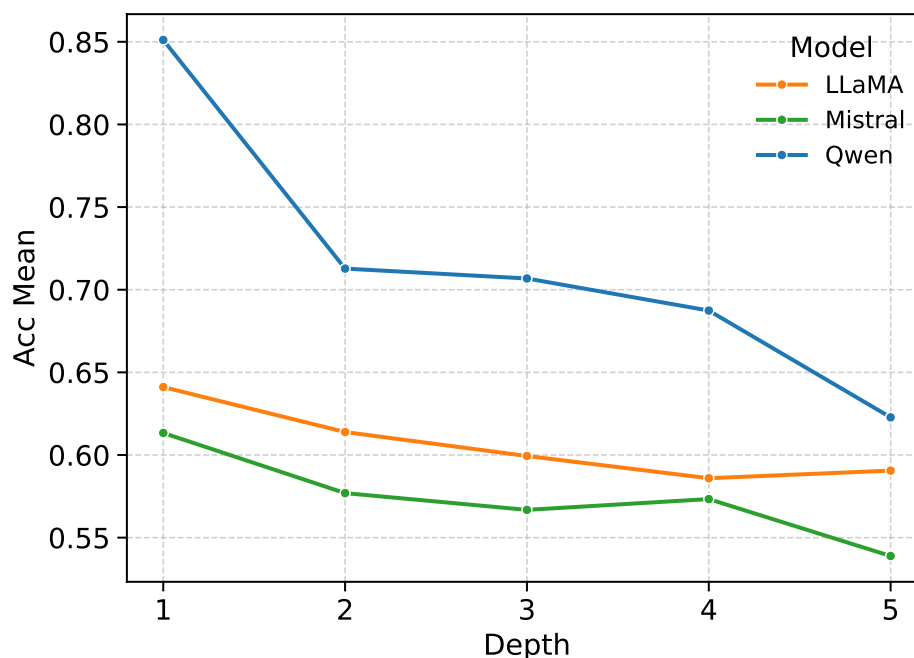


FIGURE 4.4 – Tendances de performance selon la profondeur hiérarchique des fonctions lexicales.

qui limite la marge de dégradation possible. Nous observons néanmoins une baisse significative de l’exactitude dans les niveaux les plus bas.

En reliant ces niveaux de profondeur aux disparités discutées précédemment, on constate que les FL associées à des distinctions de PdD, comme *Dérivation_Adjectivale*, se trouvent au sommet de la hiérarchie (*depth* = 1), là où les performances des modèles sont généralement élevées. À l’inverse, les FL plus complexes, comme *Func₂* discutée précédemment, sont situées plus bas dans la structure, où les classifications deviennent plus fines et plus exigeantes.

Un examen plus fin de certaines branches spécifiques de la hiérarchie permet d’illustrer de manière particulièrement nette l’effet de la profondeur sur la performance. Comme le montre la figure 4.5, nous avons sélectionné la branche correspondant à la trajectoire suivante dans la hiérarchie : *Collocation verbale* → *Verbes supports* → *Oper_i*, en nous concentrant sur le performance du modèle QWEN.

Tout d’abord, les barres en jaune correspondent aux classes de FL de plus haut niveau : les différentes collocations générales. Comme évoqué dans la sous-section précédente, les modèles, et en particulier QWEN, obtiennent de très bons résultats sur ces fonctions lexicales

fondées sur la PdD, ce qui confirme leur aisance relative à capturer des relations de collocation simples. Ensuite, les barres en vert représentent un niveau intermédiaire, où la tâche

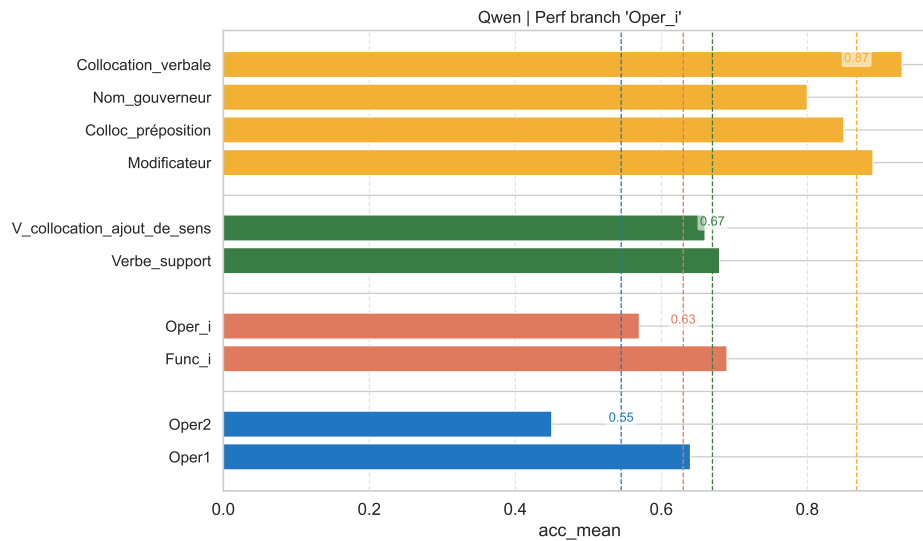


FIGURE 4.5 – Exemple de dégradation d’exactitude selon la profondeur : branche de $Oper_i$

consiste à distinguer les *Verbes supports*—qui n’ajoutent aucun contenu sémantique—d’autres collocations verbales exprimant une action sémantique plus marquée (comme ‘réaliser’, ‘causer’, etc.). Un exemple de question associée à cette tâche est présenté dans le listing 3.5. La performance de QWEN sur cette distinction est relativement stable, et proche de sa performance globale moyenne.

Listing 4.1 – Exemple de requête : verbe support

```

...
Les verbes supports sont une classe de fonctions lexicales qui,
étant donné une lexie comme mot-clé, sélectionnent une autre
lexie comme collocatif verbal qui sert de support syntaxique
sans ajouter de sens.
Voici des exemples positifs de cette classe:
fatigue -> éprouver
...

Voici des exemples négatifs de cette classe:
lit -> dormir
...

```

Les barres en rouge illustrent un niveau plus profond de cette branche, où la distinction se fait entre les sous-types de *Verbes supports* : notamment entre $Oper_i$ et $Func_i$. Ces fonctions se distinguent par le rôle syntaxique que joue le mot-clé dans la collocation : dans

Func_i, le mot-clé est sujet, tandis que dans Oper_i, il est complément d'objet direct. Comme le montre le listing 4.2, cette distinction syntactico-sémantique est plus difficile à saisir pour QWEN, dont la performance décline notablement à ce niveau.

Listing 4.2 – Exemple de requête : Oper_i

```
...
Les Oper_i sont une classe de fonctions lexicales qui, étant
donné une lexie comme mot-clé, sélectionnent une autre lexie
comme un collocatif pour construire une collocation permettant
son insertion dans une phrase sans en modifier le sens. Dans la
collocation, le mot-clé joue le rôle de complément d'objet
direct du verbe, tandis que son argument sémantique $i devient
le sujet syntaxique.

Voici des exemples positifs de cette classe:
pluie -> tomber
...

Voici des exemples négatifs de cette classe:
maladie -> souffrir
...
```

Enfin, les barres bleues correspondent au niveau terminal de cette branche : la distinction entre Oper₁ et Oper₂, qui repose sur l'argument du mot-clé qui devient sujet dans la structure. Cette distinction très fine, portant sur la structure actancielle, s'avère particulièrement difficile pour le modèle, comme l'illustre le dernier exemple ci-dessous.

Listing 4.3 – Exemple de requête : Oper₂

```
...
Oper2 est une fonction lexicale qui, étant donné une lexie
comme mot-clé, sélectionne une autre lexie comme collocatif
verbal pour construire une collocation permettant son insertion
dans une phrase sans en modifier le sens. Dans cette structure,
le mot-clé joue le rôle de complément d'objet direct du
collocatif, et son deuxième argument ($2) devient le sujet
syntaxique.

Voici des exemples positifs de cette classe:
problème -> faire face
...

Voici des exemples négatifs de cette classe:
jugement -> porter
...
```

Ces résultats renforcent l’idée que plus la distinction requise par la FL est fine et profondément implantée dans la hiérarchie, plus difficile est la tâche à distinguer pour les LLM, en particulier lorsque la tâche fait appel à des connaissances syntaxiques ou actancielle implicites. Cette observation vient appuyer notre hypothèse selon laquelle les écarts de performance entre FL sont en partie structurés par leur position dans la hiérarchie fonctionnelle.

Bien qu’il soit difficile de définir un critère théorique permettant de vérifier directement la validité de notre hiérarchie de classification, les tendances observées — notamment la baisse progressive des performances avec la profondeur — concordent avec l’intuition ayant guidé sa structuration. Cette cohérence empirique peut ainsi être considérée comme une forme de validation extrinsèque.

4.3.3 Défis posés par les FL liées à la structure actancielle

La profondeur hiérarchique joue un rôle effectivement important dans les variations de performance observées ; une autre couche de complexité émerge des structures actancielle encodées dans certaines FL. Une explication plausible réside dans le caractère conventionnel—plutôt qu’absolu—des arguments sémantiques : leur interprétation dépend souvent des usages linguistiques établis plutôt que de règles fixes. Par exemple, S_1 et S_2 , appartiennent à une classe de FL qui s’appuient directement sur la structure actancielle du mot-clé. Or, lorsque l’on compare leur performance à celle avec des nœuds de profondeur similaire comme S_{instr} ou S_{res} , qui ne font pas intervenir d’indices actanciel, on observe une divergence marquée dans les réponses des modèles, malgré une profondeur équivalente dans la hiérarchie.

Comme le montre le tableau 4.5, les FL caractérisées par des interprétations sémantiques plus explicites—sans dépendance aux arguments numérotés—sont en général mieux reconues. Cette tendance pourrait notamment expliquer pourquoi le paramètre `vl_pfm` améliore la performance de modèles comme QWEN et MISTRAL : en fournissant des signaux désambiguïsants, il compense partiellement la variabilité interprétative liée aux structures actancielle.

FL	QWEN	LLAMA	MISTRAL	Moyenne
S_1, S_2, \dots	0.60	0.54	0.53	0.56
S_{res}, S_{loc}, \dots	0.75	0.62	0.54	0.64

TABLEAU 4.5 – Contraste de performance entre des FL avec indices d’actants (S_i) et avec indices de circonstants ($S_{res}, S_{instr}, \dots$).

4.3.4 Réflexion sur les limites de la hiérarchie des FL

Comme nous l’avons décrit précédemment, notre hiérarchie des fonctions lexicales est structurée de manière arborescente selon une logique de classification priorisant d’abord la dimension de PdD, puis la dimension sémantique, et enfin la structure actancielle. Un tel choix structurel impose un ordre unique dans l’organisation des catégories, ce qui limite la possibilité d’explorer simultanément d’autres axes de distinction.

Or, certaines FL présentent des similarités syntaxiques très fortes malgré des différences sémantiques marquées. C’est notamment le cas des paires $\text{Fact}_i / \text{Func}_i$ (où le mot-clé occupe la position de sujet) ou $\text{Oper}_i / \text{Real}_i$ (où le mot-clé agit comme complément d’objet direct). D’un point de vue syntaxique, ces relations sont identiques, la différence tenant uniquement à l’interprétation sémantique : Func_i et Oper_i n’ajoutent pas de contenu sémantique, tandis que Fact_i et Real_i impliquent une notion de réalisation ou de fonctionnement.

Même pour un locuteur humain, la distinction entre certaines paires de fonctions lexicales peut s’avérer délicate, notamment lorsque la différence réside uniquement dans leur interprétation sémantique, alors que leur réalisation syntaxique est identique. C’est le cas, par exemple, de la paire Oper_2 et Real_2 , où la lexie CHANTAGE peut apparaître respectivement avec SUBIR (relation support sans ajout de sens) ou avec des verbes comme OBÉIR ou CÉDER (relation de réalisation). Intuitivement, la frontière entre ces deux classes est floue.

Curieux de savoir si les modèles rencontrent les mêmes difficultés, nous avons élargi manuellement notre ensemble d’exemples contrastifs afin d’y inclure ce type de paires syntaxiquement proches mais sémantiquement distinctes. Cela nous permet d’observer si les LLM éprouvent, eux aussi, des difficultés à distinguer ces fonctions lexicales.

Modèle	Paire de lexies	FL cible	FL vraie entre lexies	Attendue	Réponse
MISTRAL	TEMPÊTE → GRONDER	Fact_2	Func_2	Non	Oui
MISTRAL	AUBE → PARAITRE	Func_0	IncepFunc_0	Non	Oui
MISTRAL	MÉDECIN → SOIGNER	Func_0	Fact_2	Non	Oui
LLAMA	BOUCHON → BOUCHER	Func_2	Caus	Non	Oui
LLAMA	NOUVELLE → ACCUEILLIR	Real_2	Oper_2	Non	Oui
QWEN	CADAVRE → REPOSER	Fact_0	Func_0	Non	Oui
QWEN	MALADIE → CHOPER	Oper_1	IncepOper_1	Non	Oui

TABLEAU 4.6 – Exemples d’erreurs révélatrices : les modèles prédisent à tort une correspondance avec la FL cible, en raison de similarités syntaxiques entre fonctions lexicales proches.

Plusieurs exemples d’erreurs commises par les modèles sont illustré dans le tableau 4.6, et cela appuie l’hypothèse soulevée plus haut. Les paires de lexies présentées dans ce tableau

sont tous retirées des questions dont nous attendons un *Non* car leur vraie relation est différente que la FL. sont toutes extraites de questions dont la réponse attendue est *Non*, car la relation réelle entre les deux unités est différente de celle exprimée par la FL cible. En réalité, ce type de question négative permet précisément d’observer si le LLM confond la FL cible avec une fonction contrastive.

Par exemple, la paire MÉDECIN → SOIGNER est une instance de la fonction Fact_2 qui peut formuler une expression comme « le médecin soigne [qqn] » en interprétant le sens ‘exerce une fonction de ...’. Cependant, dans notre protocole, cette paire a été extraite pour interroger les modèles sur la fonction Func_0 qui ne porte aucun ajout de sens. La réponse attendue est donc clairement *Non*. Or, le fait que MISTRAL réponde *Oui* indique qu’il confond Fact_2 avec Func_0 , probablement en raison de leur forte similarité syntaxique.

Ce genre d’exemples issus du jeu de données illustrent que certaines paires, bien que distinctes sémantiquement, partagent des structures syntaxiques similaires, ce qui semble effectivement induire une certaine confusion chez les modèles. Ces résultats mettent ainsi en lumière les limites actuelles des LLM en matière de désambiguïsation fine des fonctions lexicales.

4.4 Impact de la similarité morphologique entre les mots-clés et les valeurs des FL

Les relations sémantiques et syntaxiques constituent les fondements des FL reliant deux unités lexicales. Or, dans les exemples de FL en français, ces relations s’accompagnent fréquemment de similarité morphologique entre le mot-clé et sa valeur. Il arrive en effet que l’existence d’un lien sémantique soit fortement suggérée par une similarité de forme—comme dans le cas de CONSTRUCTION et CONSTRUIRE. Toutefois, cette corrélation n’est pas toujours fiable : certains couples comme POSTE¹ (*Il était à son poste dès 8h*) et POSTE² (*J’ai envoyé le colis par la poste*) peuvent partager une forme identique tout en étant sémantiquement sans rapport. Pour juger de la pertinence d’un lien lexical entre deux unités, il est donc nécessaire d’interpréter la composition sémantico-syntaxique des lexies, et non de s’appuyer uniquement sur leur ressemblance formelle.

Cela soulève une question fondamentale concernant les modèles de langue : lorsqu’ils jugent de la validité d’une relation lexicale donnée, s’appuient-ils véritablement sur une compréhension des structures sémantiques et syntaxiques associées, ou bien se contentent-ils d’exploiter la similarité morphologique en surface ?

Afin d’approfondir cette problématique, nous calculons un score de similarité morphologique entre les paires mot-clé / valeur à l’aide de la **distance de Levenshtein** normalisée (Levenshtein, 1966), en utilisant la fonction `Levenshtein_ratio` fournie par la bibliothèque `python-Levenshtein`.⁴ Cette mesure, comprise entre 0 et 1, permet de quantifier la ressemblance formelle entre deux unités lexicales, et constitue un indicateur reconnu de proximité morphologique.

4.4.1 Corrélation entre la similarité morphologique et les réponses des modèles

Poursuivant cette idée, nous posons l’hypothèse que les modèles s’appuient surtout sur la similarité formelle entre les mots pour trancher entre une réponse *Oui* ou *Non*, sans véritablement comprendre la relation lexicale impliquée. Prenons l’exemple de la fonction S_2 . Supposons que nous soumettions au modèle deux paires de lexies : d’une part, APPARTENIR → PROPRIÉTAIRE, un exemple valide de S_2 , pour lequel la bonne réponse est *Oui*; d’autre part, AGRICULTURE → AGRICULTEUR, qui relève en réalité de la fonction S_1 , donc avec une réponse attendue *Non*. Dans ce cas, le modèle répondra-t-il *Oui* aux deux, influencé par la forte similarité morphologique, sans distinguer la différence sémantique sous-jacente ? En somme, la question posée ici est la suivante : la polarité de réponses des LLM entre *Oui* et *Non* est-elle corrélée à la similarité morphologique entre le mot-clé et la valeur ?

Pour étudier cette question, nous avons d’abord calculé, pour chaque question posée aux LLM, la distance de Levenshtein normalisée entre le mot-clé et la valeur, utilisée comme indicateur quantitatif approximatif de leur proximité formelle. Nous avons dans un premier temps mesuré la corrélation entre cette similarité morphologique et la vérité de référence—*Oui* ou *Non*—, afin d’établir une référence de performance. Dans un second temps, nous avons examiné la corrélation entre cette même similarité et les réponses effectivement produites par chaque modèle. Pour faciliter cette analyse, toutes les réponses ont été numérisées de façon binaire : une réponse *oui* est codée comme +1, et une réponse *non* comme -1, que ce soit pour les étiquettes de référence ou pour les prédictions des modèles. Nous utilisons le coefficient de corrélation de Pearson, et filtrons les fonctions lexicales pour lesquelles la corrélation entre similarité morphologique et réponse n’est pas significative ($p \leq 0,05$).

Les résultats montrent que le degré d’influence de la similarité morphologique varie selon la fonction lexicale cible. Comme le montrent les exemples présentés dans la figure 4.6,

4. <https://github.com/ztane/python-Levenshtein>

certaines cibles donnent lieu à des corrélations plus marquées, tandis que d'autres suggèrent un usage plus nuancé, voire absent, de l'indice morphologique par les modèles.

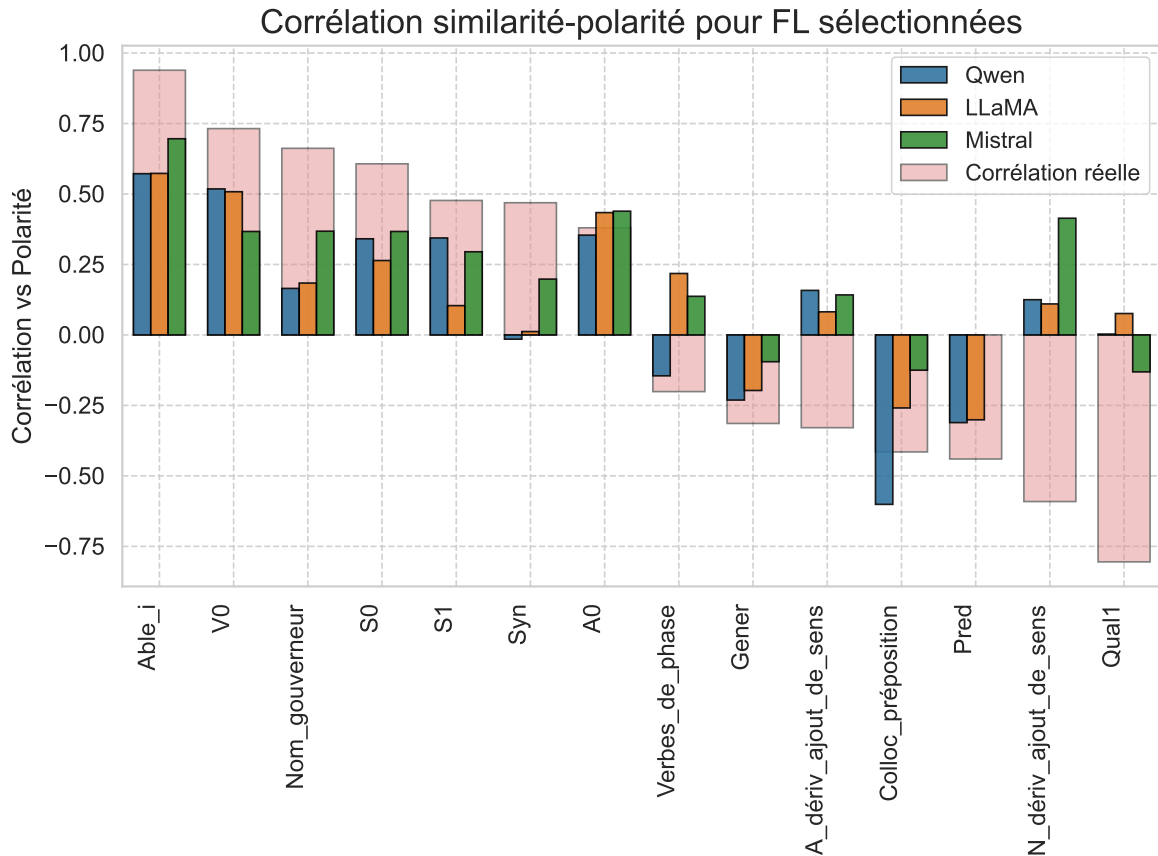


FIGURE 4.6 – Corrélation entre (i) la similarité morphologique de chaque paire de lexies (*mot-clé* et *valeur*) et (ii) la polarité de la réponse (*oui/non*).

Dans la figure 4.6, l'arrière-plan en rose pâle indique la corrélation entre cette similarité et la réponse attendue (référence), tandis que les barres colorées (bleue, orange, verte) représentent la corrélation avec les prédictions de chaque LLM. La polarité des réponses ayant été binarisée en +1 (pour *oui*) et -1 (pour *non*), des valeurs proches de +1 indiquent qu'une plus grande similarité est associée aux réponses *oui*, des valeurs proches de -1 indiquent une association avec les réponses *non*, et des valeurs proches de 0 suggèrent une absence de corrélation significative.

Nous observons tout d'abord que la corrélation entre la similarité morphologique et la polarité des réponses correctes (arrière-plan en rose pâle) est évidemment plus forte dans les FL paradigmatiques que syntagmatiques. Pour certaines FL paradigmatiques telles que $Able_i$, V_0 ou S_0 , les exemples positifs présentent généralement une forte similarité formelle

entre le mot-clé et la valeur. Cela se reflète dans la figure 4.6, où l’arrière-plan rose indique une corrélation élevée entre la similarité morphologique et la réponse attendue. Les LLM semblent « exploiter » cette ressemblance : la corrélation positive entre la similarité et la réponse *Oui* (les barres en vert/bleu/jaune) implique que ces LLM s’appuieraient sur la morphologie pour guider leur décision, ce qui se traduit aussi par leur performance élevée sur ces FL cibles dans le tableau 4.7 où leur exactitude est plus élevée. À l’inverse, dans des FL comme *Gener* ou *Pred*, où les exemples négatifs sont morphologiquement plus proches, la corrélation est négative et les modèles suivent également cette tendance, ce qui témoigne de leur capacité à tirer parti de cette structure implicite du jeu de données.

Cependant, ce comportement n’est pas universel. Pour certaines FL, bien que la corrélation de référence soit manifeste, les réponses des modèles semblent peu sensibles à cette indice, comme pour *Syn* ou *Qual₁* avec une faible corrélation de réponses. Dans certains cas (comme *N_dériv_ajout_de_sens* et *A_dériv_ajout_de_sens*), la corrélation observée dans les prédictions va même à l’encontre de celle du référence de vérité. Cette divergence est souvent accompagnée de faibles scores d’exactitude, comme illustré dans le tableau 4.7, suggérant l’incapacité des LLM à capturer les tendances morphologiques pour répondre aux questions de la tâche.

FL cible	LLAMA	MISTRAL	QWEN
Able _i	0.79	0.85	0.80
Qual ₁	0.44	0.56	0.52
V ₀	0.81	0.70	0.79
Pred	0.68	0.50	0.65
S ₀	0.70	0.62	0.70
N_Dériv_Ajout_Sens	0.44	0.35	0.30
A ₀	0.62	0.55	0.77
A_Dériv_Ajout_Sens	0.32	0.47	0.45
Syn	0.57	0.49	0.58
Gener	0.63	0.54	0.75
Nom_gouverneur	0.64	0.70	0.80
Colloc_préposition	0.68	0.65	0.85

TABLEAU 4.7 – Scores d’exactitude des modèles sur une sélection de FL représentatives.

Du côté des FL syntagmatiques, bien que la similarité morphologique entre le mot-clé et la valeur soit moins systématique, certaines fonctions comme *Nom_gouverneur* montrent néanmoins une corrélation positive dans les jeu de données. Par exemple, des couples de lexies qui se ressemblent comme *Mult(FICHE) = FICHER*, et *Cap(PLAGE) = PLAGISTE* apparaissent fréquemment dans les exemples positifs. Les LLM, notamment QWEN et LLAMA,

semblent en partie tirer profit de la régularité de cette ressemblance, avec des performances satisfaisantes sur cette FL cible. En revanche, des fonctions comme `Collocation_préposition` présentent une corrélation négative—les exemples négatifs étant morphologiquement plus similaires—et certains modèles comme QWEN, reflètent clairement cette tendance dans leurs réponses, avec une exactitude notablement supérieure.

En résumé, ces observations suggèrent que les LLM évalués exploitent effectivement la similarité morphologique comme un indice de déduction, mais de manière variée selon les FL. Lorsque cette stratégie est alignée avec la structure des données (p. ex., `Ablei`, `Nom_gouverneur`), les performances s’améliorent ; en revanche, lorsque les modèles ne parviennent pas à capter ou à appliquer cette logique (p. ex., `Syn`, `A_dériv_ajout_de_sens`), leurs réponses deviennent erratiques et leur exactitude décline. Ce constat met en lumière à la fois les forces inductives et les limites de ces modèles face aux relations lexicales encodées par les FL.

4.4.2 Biais induits par la similarité dans les exemples de la requête

Dans la section précédente, nous avons mis en évidence une corrélation entre la similarité morphologique des paires mot-clé → valeur et la polarité des réponses produites par les modèles, avec des variations notables entre différentes FL. Nous cherchons ici à explorer les causes possibles de ces disparités.

Un premier facteur que nous considérons est que les LLM ne se basent pas uniquement sur les paires directement soumises en question, mais également sur les exemples fournis dans la requête à titre d’instruction (*k*-shot). Par exemple, dans l’extrait de requête au listing 4.4 pour la fonction `A0`, on observe clairement que les exemples positifs présentent une forte similarité morphologique, tandis que les exemples négatifs sont plus disparates. Or, la section précédente montre précisément que la similarité morphologique a un impact élevé sur les réponses du modèle pour cette fonction `A0`.

Cela nous conduit à poser la question suivante : la différence de similarité morphologique entre exemples positifs et négatifs dans les requêtes pourrait-elle expliquer les variations de corrélation observées précédemment ? En d’autres termes, le design de la requête induit-il un biais qui pousse les modèles à raisonner davantage sur la forme que sur le sens ?

Listing 4.4 – Exemple d’une requête : A₀

A₀ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoi une autre lexie adjectivale dérivée qui est exactement de même sens que le mot-clé.

Voici des exemples positifs:

crime -> criminel

...

spécialement -> spécial

...

liquidation -> liquidatif

...

Voici des exemples négatifs de cette classe:

se bousculer -> nombreux

...

respecter -> attentionné

...

conte -> menteur

...

Pour tester cette hypothèse, nous comparons deux variables pour chaque FL et chaque modèle. La première est la corrélation entre la similarité morphologique et les réponses du modèle, telle que mesurée dans la section précédente. La seconde est la différence moyenne de similarité morphologique entre les exemples positifs et négatifs inclus dans les requêtes, soit $\Delta = \mu_{\text{Levenshtein-pos}} - \mu_{\text{Levenshtein-neg}}$. Nous traçons ensuite ces deux variables sous forme de diagramme de dispersion pour l’ensemble des FL, en y superposant des lignes de régression linéaire pour chaque modèle (voir figure 4.7).

Chaque point du graphique représente une fonction lexicale donnée sous un modèle spécifique, et illustre la corrélation entre l’écart de similarité morphologique dans la requête (entre exemples positifs et négatifs) et la tendance du modèle à s’appuyer sur cette similarité de surface. Les trois droites de régression appuient notre hypothèse : lorsque les exemples positifs sont morphologiquement plus proches que les négatifs, les modèles ont tendance à répondre *oui*; à l’inverse, un écart inversé entraîne plutôt des réponses *non*. On observe notamment que la pente associée à MISTRAL est plus faible, tandis que celle de QWEN est plus marquée. Cela suggère que QWEN est relativement plus apte à capter le contraste de similarité morphologique entre les exemples positifs et négatifs dans les amorces en *k*-shot, et à s’en servir pour orienter ses réponses *Oui/Non*. L’ordre relatif des pentes est cohérent avec les performances globales des modèles présentées précédemment en §4.1.1.

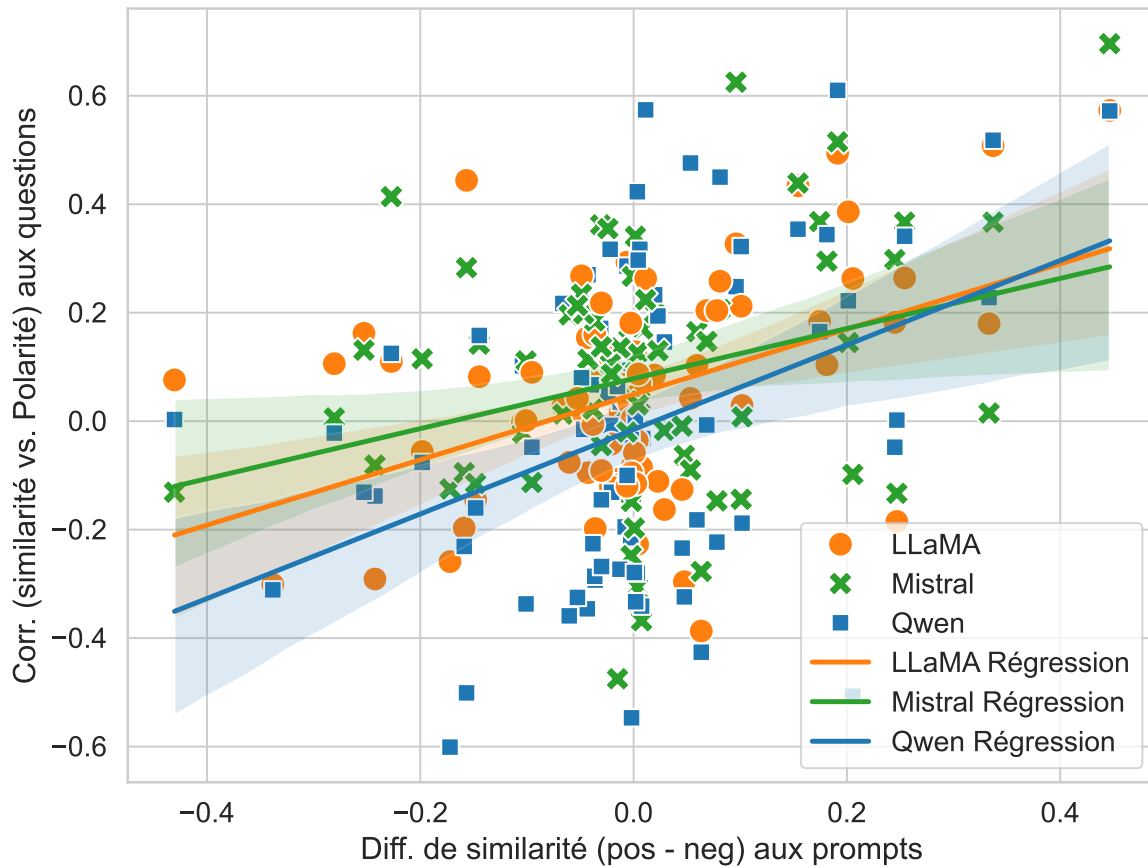


FIGURE 4.7 – Effet de la similarité des k-shot sur l'indice de corrélation des réponses

4.5 Synthèse des résultats

Nous avons tout d'abord évalué la performance globale des trois modèles (QWEN, MISTRAL, LLAMA) sur un ensemble de 82 FL cibles (§4.1.1). Les résultats montrent que leur exactitude reste relativement modeste, variant entre 50 % et 70 %. En comparant différentes configurations expérimentales (§4.2) nous avons observé que les modèles bénéficient notamment d'un plus grand nombre d'exemples (k -shot) dans les requêtes, ce qui leur permet de mieux capter les propriétés de la FL cible. De plus, QWEN et MISTRAL parviennent à exploiter efficacement les informations supplémentaires fournies par la forme propositionnelle de la valeur pour améliorer leur précision, contrairement à LLAMA qui en tire peu d'avantage. En revanche, l'ajout d'un contexte KWIC pour le mot-clé n'apporte pas de bénéfice perceptible, quel que soit le LLM.

La performance en fonction des FL révèle des disparités notables, voir 4.3. Les modèles s'en sortent généralement très bien sur les FL situées en haut de la hiérarchie, qui impliquent

des distinctions de PdD. À mesure que l'on descend dans la hiérarchie vers des FL plus spécifiques, les performances se dégradent, en particulier lorsqu'il s'agit de FL fondées sur les arguments du mot-clé. Dans ces cas, les modèles peinent à distinguer des relations plus fines, même avec des scores parfois inférieurs au hasard.

Pour comprendre les causes potentielles de ces écarts de performance entre FL, nous avons exploré l'influence de la similarité morphologique entre le mot-clé et la valeur des questions sur les réponses des modèles (§4.4). En calculant la corrélation entre cette similarité et la polarité des réponses (*Oui / Non*), nous y avons constaté une forte variabilité selon les FL. Nous avons ensuite approfondi cette analyse en mesurant l'écart de similarité entre les exemples positifs et négatifs présents dans la requête. Les résultats indiquent que plus cet écart est grand, plus les modèles ont tendance à s'appuyer sur cette similarité pour produire leurs réponses. Ces observations suggèrent que les modèles peuvent être biaisés par des régularités de surface présentes dans les exemples de la requête, ce qui affecte leur capacité à généraliser de manière fiable au-delà de ces indices formels.

Chapitre 5

Conclusion

5.1 Synthèse

Ces dernières années, les grands modèles de langues (LLM) ont dépassé leur simple fonction de générer des textes fluides et cohérents. Ils sont désormais capables d’accomplir une variété de tâches complexes en traitement automatique des langues, telles que les questions-réponses, l’extraction d’information ou la traduction automatique. Cette évolution s’accompagne d’un déplacement des axes d’évaluation : on ne se contente plus de mesurer la qualité linguistique des sorties, mais on cherche à évaluer la profondeur de la compréhension linguistique, structurelle et logique des modèles. Toutefois, la question de savoir dans quelle mesure les LLM comprennent réellement les structures sous-jacentes du langage humain reste largement ouverte.

La capacité à comprendre une langue se manifeste notamment à travers la manière dont un locuteur (ou une machine) maîtrise son lexique, lequel forme un système structuré dans lequel les unités lexicales entretiennent divers types de relations : synonymie, antonymie, dérivation morphosyntaxique, intensification, collocations verbales ou prépositionnelles, etc. Ces relations lexicales sont formalisées à travers les fonctions lexicales (FL), qui relient deux lexies—l’une jouant le rôle d’argument, l’autre celui de valeur. Le *Réseau lexical du français* (RL-fr) fournit une base de données structurée regroupant un large éventail d’instances de ces FL appliquées au français.

Dans cette étude, nous avons exploité les données du RL-fr pour évaluer la capacité des LLM à reconnaître les relations lexicales encodées par les FL. Inspirés du paradigme des *instruction naturelles*, nous avons formulé pour chaque FL cible une question binaire (Oui/Non), précédée dans la requête d’exemples positifs et négatifs illustrant cette relation. Afin

de tester la finesse de cette reconnaissance, nous avons d’abord procédé à une classification hiérarchique des FL simples et standards. Chaque nœud de cette hiérarchie a ensuite été traité comme une cible indépendante à interroger.

Pour chaque tâche, la requête inclut : (1) une description formelle de la FL cible ; (2) des exemples positifs extraits de RL-fr ; (3) des exemples négatifs sélectionnés depuis les nœuds frères dans la hiérarchie, de manière à mettre en évidence par contraste les propriétés de la FL cible ; (4) un nouvel exemple à évaluer. Pour éviter le biais de polarité dans les réponses, chaque FL cible est testée via 20 questions équilibrées : 10 positives et 10 négatives, chacune posée cinq fois avec des graines aléatoires distinctes.

Nous avons ainsi sélectionné 82 nœuds fréquents de la hiérarchie comme cibles et les avons testés sur trois modèles : QWEN, LLAMA et MISTRAL, tous exécutés avec leur configuration par défaut dans la bibliothèque `transformers`. Trois paramètres expérimentaux ont été contrôlés : (1) le nombre d’exemples dans la requête (*k-shot*, avec $k \in \{2, 6, 10\}$), (2) la présence ou non d’un contexte de mot-clé sous forme de KWIC, et (3) la présence ou non de la forme propositionnelle associée à la valeur de la FL.

Nous avons réalisé des analyses approfondies des résultats obtenus. D’abord, les taux d’exactitude atteints par les trois LLM restent relativement modestes de manière générale, ne dépassant pas 70 % dans le meilleur des cas. En comparant les différentes configurations expérimentales, nous constatons que plus le nombre d’exemples fournis dans la requête (*k-shot*) est élevé, meilleure est la performance des modèles, ce qui suggère qu’ils parviennent à exploiter les exemples dans les requêtes pour améliorer leur prise de décision. Par ailleurs, certains modèles (notamment QWEN et MISTRAL) bénéficient dans une certaine mesure de la forme propositionnelle de la valeur associée à la FL, tandis que d’autres (comme LLAMA) ne semblent pas en tirer profit. En revanche, l’ajout du contexte (KWIC) du mot-clé ne procure aucun avantage significatif, quel que soit le modèle.

En analysant la performance par FL cible, nous observons de fortes disparités. Les LLM ont tendance à bien réussir avec les FL situées en haut de la hiérarchie, en particulier celles qui impliquent des distinctions claires au niveau de la PdD. En revanche, les FL situées plus profondément dans la hiérarchie—représentant des distinctions plus fines, notamment au niveau des arguments sémantiques—sont nettement plus difficiles à traiter, avec des performances parfois inférieures au hasard.

Partant de ces constats, nous avons cherché à déterminer si la similarité morphologique entre le mot-clé et la valeur pouvait constituer un facteur explicatif de ces performances. Nous avons calculé d’abord la corrélation entre la similarité (mesurée par le ratio Levenshtein) des paires de lexies présentées dans les questions et la polarité des réponses des modèles (*oui*

ou *non*). Cette corrélation varie sensiblement selon la FL cible : pour certaines fonctions, les modèles semblent effectivement exploiter la proximité formelle pour guider leur décision, tandis que pour d’autres, cette stratégie est absente ou inefficace. Nous avons ensuite examiné si cette corrélation pouvait être influencée par la manière dont les exemples sont présentés dans la requête. En comparant la différence moyenne de similarité entre les exemples positifs et négatifs présents dans chaque requête (*prompt*), nous avons observé, à travers une analyse de régression, une tendance selon laquelle un écart plus marqué de similarité s’accompagne d’une probabilité accrue que les modèles s’appuient sur cette similarité pour répondre.

En somme, nos résultats suggèrent que les LLM actuels, malgré leurs progrès, peinent encore à intégrer la structure fine du lexique telle que modélisée par les FL. Leur comportement serait influencé par des signaux de surface, ce qui soulève des questions quant à la fiabilité réelle de leur compréhension lexicale. Ces constats ouvrent la voie à des travaux futurs visant à raffiner les méthodes d’évaluation et à améliorer les capacités des modèles dans la reconnaissance fine des relations lexicales.

5.2 Limites

Premièrement, les questions binaires (*Oui/Non*) simplifient la complexité inhérente à la compréhension des fonctions lexicales et ne reflète peut-être pas pleinement les capacités de raisonnement des modèles. Deuxièmement, notre classification actuelle repose sur une ordre allant des PdD, sémantique, jusqu’à syntaxique actancielle, ce qui pourrait ne pas rendre compte d’autres perspectives organisationnelles possibles. Troisièmement, cette recherche s’inscrit de manière cohérente dans le cadre théorique de la théorie Sens-Texte (TST), ce qui constitue une force méthodologique. Toutefois, cette focalisation en limite également la portée générale des conclusions. En outre, une évaluation humaine—incluant à la fois des participants familiers et non familiers avec la théorie des FL — pourrait constituer une base de comparaison précieuse pour les performances des modèles. Cette approche, toutefois, reste encore peu explorée du côté des participants humains. À cet égard, les travaux de Petrov *et al.* (2025a,b,c) offrent un point de référence pertinent. Dans une expérimentation préliminaire menée auprès de participants ayant reçu une formation sur les FL, les auteurs observent eux aussi de meilleures performances sur les relations paradigmatiques. Notons toutefois que leur tâche diffère de la nôtre, notamment par sa formulation analogique et son niveau de complexité.

5.3 Travaux futurs

Nous prévoyons d'étendre l'évaluation proposée dans ce banc d'essai, actuellement menée sur des LLM de taille moyenne et à poids ouverts, à plusieurs modèles plus larges et plus diversifiés.

Nous envisageons d'adapter ce jeu de données en un format de *questionnaire* destiné à des participants humains. Ce protocole permettrait d'évaluer les performances de personnes non spécialistes sur la tâche, en comparant leurs résultats avant et après une brève formation sur les FL. Par exemple, ces questionnaires pourraient être distribués à des étudiants suivant un cours en lien avec les FL, afin de mesurer la difficulté de compréhension des relations lexicales chez les non-spécialistes et d'en observer l'évolution après formation.

Nous avons exploré une première piste d'optimisation de la *requête système*, en modifiant certaines consignes afin d'inciter les modèles à raisonner indépendamment de la similarité morphologique. Un test préliminaire mené sur une question auparavant mal résolue montre que cette modification a permis au modèle de corriger sa réponse. Ces résultats suggèrent qu'il serait pertinent, dans le cadre de travaux futurs, d'intégrer explicitement ce type d'instruction dans la formulation des requêtes.

Nous prévoyons également d'assouplir le format binaire *Oui/Non* en laissant aux modèles la possibilité de justifier leurs réponses, afin d'examiner plus finement leurs capacités de raisonnement. Par ailleurs, une autre piste consisterait à reformuler les questions polaires en un format à choix multiples.

Bibliographie

ALONSO RAMOS, M. (2015). El diccionario de colocaciones del español : Una puesta al día. *Estudios de lexicografía*, 5:103–122.

APRESJAN, J. (2000). *Systematic Lexicography*. Oxford University Press.

ATILF (2024). French lexical network (fr-ln). ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

BAI, J., BAI, S., CHU, Y., CUI, Z., DANG, K., DENG, X., FAN, Y., GE, W., HAN, Y., HUANG, F., HUI, B., JI, L., LI, M., LIN, J., LIN, R., LIU, D., LIU, G., LU, C., LU, K., MA, J., MEN, R., REN, X., REN, X., TAN, C., TAN, S., TU, J., WANG, P., WANG, S., WANG, W., WU, S., XU, B., XU, J., YANG, A., YANG, H., YANG, J., YANG, S., YAO, Y., YU, B., YUAN, H., YUAN, Z., ZHANG, J., ZHANG, X., ZHANG, Y., ZHANG, Z., ZHOU, C., ZHOU, J., ZHOU, X. et ZHU, T. (2023). Qwen technical report.

BARRIOS RODRÍGUEZ, M. A. (2024). Diretes, a spanish monolingual dictionary based on lexical-semantic relations. Dans DESPOT, K. v., OSTROŠKI ANIĆ, A. et BRAČ, I., dir., *Lexicography and Semantics. Proceedings of the XXI EURALEX International Congress*, pp. 393–407, Cavtat. Institut za hrvatski jezik.

CHANG, Y., WANG, X., WANG, J., WU, Y., YANG, L., ZHU, K., CHEN, H., YI, X., WANG, C., WANG, Y., YE, W., ZHANG, Y., CHANG, Y., YU, P. S., YANG, Q. et XIE, X. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3).

CHEN, J., XU, R., FU, Z., SHI, W., LI, Z., ZHANG, X., SUN, C., LI, L., XIAO, Y. et ZHOU, H. (2022). E-KAR : A benchmark for rationalizing natural language analogical reasoning. Dans MURESAN, S., NAKOV, P. et VILLAVICENCIO, A., dir., *Findings of the Association for Computational Linguistics : ACL 2022*, pp. 3941–3955, Dublin, Ireland. Association for Computational Linguistics.

- ETTINGER, A. (2020). What BERT is not : Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.
- GLADKOVA, A., DROZD, A. et MATSUOKA, S. (2016). Analogy-based detection of morphological and semantic relations with word embeddings : what works and what doesn't. Dans ANDREAS, J., CHOI, E. et LAZARIDOU, A., dir., *Proceedings of the NAACL Student Research Workshop*, pp. 8–15, San Diego, California. Association for Computational Linguistics.
- GRIMES, J. E. (1990). Inverse lexical functions. Dans STEELE, J., dir., *Meaning-text theory : linguistics, lexicography, and implications*, pp. 350–364. University of Ottawa Press, Ottawa.
- JOUSSE, A.-L. (2010). *Modèle de structuration des relations lexicales basé sur le formalisme des fonctions lexicales*. Thèse de doctorat, Université de Montréal & Université Paris 7, Montréal/Paris.
- LAMBREY, F. (2016). Implémentation des collocations pour la réalisation de texte multilingue. Mémoire de maîtrise, Université de Montréal.
- LEVENSHTEIN, V. I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.
- L'HOMME, M.-C., LANEVILLE, M.-È. et AZOULAY, D. (2009). Le dictionnaire fondamental de l'environnement. Technical report.
- LIU, L. et LAREAU, F. (2024). Assessing BERT's sensitivity to idiomaticity. Dans BHATTIA, A., BOUMA, G., DOĞRUÖZ, A. S., EVANG, K., GARCIA, M., GIOULI, V., HAN, L., NIVRE, J. et RADEMAKER, A., dir., *Proceedings of the Joint Workshop on Multiword Expressions and Universal Dependencies (MWE-UD) @ LREC-COLING 2024*, pp. 14–23, Torino, Italia. ELRA and ICCL.
- LUX-POGODALLA, V. et POLGUÈRE, A. (2011). Construction of a French Lexical Network : Methodological Issues. Dans *Proceedings of the First International Workshop on Lexical Resources, WoLeR 2011. An ESSLLI 2011 Workshop*, pp. 54–61, Ljubljana, Slovenia.
- MANGEOT, M. (2000). Papillon Lexical Database Project : Monolingual Dictionaries and Interlingual Links. Dans *WAINS'7, 7th Workshop on Advanced Information Network and System*, p. 6, Kasetsart University, Bangkok, Thailand.

- MEL'ČUK, I. A. (1973). Towards a Linguistic 'Meaning-Text' Model. Dans KIEFER, F., dir., *Trends in Soviet Theoretical Linguistics*, pp. 33–57. Springer Netherlands, Dordrecht.
- MEL'ČUK, I. A. (1996). Lexical Functions : A Tool for the Description of Lexical Relations in a Lexicon. Dans WANNER, L., dir., *Lexical Functions in lexicography and natural language processing*. Benjamins, Amsterdam/Philadelphia.
- MEL'ČUK, I. A. (2016). *Language : From Meaning to Text*. Ars Rossica. Academic Studies Press, Boston.
- MEL'ČUK, I. A., ARBATCHEWSKY-JUMARIE, N., IORDANSKAJA, L., MANTHA, S., POLGUÈRE, A. et CLAS, A. (1999). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques IV : Recherches lexico-sémantiques IV*. Presses de l'Université de Montréal.
- MEL'ČUK, I. A., CLAS, A. et POLGUÈRE, A. (1995). *Introduction à la Lexicologie Explicative et Combinatoire*. Duculot, Louvain-la-Neuve.
- MEL'ČUK, I. A. et POLGUÈRE, A. (2007). *Lexique actif du français. L'apprentissage du vocabulaire fondé sur 20 000 dérivations sémantiques et collocations du français*. Champs linguistiques. Manuels. De Boeck, Bruxelles.
- MEL'ČUK, I. A. et POLGUÈRE, A. (2021). Les fonctions lexicales dernier cri. Dans MARENCO, S., dir., *La théorie Sens-Texte. Concepts-clés et applications*, pp. 75–155. L'Harmattan, Paris.
- MIKOLOV, T., CHEN, K., CORRADO, G. et DEAN, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*.
- MISHRA, S., KHASHABI, D., BARAL, C. et HAJISHIRZI, H. (2022). Cross-Task Generalization via Natural Language Crowdsourcing Instructions. Dans MURESAN, S., NAKOV, P. et VILLAVICENCIO, A., dir., *ACL 2022 - 60th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, pp. 3470–3487. Association for Computational Linguistics (ACL).
- OPENAI (2023). GPT-4 Technical Report.
- PETROV, A., MANCAS, A. T., BINET, V., VENANT, A., LAREAU, F., LEPAGE, Y. et LANGLAIS, P. (2025a). Q&A-LF : A French Question-Answering Benchmark for Measuring

- Fine-Grained Lexical Knowledge. Dans *Proceedings of the 15th International Conference on Recent Advances in Natural Language Processing*, Varna, Bulgaria.
- PETROV, A., VENANT, A., LAREAU, F., LEPAGE, Y. et LANGLAIS, P. (2025b). ALF : A Fine-Grained French Analogical Dataset for Evaluating Lexical Knowledge of Large Language Models. Dans *Proceedings of the 28th European Conference on Artificial Intelligence*, Bologna, Italia.
- PETROV, A., VENANT, A., LAREAU, F., LEPAGE, Y. et LANGLAIS, P. (2025c). ALF : Un jeu de données d’analogies françaises à grain fin pour l’évaluation de la connaissance lexicale des grands modèles de langue. Dans *Actes de CORIA-TALN-RJCRI-RECITAL 2025.*, vol. 1, pp. 22–49, Marseille, France. ATALA.
- POLGUÈRE, A. (2000). Une base de données lexicales du français et ses applications possibles en didactique. *Lidil-Revue de linguistique et de didactique des langues*, 21(1):75–97.
- POLGUÈRE, A. (2014). From Writing Dictionaries to Weaving Lexical Networks. *International Journal of Lexicography*, 27(4):396–418.
- RAMOS, M. A. et TUTIN, A. (1996). A Classification and Description of Lexical Functions for the Analysis of their Combinations. Dans WANNER, L., dir., *Lexical Functions in lexicography and natural language processing*. Benjamins, Amsterdam/Philadelphia.
- STEELE, J. et MEYER, I. (1990). Lexical Functions in an Explanatory Combinatorial Dictionary : Kinds and Descriptions and English Examples. Dans STEELE, J., dir., *Meaning-Text Theory : Linguistics, lexicography, and implications*, pp. 41–61. University of Ottawa Press, Ottawa.
- TOUVRON, H., LAVRIL, T., IZACARD, G., MARTINET, X., LACHAUX, M.-A., LACROIX, T., ROZIÈRE, B., GOYAL, N., HAMBRO, E., AZHAR, F., RODRIGUEZ, A., JOULIN, A., GRAVE, E. et LAMPLE, G. (2023). LLaMA : Open and Efficient Foundation Language Models.
- TURNERY, P., LITTMAN, M., BIGHAM, J. et SHNAYDER, V. (2004). Combining Independent Modules in Lexical Multiple-Choice Problems. Dans NICOLOV, N., BONTCHEVA, K., ANGELOVA, G. et MITKOV, R., dir., *Recent Advances in Natural Language Processing III : Selected papers from RANLP 2003*, pp. 101–110.
- USHIO, A., ESPINOSA ANKE, L., SCHOCKAERT, S. et CAMACHO-COLLADOS, J. (2021). BERT is to NLP what AlexNet is to CV : Can Pre-Trained Language Models Identify Analogies? Dans ZONG, C., XIA, F., LI, W. et NAVIGLI, R., dir., *Proceedings of the 59th*

Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers), pp. 3609–3624. Association for Computational Linguistics.

WANG, Y., MISHRA, S., ALIPOORMOLABASHI, P., KORDI, Y., MIRZAEI, A., NAIK, A., ASHOK, A., DHANASEKARAN, A. S., ARUNKUMAR, A., STAP, D., PATHAK, E., KARAMANOLAKIS, G., LAI, H., PUROHIT, I., MONDAL, I., ANDERSON, J., KUZNIA, K., DOSHI, K., PAL, K. K., PATEL, M., MORADSHAHI, M., PARMAR, M., PUROHIT, M., VARSHNEY, N., KAZA, P. R., VERMA, P., PURI, R. S., KARIA, R., DOSHI, S., SAMPAT, S. K., MISHRA, S., REDDY A, S., PATRO, S., DIXIT, T. et SHEN, X. (2022). Super-NaturalInstructions : Generalization via declarative instructions on 1600+ NLP tasks. Dans GOLDBERG, Y., KOZAREVA, Z. et ZHANG, Y., dir., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

WIJESIRIWARDENE, T., WICKRAMARACHCHI, R., GAJERA, B. G., GOWAIKAR, S. M., GUPTA, C., CHADHA, A., REGANTI, A. N., SHETH, A. et DAS, A. (2023). ANALOGICAL - A Novel Benchmark for Long Text Analogy Evaluation in Large Language Models. Dans BOYD-GRABER, J., OKAZAKI, N. et ROGERS, A., dir., *FINDINGS OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, ACL 2023*, pp. 3534–3549. 61st Annual Meeting of the the Association-for-Computational-Linguistics (ACL), Toronto, CANADA, JUL 09-14, 2023.

WOLF, T., DEBUT, L., SANH, V., CHAUMOND, J., DELANGUE, C., MOI, A., CISTAC, P., RAULT, T., LOUF, R., FUNTOWICZ, M., DAVISON, J., SHLEIFER, S., von PLATEN, P., MA, C., JERNITE, Y., PLU, J., XU, C., LE SCAO, T., GUGGER, S., DRAME, M., LHOEST, Q. et RUSH, A. (2020). Transformers : State-of-the-art natural language processing. Dans LIU, Q. et SCHLANGEN, D., dir., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, pp. 38–45, Online. Association for Computational Linguistics.

YUAN, S., CHEN, J., SUN, C., LIANG, J., XIAO, Y. et YANG, D. (2024). ANALOGYKB : Unlocking Analogical Reasoning of Language Models with A Million-scale Knowledge Base. Dans KU, L.-W., MARTINS, A. et SRIKUMAR, V., dir., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pp. 1249–1265, Bangkok, Thailand. Association for Computational Linguistics.

ZHAO, W. X., ZHOU, K., LI, J., TANG, T., WANG, X., HOU, Y., MIN, Y., ZHANG, B., ZHANG, J., DONG, Z., DU, Y., YANG, C., CHEN, Y., CHEN, Z., JIANG, J., REN, R., LI, Y., TANG, X., LIU, Z., LIU, P., NIE, J.-Y. et WEN, J.-R. (2023). A Survey of Large Language Models. *arXiv e-prints*, p. arXiv :2303.18223.

ŽOLKOVSKIJ, A. K. et MEL'ČUK, I. A. (1967). Essai d'une théorie sémantique applicable au traitement de langage (modèle "sens-texte"). Dans *Actes de COLING'67—Conférence internationale sur le traitement automatique des langues*, vol. 1, pp. 1–8. Association for Computational Linguistics.

ŽOLKOVSKIJ, A. K. et MEL'ČUK, I. A. (1970). Sur la synthèse sémantique. *T.A. Informations*, 2:1–85.

Annexe A

Description des nœuds ciblés

Cette annexe présente les descriptions des FL utilisées pour la génération des requêtes dans le cadre de cette étude. Elles sont sauvegardées dans un fichier CSV, et leur utilisation est décrite au chapitre 3.

FL_paradigmatiques Les `FL_paradigmatiques` sont une classe de fonctions lexicales qui prennent un mot-clé comme argument, et renvoient les dérivations syntaxiques ou sémantiques comme valeur de ce mot-clé.

Substitutive Les fonctions lexicales de `Substitutive` sont une classe de fonctions lexicales qui, pour une lexie donnée, retournent une lexie de la même partie du discours ou une expression équivalente à cette partie du discours. Elles expriment la synonymie (exacte, approximative ou générique), ou une opposition de sens (antonymes, contrastifs, conversifs, etc.).

Subst_sens_similaire Les fonctions lexicales de substitutives de sens similaire sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie désignant une même entité ou un concept sémantiquement proche.

Syn `Syn` est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie désignant son synonyme exact ou approximatif.

Conv_{ij} `Convij` est une famille de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoie une autre lexie exprimant le même sens, mais avec une conversion des arguments i et j sur le plan sémantique.

Gener `Gener` est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie qui désignent son hyperonyme ou son terme générique, c'est-à-dire un terme plus général qui le catégorise.

Subst_sens_opposé Les fonctions lexicales de substitutive opposé sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie comme valeur, désignant un antonyme, un contraste conceptuel ou simplement une conversion des arguments sémantiques.

Anti `Anti` est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie désignant son antonyme.

Contr `Contr` est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie désignant son contraste sémantique ou conceptuel, sans qu'il s'agisse nécessairement d'un antonyme.

Dérivation_nominale Les fonctions lexicales de dérivation nominale sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie nominal sémantiquement dérivée de ce mot-clé.

S₀ `S0` est une fonction lexicale de dérivation nominale vide de sens qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominal dérivée en dénotant le sens du mot-clé lui-même, sans ajouter de sens.

N_dériv_ajout_de_sens `N_dériv_ajout_de_sens` est une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie nominale dérivée enrichie par un ajout de sens, qui dénote soit un argument sémantique soit un circonstant typique de ce mot-clé.

S_i `Si` est une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un certain argument sémantique (numéro \$i) de ce mot-clé.

S₁ S₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote le premier argument (\$1) du mot-clé.

S₂ S₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote le deuxième argument (\$2) du mot-clé.

S₃ S₃ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote le troisième argument (\$3) du mot-clé.

S₄ S₄ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote le quatrième argument (\$4) du mot-clé.

N_dériv_circonstancielle Les fonctions lexicales de dérivation nominale circonstancielle sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un circonstant (instrument, méthode, manière, lieu ou résultat) typique du mot-clé.

S_{instr} S_{instr} est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un instrument typique de ce mot-clé.

S_{med} S_{med} est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un moyen ou une méthode typique de ce mot-clé.

S_{loc} S_{loc} est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un lieu typique de ce mot-clé.

S_{res} S_{res} est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie nominale dérivée sémantiquement (pas forcément morphologiquement) qui dénote un résultat typique de ce mot-clé.

Dérivation_adjectivale Les fonctions lexicales de dérivation adjectivale sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une lexie adjectivale dérivée tout en préservant un lien sémantique avec ce mot-clé.

A₀ A₀ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui est exactement de même sens que le mot-clé.

A_dériv_ajout_de_sens Les fonctions lexicales de dérivation adjectivale vide de sens, sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adjectivale dérivée qui caractérise l'argument \$i du mot-clé, soit par son propre sens, soit par le sens d'une capacité ou d'une possibilité.

A_i Les A_i sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adjectivale dérivée qui caractérise l'argument sémantique \$i du mot-clé par son sens.

A₁ A₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le premier argument (\$1) du mot-clé par son sens.

A₂ A₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le deuxième argument (\$2) du mot-clé par son sens.

A₃ A₃ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le troisième argument (\$3) du mot-clé par son sens.

A₄ A₄ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le quatrième argument (\$4) du mot-clé par son sens.

A_dériv_qualificative Les fonctions lexicales de dérivations adjectivales qualificatives sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient un adjectif ou un syntagme adjectival typiquement utilisé pour qualifier l'argument

sémantique \$i du mot-clé. Cela exprime soit une capacité ou possibilité, soit une probabilité ou tendance.

Able_i Les Able_i sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adjectivale dérivée qui caractérise l'argument \$i de ce mot-clé. Ce dérivé exprime une capacité ou une possibilité, souvent formulée par 'tel que \$i peut ~' ou 'tel qu'on peut ~ \$i'.

Able₁ Able₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le premier argument (\$1) de ce mot-clé. Cette dérivation exprime une capacité ou une possibilité, souvent formulée par 'tel que \$1 peut ~'.

Able₂ Able₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le deuxième argument (\$2) de ce mot-clé. Cette dérivation exprime une capacité ou une possibilité, souvent formulée par 'tel qu'il peut ~\$2'.

Qual_i Qual_i est une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adjectivale dérivée qui caractérise l'argument \$i comme ayant tendance à faire/être ~ .

Qual₁ Qual₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le premier argument (\$1) du mot-clé, comme ayant tendance à faire/être ~.

Qual₂ Qual₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adjectivale dérivée qui caractérise le deuxième argument (\$2) du mot-clé, comme ayant tendance à faire/être ~.

Dérivation adverbiale Les fonctions lexicales de dérivation adverbiale sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adverbiale dérivée tout en préservant un lien sémantique avec ce mot-clé.

Adv_dériv_sens_vide Adv_dériv_sens_vide sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie adverbiale dérivée de même sens.

Adv₀ Adv₀ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adverbiale dérivée de même sens que le mot-clé.

Adv₁ Adv₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie adverbiale dérivée qui caractérise la manière dont le premier argument (\$1) du mot-clé agit ou se manifeste par le sens du mot-clé.

Dérivation_verbale Les fonctions lexicales de dérivation verbale sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie verbale dérivée dérivé à partir de ce mot-clé.

V_dériv_sens_vide Les fonctions lexicales de dérivations verbales vides de sens sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie verbale dérivée dérivé à partir du mot-clé sans ajout de sens.

V₀ V₀ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie verbale dérivée dérivé ayant exactement le même sens que le mot-clé.

Pred Pred est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie verbale qui correspond au prédicat sémantique central du mot-clé.

V_dériv_ajout_de_sens V_dériv_ajout_de_sens sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une lexie verbale sémantiquement liée au mot-clé mais dont le sens n'est pas exactement le même.

Result_i Les Result_i sont une famille de fonctions lexicales qui, étant donné une lexie comme mot-clé, renvoient une autre lexie verbale qui décrit la situation de l'argument \$i résultant nécessairement du fait de mot-clé.

Result₂ Result₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, renvoie une autre lexie verbale dérivée qui décrit la situation du deuxième argument (\$2) résultant nécessairement du fait de mot-clé.

FL_syntagmatiques Les fonctions lexicales syntagmatiques sont un groupe de fonctions lexicales qui associent un mot-clé à un collocatif fréquent, en fonction de contraintes syntaxiques et sémantiques.

Modificateur Les fonctions lexicales de collocations modificateurs sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe destiné à modifier le mot-clé dans une collocation.

Modif_sens_vide Les fonctions lexicales de collocations modificateurs sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe dont le sens est déjà implicite dans la définition du mot-clé.

Epit Epit est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif épithète pléonastique qui modifie le mot-clé dans une collocation qui forme un cliché stylistique.

Redun Redun est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif épithète redondant qui modifie le mot-clé dans une collocation pour en préciser l'acception.

Modif_ajout_de_sens Les fonctions lexicales de collocations avec ajout de sens sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe qui modifie le mot-clé avec un sens spécifique.

Magn Magn est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe qui modifie le mot-clé en exprimant l'intensité ou la grandeur du mot-clé.

Ver Ver est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe qui modifie le mot-clé en exprimant le sens 'tel qu'il faut', c'est-à-dire une conformité à une norme implicite par rapport au mot-clé.

Bon Bon est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif en rôle d'adjectif ou d'adverbe qui modifie le mot-clé en exprimant un jugement positif de la part du locuteur par rapport au mot-clé.

Colloc_préposition Les fonctions lexicales de préposition sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une préposition qui l'introduit dans une collocation.

Loc_{in} Loc_{in} est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie prépositionnelle introduisant le mot-clé en tant que lieu dans une collocation.

Instr Instr est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie prépositionnelle exprimant le moyen ou l'instrument, en l'introduisant dans une collocation signifiant 'au moyen de ~'.

Nom_gouverneur Les fonctions lexicales de gouverneurs nominaux sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif nominal qui gouverne le mot-clé dans une collocation.

Unité Les fonctions lexicales de gouverneurs nominaux d'unité sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne une unité typique ou un individu responsable de ce mot-clé.

Cap Cap est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne un individu responsable de ce mot-clé.

Sing `Sing` est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne une unité de ce mot-clé.

Groupe Les fonctions lexicales de gouverneurs nominaux de groupe sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne un ensemble régulier ou un groupe effectuant l'activité du mot-clé.

Mult `Mult` est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne un ensemble régulier de ce mot-clé.

Equip `Equip` est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie nominale gouvernant le mot-clé dans une collocation qui désigne un groupe effectuant l'activité du mot-clé.

Collocation_verbale Les fonctions lexicales de collocations verbales sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent un verbe pour construire une collocation avec ce mot-clé.

Verbe_support Les verbes supports sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif verbal qui sert de support syntaxique sans ajouter de sens.

Func_i Les `Funci` sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans la collocation, le mot-clé occupe la position de sujet du verbe, tandis que son argument sémantique \$i (s'il y a lieu) devient le complément d'objet.

Func₀ `Func0` est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans la collocation, le mot-clé joue le rôle de sujet du verbe sans qu'aucun autre argument du mot-clé se manifeste dans la phrase.

Func₁ Func₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans cette structure, le mot-clé joue le rôle de sujet du verbe, et son premier argument (\$1) devient le complément d'objet.

Func₂ Func₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans cette structure, le mot-clé joue le rôle de sujet du verbe, et son deuxième argument (\$2) devient le complément d'objet.

Func₃ Func₃ est une fonction lexicale qui, étant donné un mot-clé, sélectionne une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans cette structure, le mot-clé joue le rôle de sujet du verbe, et son troisième argument (\$3) devient le complément d'objet.

Oper_i Les Oper_i sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans la collocation, le mot-clé joue le rôle de complément d'objet direct du verbe, tandis que son argument sémantique \$i devient le sujet syntaxique.

Oper₁ Oper₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme un collocatif pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans cette structure, le mot-clé joue le rôle de complément d'objet direct du collocatif, et son premier argument (\$1) devient le sujet syntaxique.

Oper₂ Oper₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans cette structure, le mot-clé joue le rôle de complément d'objet direct du collocatif, et son deuxième argument (\$2) devient le sujet syntaxique.

Oper₃ Oper₃ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation permettant

son insertion dans une phrase sans en modifier le sens. Dans la collocation, le mot-clé joue le rôle de complément d'objet direct du collocatif, et son troisième argument (\$3) devient le sujet syntaxique.

Oper₄ Oper₄ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation permettant son insertion dans une phrase sans en modifier le sens. Dans la collocation, le mot-clé joue le rôle de complément d'objet direct du collocatif, et son quatrième argument (\$4) devient le sujet syntaxique.

V_collocation_ajout_de_sens Les fonctions lexicales de collocations verbales avec ajout de sens sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif verbal pour construire une collocation exprimant un sens spécifique. Dans la collocation, le verbe prend le mot-clé comme sujet ou objet syntaxique.

verbe de réalisation (Vreal) Les fonctions lexicales de collocations de verbes de réalisation sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif verbal pour construire une collocation exprimant le sens ' ~ se réalise', ' ~ fonctionne' ou 'utiliser ~'.

Real_i Les Real_i sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation le verbe prend le mot-clé comme complément d'objet direct, exprimant l'idée de '\$i utilise ~' dont l'argument \$i du mot-clé occupe la position de sujet du verbe.

Real₁ Real₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme complément d'objet direct en exprimant le sens de '\$1 utilise ~', et le premier argument (\$1) du mot-clé joue un rôle de sujet du verbe.

Real₂ Real₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette

collocation, le collocatif verbal prend le mot-clé comme complément d'objet direct en exprimant le sens de '\$2 utilise ~', et le deuxième argument (\$2) du mot-clé joue un rôle de sujet du verbe.

Real₃ Real₃ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme complément d'objet direct en exprimant le sens de '\$3 utilise ~', et le troisième argument (\$3) du mot-clé joue un rôle de sujet du verbe.

Real₄ Real₄ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme complément d'objet direct en exprimant le sens de '\$4 utilise ~', et le quatrième argument (\$4) du mot-clé joue un rôle de sujet du verbe.

Real_Ω Real_Ω est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme complément d'objet direct en exprimant le sens de '\$Ω utiliser ~', dont le sujet \$Ω n'est pas un argument sémantique du mot-clé.

Fact_i Fact_i sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme sujet et exprime un sens '~ fonctionne pour \$i' ou '~se réalise pour \$i'.

Fact₀ Fact₀ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme sujet et exprime un sens '~ fonctionne' ou '~ se réalise', et aucun autre argument du mot-clé ne se manifeste comme dépendant syntaxique du verbe.

Fact₁ Fact₁ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette

collocation, le collocatif verbal prend le mot-clé comme sujet et exprime un sens ‘~ fonctionne pour \$1’ ou ‘~se réalise pour \$1’, dont le premier argument (\$1) du mot-clé joue un rôle de complément d’objet du verbe.

Fact₂ Fact₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme sujet et exprime un sens ‘~ fonctionne pour \$2’ ou ‘~se réalise pour \$2’, dont le deuxième argument (\$2) du mot-clé joue un rôle de complément d’objet du verbe.

Labreal₁₂ Labreal₁₂ est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation. Dans cette collocation, le collocatif verbal prend le mot-clé comme complément d’objet indirect et exprime un sens ‘\$1 soumet \$2 à l’action de ~’.

Verbes_de_phase Les fonctions lexicales de collocation Verbes_de_phase sont une classe de fonctions lexicales qui, étant donné une lexie comme mot-clé, sélectionnent une autre lexie comme collocatif pour construire une collocation exprimant un changement de phase du mot-clé, soit les sens ‘début’, ‘fin’, ‘continuation’, ou ‘préparation’.

Incep Incep est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation exprimant l’action de commencement du mot-clé.

Fin Fin est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation exprimant la mise en fin du mot-clé.

Verbes_causatifs Les verbes causatifs sont une classe de fonctions lexicales qui, pour un mot-clé donné, renvoient un collocatif verbal de ce mot-clé qui exprime un sens lié à la causalité : ‘causer ~’, ‘faire cesser ~’ ou ‘ne pas faire cesser ~’.

Caus Caus est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation exprimant généralement le sens ‘causer que ~ ait lieu’.

Liqu Liqu est une fonction lexicale qui, étant donné une lexie comme mot-clé, sélectionne une autre lexie comme collocatif verbal pour construire une collocation exprimant le sens ‘causer que ~ s’arrête’ ou ‘faire cesser ~’.