Créativité Computationnelle : conception et développement d'algorithmes pour la génération automatique de phrases littéraires

Luis Gil Moreno Jimenez

▶ **To cite this version:**

Luis Gil Moreno Jimenez. Créativité Computationnelle : conception et développement d'algorithmes pour la génération automatique de phrases littéraires. Algorithme et structure de données [cs.DS]. Université d'Avignon, 2022. Français. NNT : 2022AVIG0107 . tel-04050746

HAL Id: tel-04050746
https://theses.hal.science/tel-04050746

Submitted on 29 Mar 2023

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

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

ACADÉMIE D'AIX-MARSEILLE
AVIGNON UNIVERSITÉ

THÈSE

présentée à Avignon Université pour obtenir le diplôme de DOCTORAT

SPÉCIALITÉ : Informatique

École Doctorale 536 « Agrosciences et Sciences »
Laboratoire Informatique d'Avignon (EA 4128)

Créativité computationnelle : Conception et développement d'algorithmes pour la génération automatique de phrases littéraires

par
Luis-Gil MORENO-JIMÉNEZ

Soutenue publiquement le 18-11-2022 devant le jury composé de :

M^{me}	Béatrice DAILLE	Professeure, Nantes Université, France	Rapportrice
M.	Gerardo REYES-SALGADO	Professeur, CENIDET, Mexique	Rapporteur
M^{me}	Pascale SEBILLOT	Professeure, IRISA, France	Examinatrice
M.	Rachid ELAZOUZI	Professeur, Avignon Université, France	Examinateur
M.	Luis MENESES-LERIN	MCF, Université d'Artois , France	Examinateur
M.	Antoine DOUCET	Professeur, La Rochelle Université , France	Examinateur
M.	Rafael PEREZ Y PEREZ	Professeur, UAM-Cuajimalpa, Mexique	Examinateur
M^{me}	Roseli S. WEDEMANN	Professeure, UERJ, Brésil	Codirectrice
M.	Juan-Manuel TORRES-MORENO	MCF HC HDR, Avignon Université, France	Directeur

Laboratoire Informatique d'Avignon

Remerciements

J'aimerais remercier chaleureusement à mes directeurs de thèse M Juan-Manuel TORRES-MORENO et Mme Roseli WEDEMANN, pour leur intérêt, leur soutien et leurs nombreux conseils durant le développement de ce projet de recherche. Ce projet n'aurait pas été possible sans le soutien de mes proches, que j'ai en quelque sorte abandonné ces derniers mois pour réussir cette étape de ma vie, merci pour leurs conseilles, leurs attentions et leurs encouragements qui m'ont accompagné au cours des dernières années. Je remercie particulièrement à Antoine, Arthur, Alix, Moisés, Timothée, Clém (merci trop pour ta compagnie et ta patience) et Patty, dont leurs suggestions et commentaires ont été de grand importance pour l'élaboration de ma thèse. Merci aussi aux membres du jury de ma soutenance pour son inestimable participation et remarques : Mme Béatrice DAILLE et M Gerardo REYES, les rapporteurs et M Antoine DOUCET, Mme Pascale SÉBILLOT, M Rafael PEREZ, M Luis MENESES et M Rachid ELAZOUZI, les examinateurs.

Un grand merci à ma mère, pour son soutien moral et matériel, mais surtout pour sa confiance inébranlable dans mes décisions, qui ne sont pas toujours à la portée de tous. J'ai aussi une pensée particulière pour ma grand-mère, qui m'a appris à relever des défis sans craindre l'échec et à profiter de la vie sans redouter le regard des autres.

Un enorme agradecimiento a mi madre, pour su apoyo moral y material, pero sobre todo por su confianza inquebrantable frente a mis decisiones que no siempre son comprendidas por todos. Un pensamiento especial a mi abuelita, de quién aprendi a asumir los retos sin miedo al fracaso y a disfrutar de la vida sin temor a las incisivas miradas ajenas.

Au terme de ce parcours, je remercie toutes les personnes qui ont accepté de participer comme évaluateurs des résultats produits tout au long de cette these. Enfin, merci au *Consejo Nacional de Ciencia y Tecnología* (CONACYT) du Mexique, au Laboratoire Informatique d'Avignon (LIA) et à la Fédération de recherche Agorantic d'avoir financé partiellement ce projet de recherche.

Résumé

Dans le présent ouvrage, nous abordons de manière générale l'étude de la créativité, avec un intérêt particulier pour la manière dont elle est créée à l'aide de dispositifs artificiels, et nous présentons un traitement plus ciblé et plus formel de la génération artificielle de texte littéraire. Dans *The creative mind : Mythes and mécanismes* (Boden, 2004), Margaret Boden explique que le processus créatif est un chemin intuitif suivi par les humains pour générer de nouveaux artefacts qui sont appréciés pour leur nouveauté, leur importance pour la société et leur beauté. Elle propose une classification de la créativité dans les trois catégories suivantes :

— *Créativité combinatoire*, où des éléments connus sont fusionnés pour la génération de nouveaux éléments ;
— *Créativité exploratoire*, où la génération se fait à partir de l'observation ou de l'exploration ; et
— *Créativité transformationnelle*, où les éléments générés sont le produit de modifications ou d'expériences appliquées aux objets produits par la créativité exploratoire.

La recherche de processus automatisés capables de générer des artefacts de manière créative a récemment donné naissance à un domaine de recherche appelé *Créativité computationnelle*, qui offre des perspectives intéressantes dans divers domaines artistiques tels que les arts visuels, la musique et la littérature. Bien que des avancées significatives aient été réalisées dans ce domaine, il existe des difficultés et des limites liées à la complexité inhérente à la compréhension du processus créatif chez l'humain.

Notre objectif principal dans cette étude concerne la Génération Automatique de Texte (GAT) et, plus particulièrement, la génération de phrases littéraires. Nous visons donc le problème du développement de techniques automatiques (algorithmes) pour générer des objets linguistiques qui sont des phrases ou des parties de paragraphes qui peuvent être perçues comme appartenant à un texte littéraire. La plupart des recherches concernant la GAT évitent le genre littéraire en raison de sa complexité. Certaines difficultés fondamentales concernent l'ambiguïté du sens et même l'absence d'une définition universelle de ce qu'est un texte littéraire. Également, les documents littéraires font souvent

référence à des mondes ou des situations imaginaires ou allégoriques, contrairement aux genres qui traitent de la communication écrite de faits. Ces caractéristiques et d'autres, comme l'élégance ou l'utilisation de mots rares dans la littérature, font de la génération et de l'analyse automatiques de textes littéraires une tâche complexe et difficile.

En raison des difficultés évoquées et afin d'aborder le problème de la GAT de manière réalisable, nous partons d'un point de vue pragmatique et nous adoptons une définition opérationnelle de ce qu'est une phrase littéraire, basée sur la structure des corpora littéraires. Nous considérons ainsi qu'une phrase est littéraire, si elle est possède une structure grammaticale et un vocabulaire existant dans un corpus suffisamment large et considéré comme littéraire par les personnes. Pour atteindre nos objectifs, nous avons collecté des textes littéraires et constitué trois corpora en français, espagnol et portugais, composés exclusivement de documents littéraires, tels que des romans, des nouvelles, des récits, du théâtre et poésie.

Nous présentons dans cette thèse une nouvelle approche pour la génération de phrases littéraires. Notre proposition est basée sur trois nouveaux corpora littéraires que nous avons construits, ainsi que des techniques de réseaux de neurones artificiels, des modèles de langage et une d'analyse syntaxique superficielle. Nos modèles GAT analysent les corpora littéraires afin d'extraire et d'exploiter leurs structures grammaticales, sémantiques et linguistiques. Nous avons également considéré la génération de rimes (assonante et consonante), en tenant compte la rime sémantique. Nous avons également proposé plusieurs protocoles d'évaluation manuelle, permettant de mesurer la qualité des phrases générées par nos modèles GAT littéraires. Les résultats que nous avons obtenus sont assez encourageants. Nos systèmes génèrent des phrases grammaticalement correctes et suffisamment cohérentes, perçues comme littéraires dans une bonne mesure. De plus, ces résultats soutiennent notre affirmation (notre hypothèse) selon laquelle il est possible de générer, à partir de structures de phrases littéraires connues, de nouvelles phrases avec une nouvelle sémantique, et en tenant compte également de la signification émotionnelle des textes d'origine.

Abstract

In the present work, we broadly approach the study of creativity, with a special interest on how it is created with artificial devices, and present a more focused and formal treatment of the artificial generation of literary texts. In *The creative mind : Myths and Mechanisms* (Boden, 2004), Margaret Boden discusses how the creative process is an intuitive path followed by humans to generate new artifacts that are valued for their novelty, significance to society and beauty. She proposes a classification of creativity into the following three categories :

— *Combinatorial Creativity*, where known elements are merged for the generation of new elements ;
— *Exploratory Creativity*, where generation occurs from observation or exploration ; and
— *Transformational Creativity*, where the generated elements are the result of modifications or experimentation applied to objects produced by exploratory creativity.

The search for automated processes capable of creatively generating artifacts has recently given rise to a field of research called *Computational Creativity*, which offers interesting perspectives in various fields of art such as the visual arts, music, and literature. Although there have been significant advances in the field, there are difficulties and limitations related to the inherent complexity of understanding the creative process in the human mind.

Our main purpose in this study regards Automatic Text Generation (ATG) and, more specifically, the generation of literary sentences. We thus aim at the problem of developing automatic techniques (algorithms) for generating linguistic objects which are sentences or parts of paragraphs that can be perceived as belonging to a literary text. Most research regarding ATG avoids the literary genre due to its intricate complexity. Some basic difficulties involve the ambiguity of meaning and even the constitution of a universal definition of what a literary text is. Literary documents often refer to imaginary or allegorical worlds or situations, in contrast to genres that deal with the written communication of facts. These characteristics and others, such as the drive for elegance or the use of rare words in literature, make the tasks of automatic generation and analysis of literary texts a complex and difficult endeavor.

Due to the aforementioned difficulties and in order to approach the ATG problem in a feasible way, we start from a pragmatic point of vue and take an operational definition of what a literary sentence is, based on the structure of a literary corpus. We thus consider that a sentence is literary, if it consists of a grammatical structure and vocabulary existing in a sufficiently large corpus that is considered to be literary by human beings. To achieve our goals, we have collected literary texts and produced three corpora in French, Spanish and Portuguese, composed exclusively of literary documents, such as novels, short stories, narrative, drama and poetry.

We thus present in this thesis a new approach to the generation of literary sentences. Our proposal is based on the three new literary corpora that we have constructed, artificial neural network techniques, classical language models and shallow parsing. Our ATG models analyze the literary corpora in order to extract and exploit their fine grammatical, semantic, and linguistic structure. We have also regarded the generation of rhyme (assonant and consonant), considering semantic rhyme. We have also proposed manual evaluation protocols that can measure the quality of the sentences generated by our literary ATG models. The results that we have obtained are quite encouraging, because our systems generate grammatically correct and sufficiently coherent sentences, which are perceived as literary to a good degree. Moreover, these results support our claim (our working hypothesis) that it is possible to generate, from known literary sentence structures, new sentences with new semantics, considering also the emotional meaning of an original text.

Table des matières

1 Introduction **10**
- 1.1 L'évolution du processus créatif et son implication dans les Beaux arts . . . 10
- 1.2 Créativité computationnelle . . . 11
  - 1.2.1 Génération automatique de texte et créativité computationnelle . . . 13
  - 1.2.2 Les défis de la CC pour la génération de texte littéraire . . . 14
- 1.3 Objectifs . . . 16
- 1.4 Contributions . . . 16
- 1.5 Plan de la thèse . . . 17

2 Créativité computationnelle **18**
- 2.1 Analyse sémantique au moyen d'apprentissage profond . . . 19
  - 2.1.1 Plongements lexicaux . . . 19
  - 2.1.2 Modèle *Skip-gram* . . . 19
- 2.2 Génération Automatique de Texte (GAT) . . . 21
  - 2.2.1 Génération de texte non littéraire . . . 21
  - 2.2.2 Génération de texte littéraire . . . 25
- 2.3 Émotions dans les textes . . . 28
- 2.4 Corpus pour l'analyse littéraire ou émotionnelle . . . 31
- 2.5 Conclusion . . . 34

3 Construction de corpus littéraires **35**
- 3.1 Description des corpus MegaLite . . . 36
  - 3.1.1 Corpus MegaLite-Es . . . 36
  - 3.1.2 Corpus MegaLite-Fr . . . 37
  - 3.1.3 Corpus MegaLite-Pt . . . 37
  - 3.1.4 Prétraitements du corpus MegaLite . . . 38
- 3.2 Versions alternatives . . . 39
  - 3.2.1 Modèle de représentation continue (Word2vec) . . . 40
  - 3.2.2 Version étiquettes grammaticales (POS) et version lemmatisée . . . 42

3.2.3 Modèle de langue *n*-grammes . . . 44
3.3 Corpus littéraire d'émotions **LiSSS** . . . 45
3.3.1 Annotation d'émotions . . . 46
3.3.2 Stratégie de vote démocratique . . . 48
3.4 Application des corpus littéraires : une analyse stylistique statistique . . . 49
3.4.1 Divergence de distribution de probabilités appliquée aux styles . . . 51
3.5 Conclusion . . . 52

4 Modèles pour la génération de texte littéraire 55
4.1 Génération de texte : une approche stochastique (Stoch) . . . 57
4.1.1 Première étape : étude stochastique de la langue . . . 57
4.1.2 Deuxième étape : étude sémantique de la langue . . . 59
4.2 Génération de texte au moyen de *canned text* . . . 61
4.2.1 Génération des Structures Grammaticales partiellement-Vides . . . 61
4.2.2 Modèle d'analyse pertinent avec Word2vec (CaP) . . . 62
4.2.3 Modèle de composition de vecteurs (CaV) . . . 64
4.2.4 Modèle basé sur des traits psychologiques (CaT) . . . 67
4.3 Génération de rimes sémantiques (CaR) . . . 69
4.3.1 Rime sémantique . . . 70
4.3.2 Production de rimes avec similarité sémantique . . . 70
4.3.3 Première étape : *canned text* . . . 72
4.3.4 Deuxième étape : sélection du vocabulaire . . . 72
4.4 Conclusion . . . 76

5 Expériences de nos modèles génératifs 78
5.1 Description des expériences . . . 79
5.1.1 Modèle Stoch . . . 79
5.1.2 Modèle CaP . . . 80
5.1.3 Modèle CaV . . . 81
5.1.4 Modèle CaT . . . 82
5.2 Protocoles d'évaluation . . . 83
5.2.1 Évaluation linguistique . . . 83
5.2.2 Évaluation littéraire et émotionnelle . . . 86
5.3 Modèle CaR . . . 88
5.3.1 Évaluation de la rime en espagnol . . . 88
5.3.2 Réalisation des phrases en français . . . 89

6 Conclusion et perspectives 94

A Caractérisation du corpus MegaLite 101

B Exemples des phrases générées **105**

Liste des illustrations **113**

Liste de tableaux **114**

Bibliographie **115**

Bibliographie personnelle **124**

Chapitre 1

Introduction

1.1 L'évolution du processus créatif et son implication dans les Beaux arts

L'humain est le seul être sur terre possédant des systèmes de communication plus complexes en lui permettant d'exprimer ses besoins physiques, tels que la faim, la douleur et la satisfaction. En plus, avec l'évolution de nos aptitudes cognitives comme l'observation, l'attention, la mémoire et le langage, l'humain est aussi capable d'exprimer et d'interpréter des éléments intangibles abstraits comme les sentiments ou les émotions. Cela nous a donné la possibilité de nous développer considérablement en tant que société, faisant de nous l'espèce prédominante parmi toutes les autres (Clutton-Brock et al., 2009; Darwin, 1909).

Une des premières manifestations artistiques exprimant une émotion a été trouvée dans la région Madhya Pradesh en Inde, la sculpture Daraki-Chattan, datant des années 700 000 av. J.-C. (Bednarik, 1996). À partir de ce moment, on perçoit une évolution dans les artefacts trouvés ultérieurement (voir figure 1.1a). Ces découvertes ont été les précurseurs de ce que nous appelons aujourd'hui les beaux arts. Le terme a été inventé en 1746 dans l'ouvrage *Les Beaux-Arts réduits à un même principe* (Batteux, 1824), dans lequel sont regroupées les techniques propres à la bonne exécution de certains des principaux arts comme la danse, la peinture, la sculpture, la musique et la littérature.

Bien qu'il n'existe pas une méthode universelle pour la production artistique, Boden (2004) décrit le processus créatif comme le moyen adéquat pour la réalisation d'artefacts artistiques, perçus comme beaux et précieux pour l'humanité. Le processus créatif consiste en une série d'étapes allant de la concep-

(a) *Peinture de la caverne « El castillo » 39 000 av. J.-C.*

(b) *« Here Was The Final Blind Hour » .*

FIGURE *1.1* **–** *Exemples des anciennes et nouvelles pièces d'art. (Source : `https://www.ancienthistorylists.com/pre-history/top-10-oldest-art-ever-discovered/`).*

tualisation d'une idée ou d'un sentiment à son expression au moyen des beaux-arts. Une des premières approches pour extrapoler ce processus à un espace symbolique compréhensible pour les machines est proposée dans (Colton et al., 2012), *Computational creativity : The final frontier ?*. Ce livre aborde la créativité computationnelle, un domaine de recherche dans lequel les scientifiques analysent le processus créatif afin de concevoir des modèles automatisés pour proposer de nouveaux paradigmes pour la création d'artefacts artistiques.

1.2 Créativité computationnelle

Les recherches en Créativité computationnelle (CC) ont focalisé leurs études sur l'analyse des éléments qui interagissent au cours du processus créatif. L'objectif consiste à adapter ce processus dans le domaine du calculable, en trouvant des méthodes automatiques efficaces pour représenter les capacités cognitives de l'être humain dans un espace compréhensible par les machines. Le concept *créativité* a été décomposé en trois catégories (Boden, 2004).

1. *Créativité combinatoire* (CCO), où des artefacts connus sont fusionnés pour la génération de nouveaux artefacts ;
2. *Créativité exploratoire* (CE), où la génération se fait à partir de l'observation ou de l'exploration ; et
3. *Créativité transformationnelle* (CT), où les éléments générés sont le produit de modifications ou d'expériences appliquées aux objets produits par la créativité exploratoire.

E não há deus nem preceito, nem arte,
um palácio de arte e plástica
as máquinas pasmadas de aparelhos
num mundo de poesias e versos

o seu macaco era duas máquinas
horaciano antes dos poetas
para as consolas dos computadores
num mundo de poesias e carmes

--- traduction approximative ---

Et il n'y a ni dieu ni précepte, ni art,
un palais d'art et de plastique
les machines étonnées d'appareils
dans un monde de poésie et de ver

votre singe était deux machines
horaciano avant des poètes
pour consoles d'ordinateurs
dans un monde de poésies et chants

(a) *Sonnet généré en portugais par PoeTryMe (Oliveira et Cardoso, 2015).*

(b) *Art réalisé par WOMBO Dream (`https://app.wombo.art/`).*

FIGURE *1.2* **–** *Artefacts artistiques réalisés par des modèles d'IA.*

Ces nouveaux concepts ont été utiles pour assimiler de manière pratique le processus créatif pour la génération d'artefacts artistiques. Au cours de la dernière décennie, nous avons assisté au développement d'une quantité importante de travaux où des quantités massives de données ont été traitées à partir des approches de CC pour la production de peintures, de musique ou de textes littéraires. En conséquence, nous disposons de modèles d'intelligence artificielle pour générer des tableaux, voir par exemple : « *Here Was The Final Blind Hour* » du projet **Neural net** [1], tirage d'archives, 2016 (voir figure 1.1b).

Il existe également des modèles comme celui proposé par Minu et al. (2022) qui est capable de générer de la musique à partir de représentations matricielles (*Piano roll*) et des réseaux neuronaux profonds. Le projet *PoeTryMe* de Oliveira et Cardoso (2015) permet la production de poèmes (voir figure 1.2a) ; ou bien le modèle proposé par Pérez y Pérez (2015) avec le système MEXICA qui est capable de produire des récits préhispaniques.

1. Site web : `https://miketyka.com/?s=deepdream`

1.2.1 Génération automatique de texte et créativité computationnelle

La génération automatique de textes (GAT) est un domaine du traitement automatique de la langue naturelle (TAL) [2] qui a fait des progrès significatifs ces dernières années (Oruh et al., 2022; Alsayadi et al., 2022; Minu et al., 2022). L'objectif étant de développer des modèles informatiques capables de mimer les capacités langagières humaines. Certains de ces travaux visent uniquement l'automatisation et l'incrément de la productivité dans les domaines académique et technologique. Les modèles pour la création de chat-bots ou la génération de résumés textuels sont quelques exemples des sujets les plus abordés dans le GAT (Torres-Moreno, 2014; Sridhara et al., 2010a; Fu et al., 2014).

Cependant, des sujets comme la génération automatique de textes littéraires ont été très peu abordés par la communauté scientifique du TAL. Ce n'est que récemment que des recherches ont été entreprises sur des modèles génératifs de littérature, focalisant sur la production de poèmes ou des récits. La complexité de la génération de textes littéraires artificiels réside dans leurs caractéristiques très particulières : structure syntaxique, composition lexicale et relations sémantiques. Un écrivain est capable de gérer ces éléments avec ses compétences linguistiques et ses connaissances extra-linguistiques pour réussir à évoquer des émotions chez le lecteur. Pour un système automatique cela est une autre affaire. Dans cette thèse, notre objet d'étude est l'analyse de documents littéraires afin de produire leur modélisation dans un espace de représentation adéquat pour la génération automatisée de phrases artificielles.

Afin de répondre à nos objectifs, nous avons combiné des méthodes probabilistes, des modèles de langage (Manning et Schütze, 1999) et des méthodes classiques comme la méthode « texte en boîte » (Molins et Lapalme, 2015) pour la génération de texte. Également, nous avons fait appel à la méthode Word2vec (Mikolov et Zweig, 2012; Huang et al., 2012) pour l'analyse sémantique. Par contre, certaines méthodes récentes basées sur des réseaux de neuronaux profonds, comme les *transformers* (Radford et al., 2019) n'ont pas été retenues dans cette étude. En effet, au début de ce projet, les *transformers* n'étaient encore pas assez explorés et, lorsqu'ils ont atteint un état de maturité et d'exploitabilité, nos expériences étaient à un état avancé et le retour en arrière n'était pas une option envisageable. En outre, nous avons considéré que des méthodes GAT basées sur les réseaux neuronaux fonctionnant comme une boîte noire, ne facilite

2. Le traitement automatique des langues (TAL) est un domaine multidisciplinaire impliquant la linguistique, l'informatique et l'intelligence artificielle, qui vise à créer des outils de traitement de la langue naturelle pour diverses applications. Le TAL est sorti des laboratoires de recherche pour être progressivement mis en œuvre dans des applications informatiques nécessitant l'intégration du langage humain à la machine. Source : `https://fr.wikipedia.org/wiki/Traitement_automatique_des_langues`

ni leur adaptation ni leur explicabilité. Cela laisse comme seule alternative, un développement basé sur des implémentations superficielles, que nous considérons insuffisantes pour aborder un problème aussi complexe que l'analyse et la production littéraire artificielle.

1.2.2 Les défis de la CC pour la génération de texte littéraire

La génération de texte littéraire est un défi particulièrement intéressant par rapport à la génération de textes non littéraires. Surtout, en considérant que la perception littéraire ou *littérarité*[3] d'un texte sera perçue différemment chez les lecteurs. Selon Aron (1984), la littérarité est une propriété intrinsèque du texte littéraire et l'analyse du texte doit permettre, en théorie de la détecter. On assume donc que la subjectivité dans la littérarité complique la tâche d'évaluation et rend presque impossible de garantir qu'un texte produit par les algorithmes de GAT sera perçu comme littéraire. L'évaluation dévient encore plus difficile étant donné l'absence d'une définition universelle de littérature, puisque, pour parvenir à une perception littéraire homogène, il faudrait partir d'une définition littéraire unique. Pour positionner notre travail, dans cette thèse, nous avons initialement introduit une définition de ce qui est une « phrase littéraire » :

> **Définition 1** : *Une phrase littéraire est une phrase qui diffère des phrases de langue générale, car elle contient des éléments (noms, verbes, adjectifs, adverbes) qui sont perçus comme élégants ou moins familiers que leurs équivalents en langue générale.*

Avec cette idée, par exemple, la phrase en langue générale :

— *« Je me suis arrêté pour jeter un coup d'œil aux vieux livres dans la bouquinerie pas loin de chez moi. »*,

pourrait être légèrement réécrite pour générer, en accord avec notre définition, les phrases littéraires suivantes.

— *« Je me suis arrêté pour regarder des livres anciens dans la librairie près de ma maison. »*
— *« J'ai regardé pendant quelques instants quelques vieux livres dans la librairie à proximité de ma maison. »*
— *« J'ai feuilleté quelques instants des livres antiques dans la librairie près de ma demeure. »*

Bien sûr, un auteur pourrait décider d'écrire un texte littéraire en se basant exclusivement sur des phrases appartenant à la langue générale. Par exemple,

3. « La littérarité est ce qui est propre à la littérature » (`https://fr.wikipedia.org/wiki/Littérarité`). « Littérarité : Théorie sémiotique de la littérature qui doit permettre de caractériser tout texte littéraire par rapport à ceux qui ne le sont pas » (`http://stella.atilf.fr/Dendien/scripts/tlfiv5/advanced.exe?8;s=317538495;`).

José Agustín dans « De perfil » où l'extrait : « *... me quedé dormido en el Jardín. Supongo que el sol y lo fresco del aire crearon el término exacto para adormecerme.* »[4] [5], utilise de belles expressions littéraires dans un texte débordant de langue généralle et vulgaire. Une définition comme celle que nous avons proposée reste donc insuffisante pour produire des phrases littéraires de façon automatisée. Nous avons contourné ce problème en faisant appel à une définition opératoire de phrase littéraire qui sera expliquée au le Chapitre 4.

Un autre problème qui complique la tâche de la génération automatique de textes littéraires est le manque de corpus constitués exclusivement de documents littéraires. Comme déjà évoqué, les textes littéraires représentent un défi majeur pour les recherches en CC et GAT. Tout d'abord, parce que le niveau du discours littéraire est plus complexe que celui d'autres genres, tels que le genre journalistique, scientifique ou académique. Deuxièmement, les documents littéraires font souvent référence à des mondes ou des situations imaginaires, allégoriques ou métaphoriques, contrairement aux genres journalistique ou encyclopédique qui décrivent principalement des situations ou des événements factuels. Et finalement parce qu'il est difficile d'établir une frontière précise entre langue générale et langue littéraire.

Pour l'analyse textuelle (peu importe le genre), il est nécessaire de disposer de corpus de taille et de qualité adéquates. Avec ces ressources, nous serons en mesure d'extraire et d'analyser les éléments nécessaires pour interpréter et simuler le processus créatif, à partir d'une approche computationnelle pour la génération de phrases littéraires. Cependant, nous nous trouvons dans une situation précaire au niveau de corpus littéraires, dont la plupart sont dans des langues disposant de beaucoup de ressources comme l'anglais (Papay et Padó, 2020; Yang et Coxhead, 2020), en délaissant des langues riches et complexes en termes de syntaxe et de lexique telles que le français, l'espagnol ou le portugais.

D'autre part, il y a une très grande disponibilité de corpus composés de documents journalistiques, de documents techniques, d'opinions et de textes extraits de réseaux sociaux. Cependant, ce type de documents n'est pas vraiment utile dans les travaux de CC, car ils ne présentent pas la richesse linguistique littéraire, comme le vocabulaire ou les structures syntaxiques complexes. Les textes journalistiques ou techniques ont une structure fixe, avec un vocabulaire très formel cherchant à exprimer des idées concrètes. En outre, les corpus issus de textes extraits des réseaux sociaux comme dans (Navas-Loro et al., 2017; Villena-Román et al., 2013) contiennent une telle quantité de fautes d'orthographe, et sont pollués par des symboles comme les émoticônes ou les *emojis*, qui ne sont pas exploitables par les modèles de génération littéraires. Pour ces

4. J. Agustín. *De perfil*, Joaquín Mortiz, México, 1993.

5. « ... Je me suis endormi dans le jardin. Je suppose que le soleil et la fraîcheur de l'air ont créé le terme idéal pour me bercer. »

raisons, une partie de nos travaux a été consacrée à la construction de corpus littéraires. Cette partie sera détaillée dans le Chapitre 3.

1.3 Objectifs

Ce travail de thèse porte ainsi sur l'étude de la génération littéraire au niveau des phrases. Dans ce but, nous nous sommes fixés les objectifs spécifiques suivants :

1. Construire des corpus de documents littéraires en espagnol, en français et en portugais. Ces corpus doivent être d'une taille permettant l'étude de la langue et le développement d'algorithmes d'apprentissage.
2. Proposer une architecture pour le contrôle sémantique à partir d'un schéma interprétable par la machine.
3. Développer des modèles de production de phrases artificielles en espagnol et en français avec une structure syntaxique correcte, cohérente et capable d'être perçue comme littéraire, à partir d'éléments tels que le vocabulaire et la sémantique.
4. Proposer une stratégie de génération de phrases artificielles basée sur l'homosyntaxe, c'est-à-dire respectant la structure syntaxique d'un texte de base mais avec une sémantique différente.
5. Développer des modèles capables de produire des paragraphes en espagnol et en français.
6. Concevoir des protocoles d'évaluation pour mesurer la qualité du texte produit par nos systèmes.

1.4 Contributions

Au cours de nos travaux, diverses méthodes et approches ont été expérimentées et validées. Ces méthodes et les ressources développées sont énoncées ci-après.

— Les corpus **MegaLite-Es/Fr/Pt** (composés de textes littéraires en espagnol, français et portugais) ont été construits pour l'entraînement de nos modèles. Ces corpus contiennent un nombre similaire de documents.
— Nous avons construit le corpus LiSSS pour l'analyse d'émotions. Ce corpus contient des phrases littéraires en espagnol, classées dans les cinq émotions principales (Goldberg, 1990; Barrick et Mount, 1991).
— Un modèle pour la production de phrases littéraires artificielles a été développé. Il permet la génération de phrases par homosyntaxe en français

et en espagnol. Pour la sélection du vocabulaire, ce modèle est guidé par un contexte fourni par l'utilisateur.
— Un algorithme basé sur des représentations vectorielles et neuronales a été proposée pour l'analyse et l'interprétation sémantiques. Cette méthode permet d'établir des références sémantiques pour la génération de texte par homosyntaxe.
— Un modèle pour la production de couples de phrases rimées en espagnol. Ce modèle utilise des algorithmes précédents, ainsi que des ressources spécialisées comme le système RIMAX (Urrea et Torres-Moreno, 2019) pour la production des rimes sémantiques.

1.5 Plan de la thèse

Ce document de thèse est structuré comme suit : dans le Chapitre 2, nous détaillons les recherches menées dans l'état de l'art. Nous y étudions les travaux sur la génération automatique de textes, la génération de littérature, l'étude de corpus littéraires et non littéraires et leurs applications. Cette analyse nous permettra de comparer nos résultats en considérant les particularités entre les différentes propositions.

Dans le Chapitre 3, nous décrivons les corpus développées, **MegaLite** et **LiSSS**. Nous décrivons également des expériences comparatives entre le français et l'espagnol, réalisées au niveau syntaxique en utilisant les deux corpus.

Dans le Chapitre 4, les modèles proposés pour la génération de phrases littéraires sont présentés et détaillés. Des premières expériences utilisant des méthodes stochastiques à la mise en œuvre de méthodes de réinterprétation sémantique en utilisant Word2vec, modèles de langage, *shallow parsing* et approches de Recherche d'information.

Les expériences réalisées avec les modèles proposés sont présentées dans le Chapitre 5. Nous expliquons les ressources linguistiques utilisées pour l'entraînement de chaque modèle. Nous détaillons aussi les différents protocoles d'évaluation et les résultats obtenus.

Enfin, au Chapitre 6, les résultats obtenus lors des évaluations sont analysés et interprétés. Dans ce chapitre, nous tirons des conclusions pour chaque modèle et nous expliquons le processus logique qui mène à nos différentes propositions. Sur la base de nos conclusions, nous esquissons des perspectives nous semblant pertinentes pour des travaux à venir.

Chapitre 2

Créativité computationnelle

Sommaire

2.1 Analyse sémantique au moyen d'apprentissage profond . . . 19

2.1.1 Plongements lexicaux 19

2.1.2 Modèle *Skip-gram* 19

2.2 Génération Automatique de Texte (GAT) 21

2.2.1 Génération de texte non littéraire 21

2.2.2 Génération de texte littéraire 25

2.3 Émotions dans les textes 28

2.4 Corpus pour l'analyse littéraire ou émotionnelle 31

2.5 Conclusion 34

Dans son livre *The Creative Mind* Boden (2004) aborde le concept de créativité comme un phénomène dont les experts n'ont pas un accord universel pour décrire objectivement ce qu'est la créativité chez les humains. Selon Boden, pour qu'un artefact soit considéré comme le résultat d'un processus créatif, il doit être considéré nouveau et artistique. Étant donné que ces caractéristiques peuvent être observables, et donc mesurables, elles pourraient également être transposées au domaine du calculable.

Au sein de la CC, de nombreux chercheurs se sont attelés à modéliser le processus créatif longtemps considéré comme exclusif à l'homme. Dans ce chapitre nous allons présenter l'état de l'art de certaines travaux concernant la CC en lien directe avec les techniques et les algorithmes qui seront développés tout long de cette thèse.

2.1 Analyse sémantique au moyen d'apprentissage profond

Le défi de trouver une méthode efficace pour représenter un lexique dans un espace computationnel adéquat a été abordé depuis longtemps par de nombreux chercheurs (Bengio et al., 2013; Manning et Schütze, 1999; Spärck-Jones, 1972). Mikolov et al. (2013a) proposent un modèle de réseaux de neurones capable d'estimer la représentation numérique d'un lexique donné dans un espace vectoriel continu (représentation dense). C'est-à-dire, où chaque mot du lexique sera représenté par un vecteur numérique connu comme « plongement lexical » ou simplement «plongement», plus fréquemment trouvé dans la littérature comme *word embedding*.

2.1.1 Plongements lexicaux

Un plongement est la représentation numérique dense d'un mot au sein d'un lexique. Cette représentation est calculée sous forme vectorielle dont les dimensions dépendent de la taille du lexique analysé et de la structure du réseau de neurones utilisée. Les plongements sont capables de fournir des informations nécessaires sur la relation sémantique entre les mots du lexique étudié sous l'hypothèse que les mots apparaissant fréquemment dans des contextes similaires sont proches dans l'espace des plongements. (Goodfellow et al., 2014).

Plusieurs modèles basés sur la représentation numérique dense d'un vocabulaire ont été proposés dans la littérature (Bengio et al., 2013; Mikolov et Zweig, 2012). En fonction des objectifs fixés dans cette thèse, nous nous sommes concentrés en particulier sur le modèle *skip-gram*, que l'on décrit à continuation.

2.1.2 Modèle *Skip-gram*

Mikolov et al. (2013a) ont introduit l'architecture *Skip-gram*, une méthode d'implémentation du modèle Word2vec pour la génération des plongements qui nécessite des corpus de taille significative (Mikolov et al., 2013b). Cette architecture permet la prédiction d'un contexte étant donné un mot Q. Le contexte déterminé par le modèle est donné sous la forme d'un ensemble des mots, les plus proches à Q. L'architecture *Skip-gram* tente de maximiser la classification d'un mot en fonction des autres mots de la même phrase. Plus précisément, le modèle utilise le mot actuel comme entrée d'un classificateur log-linéaire avec une couche de projection continue, puis prédit les mots dans une certaine fenêtre autour du mot actuel. Mikolov et al. (2013a) ont constaté que l'élargissement de la fenêtre améliore la qualité des plongements résultants mais qui augmente également la complexité des calculs.

L'architecture du modèle est présentée dans la figure 2.1. Cette figure montre également l'architecture *bag-of-words* ou (CBOW). CBOW contrairement à *Skip-gram*, prédit un mot à partir d'un contexte donné, ce comportement n'obéit pas aux objectifs ne notre recherche. De ce fait, dans nos expériences nous avons utilisé exclusivement l'architecture *Skip-gram*.

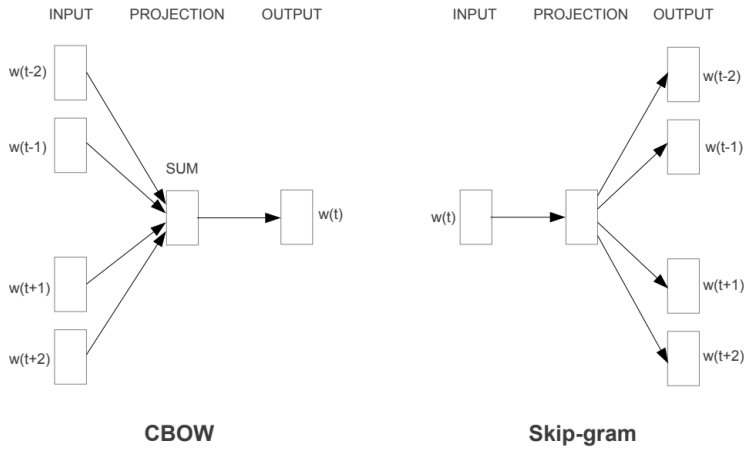


FIGURE 2.1 – *Schémas des implémentations de modèles Word2vec (CBOW et Skip-gram) (Mikolov et al., 2013a).*

Pour évaluer la qualité de *Skip-gram*, Mikolov et al. (2013c) ont proposé un protocole d'analyse différent de celui habituellement employé dans des travaux connexes, où l'évaluation consiste à étudier l'ensemble des mots les plus proches au terme Q (Bengio et al., 2013; Goodfellow et al., 2014). Il a été constaté qu'il y a différentes façons d'examiner la similarité entre les mots. Par exemple, quand on étudie le mot *big* et sa similarité avec *bigger*, on trouve une signification similaire que quand on analyse *small* et *smaller*. Dans cette logique, Mikolov propose une évaluation basée sur l'étude des paires de mots, dont la prémisse repose sur la question suivante : Quel est le mot le plus similaire à *small* dans le même sens que *biggest* est similaire à *big* ?

De façon assez surprenante, ce type de question trouve une réponse en effectuant de simples opérations algébriques dans l'espace des plongements. Par exemple, pour trouver un mot similaire à *small* dans le même sens que *bigger* est similaire à *big*, on peut simplement calculer le vecteur : $X = \overrightarrow{bigger} - \overrightarrow{big} + \overrightarrow{small}$. Ensuite, on cherche dans l'espace vectoriel le plongement le plus proche à X, mesuré avec une distance cosinus, et on l'utilise comme réponse à la question. Lorsque les plongements ont été bien entraînés, il est possible de trouver la réponse correcte (*smaller*) en utilisant cette méthode.

2.2 Génération Automatique de Texte (GAT)

L'écriture est une faculté conçue par et pour les humains, où plusieurs compétences sont mises en œuvre pour donner comme résultat un texte capable de transmettre des informations pertinentes. Le processus d'écriture requiert un ensemble de compétences linguistiques et de connaissances extra-linguistique que l'écrivain doit utiliser pour exprimer une idée (Sharples, 1996). C'est pourquoi il associe l'écriture au processus créatif de Boden.

2.2.1 Génération de texte non littéraire

Automatiser ce processus passionne la communauté scientifique qui s'est penchée sur la tâche de GAT au cours des dernières années. Szymanski et Ciota (2002) présentent un modèle basé sur des séquences ou chaînes de Markov pour la génération stochastique de texte. Les chaînes de Markov sont un processus stochastique qui sert à prédire l'état futur d'un système à partir de son état actuel. La figure 2.2 montre l'exemple d'un système de Markov à deux états A et E, ainsi que ses probabilités de transition. La transition de l'état A à l'état E peut se faire avec une probabilité de 0,4 ; et la transition inverse, d'E vers A, avec une probabilité de 0,7. Les probabilités de rester sur le même état A ou E sont respectivement de 0,6 et 0,3.

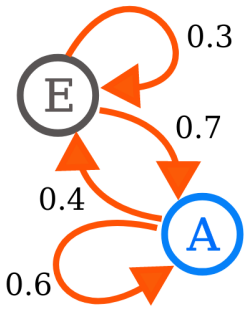


FIGURE 2.2 – *Illustration basique d'une chaîne de Markov (Source : `https://fr.wikipedia.org/wiki/Chaîne_de_Markov`).*

Szymanski et Ciota (2002) ont tenté de générer des mots syntaxiquement corrects en polonais à partir d'une analyse stochastique, avec un modèle de chaînes de Markov. Il faut dire que ces travaux ne tiennent pas compte du contexte ou du sens sémantique. Un état X_i, représente l'$i^{ième}$ caractère d'un mot dans la langue polonaise. L'algorithme utilise une matrice de transitions **M**, dont les éléments M_{ij} sont les probabilités d'occurrence de l'élément X_i, sous la condition que l'élément précèdent soit une séquence $X_{i-1}, X_{i-2}, X_{i-3}, \ldots, X_{i-k}$. Les expériences ont été effectuées en polonais, en utilisant 35 lettres qui représentent la totalité des états possibles. Cette méthode donne des bons résultats avec

des bigrammes, affichant une précision de 90%. Cependant, les performances tombent rapidement lorsqu'ils utilisent de tri-grammes ou de quadri-grammes.

Sridhara et al. (2010a) présentent un algorithme permettant de générer automatiquement des commentaires descriptifs de code Java. Leur modèle est composé de trois étapes (voir figure 2.3) :

1. Étape 1 : sélectionner le contenu (*s_units*) qui sera incrusté dans le commentaire descriptif. Ce contenu est sélectionné à partir de l'analyse du code. L'algorithme focalise sur les noms d'arguments, les valeurs retournées depuis les méthodes, les déclarations des variables, etc.
2. Étape 2 : générer du texte cohérent pour exprimer le contenu retenu dans la première étape. Le modèle génère des phrases isolées pour lesquelles chaque *s_unit* est associée à une action. L'action est déterminée selon le bloc de code où la *s_unit* a été trouvée ; par exemple les noms des méthodes, des fonctions ou des classes.
3. Étape 3 : fusionner les phrases afin de donner un résumé lisible et cohérent.

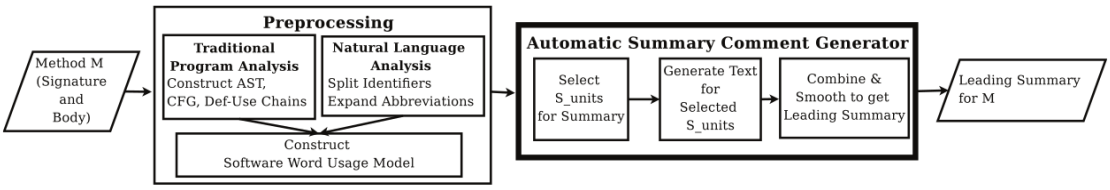


FIGURE 2.3 – *Schéma pour la génération des résumés descriptifs de code Java (Sridhara et al., 2010a).*

D'un autre côté, Lebret et al. (2016) proposent un algorithme pour la génération des mots à partir des données structurées. L'algorithme est basé sur un modèle de langue [1]. Pour produire un mot, l'algorithme utilise une chaîne de Markov. Chaque mot fait partie d'un vocabulaire contenu dans une table de faits (TdF). La TdF contient des entrées avec une séquence de mots et un contexte descriptif de cette séquence. Les auteurs ont enrichi cette table avec d'autres informations telles que le nombre d'occurrences et la position du document où cette séquence a été trouvée. Ces informations permettent d'établir un modèle de langue adéquat pour choisir le mot le plus cohérent afin de produire un texte. Pour évaluer le système, les auteurs ont créé le corpus **WIKIBIO** contenant 728 321 articles venant de Wikipédia, et plus spécifiquement du projet *WikiProject Biography* [2]. Un post traitement a été exécuté sur **WIKIBIO**, puis

1. Un modèle de langue est un algorithme qui capture les caractéristiques statistiques saillantes de la distribution des séquences de mots dans une langue naturelle, permettant de faire des prédictions probabilistes du mot suivant en fonction des précédents (Bengio, 2008).
2. `https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Biography`

80% du corpus a été utilisé pour entraîner le modèle et construire la TdF, le reste a été utilisé pour les tests. Les auteurs ont évalué ce système à l'aide des mesures BLEU [3] (Papineni et al., 2002) et ROUGE [4] (Lin, 2004) avec un score moyen de 34,7 et 25,8 respectivement. BLEU et ROUGE sont fréquemment utilisées pour comparer les modèles de Génération Automatique de Texte (GAT).

Des réseaux de neurones récurrentes comme les *Long short-term memory* (LSTM) (Hochreiter et Schmidhuber, 1997) ont aussi été utilisés dans la génération automatique de texte. Les LSTM sont des unités d'un réseau de neurones artificiel récurrent (RNN) utilisée dans l'apprentissage profond. Les RNN sont constitués d'unités (neurones), présentant des connexions récurrentes interconnectées par des synapses, interagissant non-linéairement pour qu'il y ait au moins un cycle dans la structure. Les RNN ont été utilisés en traitement automatique de la parole ou de texte (Oruh et al., 2022; Kumar et al., 2022; Alsayadi et al., 2022).

Le LSTM possède également des connexions de rétroaction. Il peut traiter non seulement des jeux de données uniques (comme des images) mais aussi des séquences entières de données (comme la parole ou la vidéo). Un réseau neuronal de type LSTM basé sur des analyseurs contextuels a été proposé pour la génération d'histoires fictives (Clark et al., 2018). Ce travail propose un mécanisme de génération textuelle au niveau de phrases guidé par trois informations :

— le contexte défini par la phrase précédente ;
— le contexte défini par la phrase actuelle ;
— le contexte général défini par le document en entier.

Le modèle identifie les entités nommées [5] (EN) utilisées dans le document, puis les associe à un des trois contextes évoqués. Ces associations vont donc permettre de générer une nouvelle narration avec un lien en commun parmi les EN détectées. Une évaluation manuelle a été effectuée en donnant au modèle un document avec 49 phrases (le contexte général), une phrase d'entrée (le contexte actuel) et un ensemble des phrases hors contexte pour essayer de tromper le modèle et vérifier son comportement. Le modèle devait choisir parmi ces

3. BLEU est une métrique axée sur la précision qui calcule le chevauchement des n-grammes des textes de référence et des textes générés. Ce chevauchement des n-grammes signifie que le schéma d'évaluation est indépendant de la position des mots, à l'exception des associations de termes des n-grammes.

4. Rouge est très similaire à la définition de BLEU, la différence étant que Rouge est axée sur le rappel alors que BLEU était axée sur la précision. Il existe 3 types de Rouge mais la plus courante est n-rouge, qui signifie un chevauchement de n-grammes, par exemple (2-rouge, 1-rouge pour les 2-grammes et 1-grammes respectivement).

5. Une entité nommée est une expression linguistique référentielle, souvent associée aux noms propres et aux descriptions définies.

phrases celles qui doivent poursuivre le texte du document. Les chercheurs ont comparé la phrase sélectionnée par le modèle avec celle sélectionnée manuellement. Ce modèle a obtenu une précision de 65%.

Kharazmi et Kharazmi (2017) proposent une analyse qui consiste à calculer, à partir d'une phrase, une matrice des plongements générés à l'aide d'un modèle Word2vec. Chaque plongement représente numériquement les mots de la phrase traitée. Cette matrice véhicule une information sémantique. Cette information, combinée avec un modèle de langue de n-grammes et une analyse stochastique, permet la création de phrases courtes et cohérentes. L'évaluation consiste à comparer manuellement la cohérence du texte d'entrée et des phrases générées. Les résultats montrent une haute précision après une évaluation manuelle, pour laquelle le modèle reçoit un ensemble de textes, modifiés manuellement, qui doivent être traités pour construire de nouvelles phrases.

Kiddon et al. (2016) proposent un modèle neuronal basé sur un réseau récurrent. Le réseau permet de prédire le meilleur contexte à partir d'une liste de mots-clés (liste de contrôle neuronal) à utiliser pendant la génération textuelle. Cette liste sera actualisée à chaque fois qu'un nouveau mot est introduit dans le texte. Le modèle a été évalué en générant du texte dans deux contextes : « Hôtel » et « Restaurant ». Pour mesurer la performance, les chercheurs ont employé BLEU et METEOR [6] (Denkowski et Lavie, 2014). Les résultats de l'évaluation pour le texte généré dans le contexte «Hôtel» a été : $BLEU = 90,61$ et $METEOR = 62,10$. Pour le contexte «Restaurant» a été : $BLEU = 77,82$ et $METEOR = 54,42$.

Il existe d'autres techniques pour la GAT basées sur des stratégies différentes. Molins et Lapalme (2015) proposent une méthode basée sur des schémas de représentations syntaxiques. Ces schémas contiennent l'information grammaticale du mot à produire afin qu'il soit cohérent avec le reste de la phrase. Des tests en français et en anglais ont été effectués sans aucune mesure de précision reportée mais avec des résultats encourageants selon les auteurs.

Finalement, Barros et al. (2017) proposent une approche supervisée visant la génération d'inflexions pour les verbes en espagnol. Leur système prend en entrée un verbe dans sa forme canonique (lemme) et les caractéristiques souhaitées telles que la personne, le nombre, le temps. Basé sur ces informations, leur système est capable de prédire la conjugaison grammaticale appropriée. L'algorithme met en œuvre des règles grammaticales afin de produire les flexions, pour tous les modes en espagnol (indicatif, subjonctif et impératif).

6. La métrique d'évaluation automatique METEOR note les hypothèses de traduction automatique en les alignant sur une ou plusieurs traductions de référence. Les alignements sont basés sur les correspondances exactes, les racines, les synonymes et les paraphrases entre les mots et les phrases.

Modèles	**Fluidité**		**Cohérence**		**Éloquence**		**Poétique**	
Caractères	*5*	*7*	*5*	*7*	*5*	*7*	*5*	*7*
RNNPG	4,01	3,44	3,18	3,12	3,20	3,02	2,80	2,68
Personnes	4,31	4,19	3,81	4,00	3,61	3,91	3,29	3,49

TABLE 2.1 – *Résultats de l'évaluation du modèle proposé par Zhang et Lapata (2014).*

Les algorithmes de génération de texte présentés au long de cette section n'ont aucune orientation littéraire. Ils sont utilisés pour produire du texte dans des domaines majoritairement techniques. Toutefois, à partir des années 1990, l'idée de générer des textes littéraires a commencé à être explorée, comme nous le verrons dans la section suivante.

2.2.2 Génération de texte littéraire

Les premières créations littéraires remontent à l'antiquité (Lecoq, 2021), et il est admis qu'il s'agit d'un domaine exclusif des êtres humains (Colton et al., 2012). Or, de nos jours, plusieurs recherches dans le domaine de la Créativité computationnelle, ont permis de générer du texte qui pourrait être considéré comme littéraire, tels que les poèmes, la narrative, les contes, etc.

Zhang et Lapata (2014) ont utilisé des RNN pour générer des poèmes. Leur modèle est basé sur deux prémisses : *quoi dire ?* et *comment le dire ?* Leur système a été entraîné sur un corpus contenant diverses associations entre des rôles et des faits. Ces éléments ont été combinés pour générer de nouvelles associations. Les auteurs ont également inclus la rime dans le modèle en établissant une structure semi-rigide du poème ce qui permet d'établir des éléments de rime entre des paires de vers. Un protocole d'évaluation manuelle a été proposé dans lequel ils ont demandé à 30 personnes de lire 30 poèmes. Ces poèmes ont été divisés en deux parties. La première faisant 5 mots et la seconde faisant 7 mots. Les critères évalués sont : la fluidité, la cohérence, l'éloquence, le niveau poétique. Les critères sont évalués sur une échelle allant de 0 à 5 avec 0=«très mauvais» et 5=«très bon». Le résultat de cette évaluation est présenté dans la table 2.1. Une comparaison entre les poèmes générés par le modèle évalué et ceux écrits par les humaines montre de bonnes performances.

Dans (Oliveira, 2012; Oliveira et Cardoso, 2015) est présenté le système PoeTryMe pour la génération des poèmes basé sur des *templates*. Les *templates* ou «cadres» sont généralement des structures syntaxiques fixes, qui peuvent être modifiées afin d'introduire un vocabulaire adéquat afin d'exprimer une idée souhaitée. La génération de texte au moyen des *templates* est aussi appelée méthode de texte en boîte ou *canned text*. L'avantage de cette méthode est surtout l'optimisation du temps, vu que les structures sont censées être grammaticale-

ment correctes. Les chercheurs peuvent ainsi se concentrer spécifiquement sur la sélection du vocabulaire que contiendra le texte. À ce propos, les auteurs utilisent un modèle d'analyse sémantique pour choisir le vocabulaire qui sera introduit dans le poème généré.

Un travail basé sur le concept de texte en boîte est présenté dans (Agirrezabal et al., 2013). La méthode est assez intéressante puisqu'à partir de l'analyse d'étiquettes POS[7] extraites de plusieurs corpus, un modèle de langue basé sur la probabilité d'apparition de chaque élément est généré. Cette méthode stochastique est utilisée pour générer des nouvelles structures et procéder ainsi au remplacement des étiquettes POS. Trois expériences ont été réalisées pour leur substitution. Dans la première, toutes les étiquettes des séquences POS sont remplacées par des mots qui respectent la grammaticalité de l'étiquette. Dans la deuxième expérience, seuls les adjectifs et les noms sont remplacés selon les mêmes conditions. Enfin, en troisième expérience, seuls les noms sont remplacés par des mots ayant un rapport syntaxique spécifique. C'est-à-dire, remplacer chaque mot de la séquence par son équivalent en hyponymes, antonymes, hyperonymes, etc. Le résultat est une nouvelle phrase ayant un message similaire avec des mots différents. Deux linguistes ont évalué 135 strophes dans un test de Turing. Les linguistes ont classé les strophes générées par les personnes et par la machine. Le test a donné de meilleurs résultats pour la troisième expérience avec une précision de 75%.

Riedl et Young (2006) modélisent le concept de créativité, défini par Boden, à partir de l'analyse et la formalisation des éléments créatifs. Ils proposent le modèle *Intent-driven Partial Order Causal Link* (IPOCL), basé sur la réalisation d'un schéma de génération d'histoires. Le schéma consiste en un ensemble de règles linguistiques qui établissent des connexions entre deux éléments (les noms) au moyen d'actions (les verbes) ou de définitions (les adjectifs). Ces connexions peuvent à leur tour dépendre de conditions spécifiques. Ce modèle permet de relier les règles linguistiques afin de construire un schéma adapté à la génération d'histoires. Pour cela, les auteurs ont considéré les éléments suivants : l'ensemble des concepts pertinents en fonction du contexte ; la langue dans laquelle les règles sont définies ; une fonction permettant d'analyser les règles linguistiques, ainsi que leur liaison aux éléments contextuels, et une fonction permettant de relier plusieurs règles linguistiques de manière cohérente et simple. Le modèle a été évalué avec un protocole basé sur des questions/réponses. Ce protocole est capable de générer un modèle cognitif qui contient les thèmes qui sont censés être aussi dans le récit généré par IPOCL. Le modèle cognitif et le récit généré par IPOCL ont été comparés. Les auteurs affirment que le récit généré par leur modèle est cohérent et contient essentiellement les thèmes déterminés par le modèle cognitif.

7. *Part-Of-Speech* tags ou étiquettes grammaticales.

Pérez y Pérez (2015) a présenté une étude concernant l'état de l'art de la CC. Plusieurs tentatives de génération automatique de textes littéraires y sont étudiées. Par exemple, le modèle « Through the park » (Montfort, 2008b) capable de générer des récits historiques en manipulant le rythme de la narration. Dans les œuvres « About So Many Things » (Montfort, 2008a) et « Taroko Gorge » (Montfort, 2009) sont présentés des textes générés automatiquement. Le premier travail montre des strophes de 4 lignes étroitement liées les unes aux autres. Cela se fait par le biais d'une analyse grammaticale qui établit des liens entre les entités de différentes lignes. La deuxième œuvre montre quelques poèmes courts générés automatiquement avec une structure plus complexe que celle des strophes. L'inconvénient de ces deux approches est l'utilisation d'une structure inflexible, qui génère des textes répétitifs avec une grammaticalité limitée.

Le projet MEXICA [8] essaie de modéliser la génération de récits. L'objectif de ce projet consiste à générer des récits complets en utilisant des textes de l'époque précolombienne (Pérez y Pérez, 2015). Pour cela, MEXICA exploite une base de connaissances donnée par l'utilisateur, qui est composée de récits modifiables manuellement. MEXICA analyse ces récits et détecte les liens entre les personnages, qui peuvent être par exemple, une relation amoureuse ou amicale entre deux personnages. Le système génère une nouvelle histoire en prenant un des liens analysés et en donnant une nouvelle continuité selon un ensemble de postconditions manuellement construit. Cette liste de postconditions indique par exemple que s'il existe un lien d'amour de deux hommes vers la même femme, la suite de l'histoire doit contenir des éléments négatifs. Un ensemble de nouveaux liens avec des sentiments négatifs doit être analysé et introduit dans l'histoire. Ce processus créatif appelé E-R *(Engaged and Reflexive)* (Sharples, 1996) est itéré jusqu'à ce que l'utilisateur le considère comme achevé et que le résultat corresponde aux attentes de sémantique et de cohérence.

Un autre travail considérant la génération de récits est celui de Gervás et al. (2015). Dans ce travail, l'objectif visé n'est pas la génération de texte mais la création des structures permettant la génération ou bien l'évaluation syntaxique d'un texte donné. Pour la génération de ces structures, un algorithme réutilise et combine de manière logique des entités comme des scénarios, des actions et des personnages. Les entités ont été extraites des récits ou romans provenant de divers écrivains. La méthode reçoit donc une requête donnée par l'utilisateur, ensuite elle retournera toutes les structures liées à cette requête. Quand deux structures peuvent être mélangées, l'algorithme combine les personnages et les actions afin de retourner à l'utilisateur une structure plus complexe au lieu de deux structures basiques. La structure retournée pourrait être modifiée pour générer un nouveau texte. Cependant, cette dernière partie n'est pas comprise dans cet article. Aucune évaluation n'a été fournie.

8. Voir `https://www.youtube.com/watch?v=4h_jRqkT67U`

2.3 Émotions dans les textes

Dans cette section, nous présentons quelques travaux concernant l'analyse d'émotions et des personnalités, afin d'étudier comment ces aspects peuvent influencer les émotions véhiculées par les auteurs lors de la rédaction de leurs œuvres.

Le modèle sentiGAN est composé de différents générateurs visant la production de texte ayant un contexte émotionnel (Ke et Xiaojun, 2018). Bien que n'étant pas un modèle de génération de texte littéraire, sentiGAN semble intéressant puisque le contexte émotionnel est très important pour générer ce genre de texte. Il s'agit d'une mise à jour du modèle *Generative Adversarial Net* (GAN) qui a donné des résultats encourageants dans la génération de textes, bien que présentant certains problèmes de qualité et de cohérence (Goodfellow et al., 2014). sentiGAN utilise l'analyse sémantique des données fournies par l'utilisateur pour créer un contexte. L'idée principale suggère d'établir un nombre défini de générateurs de texte produisant un texte lié à une émotion préalablement définie. Pour leur évaluation, les chercheurs ont généré 5 000 phrases négatives et 5 000 phrases positives dans le sens émotionnel. Pour chaque paire de phrases, trois ensembles de données de critiques ont été utilisés comme corpus d'entraînement : critiques de films (MR), critiques de bières (BR) et critiques de clients (CR). La première évaluation est effectuée avec un modèle de classification de sentiments (Hu et al., 2016). Les chercheurs ont comparé la cohérence entre l'émotion indiquée pour le modèle et l'émotion hypothétique dans la phrase. La précision rapportée était : MR=88% ; BR=84% ; CR=80%.

Les générateurs sont entraînés suivant deux schémas :
— un ensemble d'éléments linguistiques à éviter pour générer le texte ;
— un ensemble d'éléments liés à l'émotion du générateur.
Grâce aux calculs de distance, des heuristiques et des modèles probabilistes, le générateur crée un texte aussi éloigné que possible du premier schéma et aussi proche que possible du second.

Des nouvelles recherches, ont étudié la relation entre certaines caractéristiques du texte littéraire et la personnalité des auteurs. Cette relation peut être comprise comme l'ensemble d'attributs comportementaux, tempéramentaux, émotionnels et mentaux qui caractérisent un individu (Wedemann et Plastino, 2016; Siddiqui et al., 2018). Au moyen de réseaux de neurones convolutifs (CNN), Majumder et al. (2017) ont proposé un modèle d'extraction des caractéristiques de la personnalité à partir de documents textuels. Cinq réseaux de neurones avec la même architecture ont été proposés mais entraînés pour traiter individuellement chacune des personnalités suivantes : Extraversion (EXT), Neuroticisme (NEU), Agréabilité (AGR), Conscience (CONS), Ouverture (OUV). Leur modèle fonctionne suivant cinq étapes :

Modèle CNN	**EXT**	**NEU**	**AGR**	**CONS**	**OUV**
P en %	58,09	59,38	56,71	57,30	62,68

TABLE 2.2 – *Précision obtenue par le modèle proposé par Majumder et al. (2017).*

1. Prétraitement : les documents sont découpés en phrases standardisées en minuscules.
2. Extraction des caractéristiques : les caractéristiques globales sont extraites du document (nombre de mots, longueur moyenne par phrase, etc.).
3. Filtrage : les phrases n'apportant aucun contenu sémantique sont filtrées car elles pourraient générer du bruit et de mauvais résultats.
4. Extraction de caractéristiques au niveau des mots : les mots, représentés par leurs plongements, génèrent une représentation continue des phrases dans un espace à $n-$dimensions.
5. Classification : les plongements sont employés pour entraîner un réseau convolutionnel CNN afin d'effectuer une classification.

Les auteurs ont évalué leur modèle à l'aide d'un corpus de styles linguistiques (Pennebaker et King, 1999), habituellement utilisé pour comparer les techniques de détection d'émotions. Les valeurs de précision obtenues pour chaque émotion sont reportées dans la table 2.2.

Une méthode très utilisée en analyse d'émotions est la logique floue. Cette méthode permet de classer un élément dans plusieurs catégories avec un degré d'appartenance au lieu d'une classification exclusive. Nadali et al. (2010) étudient l'importance des méthodes floues pour l'analyse des sentiments. Ils ont trouvé que contrairement aux méthodes de classification classiques, où les classes sont communément définies comme négative, neutre et positive, les classifieurs flous permettent l'utilisation de classes plus adaptées qui définissent mieux l'élément analysé. Suivant cette idée, Matiko et al. (2014) présentent un algorithme de classification floue d'émotions positives et négatives à partir d'électroencéphalogrammes.

L'analyse d'émotions est une tache très étudiée en TAL. Par exemple, Tashtoush et Al Aziz Orabi (2019) proposent un modèle de logique floue pour l'analyse de sentiments. Leur modèle a été entraîné avec un corpus de trois millions de tweets concernant les avis de produits sur internet. Les tweets ont été étiquetés selon l'émotion perçue en huit catégories : { Joie, Tristesse, Colère, Dégoût, Confiance, Peur, Surprise, Anticipation }. Pour l'étape de *fuzzification* [9], les

9. La fuzzification est le processus de conversion d'une valeur d'entrée nette en une valeur floue qui s'effectue par l'utilisation des informations de la base de connaissances. Bien que différents types de courbes puissent être observées dans la littérature, les gaussiennes, triangulaires

mots et les émoticons ont été considérés comme des valeurs linguistiques et traités séparément sous une échelle d'appartenance : basse, moyenne et élevée. Le processus de *fuzzification* se déroule d'une manière simple, en considérant des règles floues basiques du type *if-then*. Par exemple, le tweet : « *J'ai besoin d'aide avec mon compte, il est désactivé à chaque fois que j'essaie de me connecter et ça devient vraiment frustrant !*» sera classé comme « tristesse moyenne », avec un degré de 0,174 et comme « tristesse élevée » avec un degré de $0,826$. Également ce tweet sera classé comme « colère élevée » avec un degré de 0,375 et comme « colère très élevée » avec un degré de $0,625$.

Des travaux concernant l'analyse de polarités émotionnelles ont été proposés pour classer les avis de produits dans les catégories négative, neutre et positive, avec des résultats intéressants (Indhuja et Reghu, 2014). Dragoni et al. (2015) proposent un modèle basé sur la logique floue pour modéliser et analyser les polarités des concepts. Pour entraîner leur modèle, ils ont utilisé les ressources WordNet [10] et SenticNet [11]. Ainsi, les adjectifs, verbes, et noms extraits de WordNet sont associés aux différents concepts de SenticNet pour construire des règles floues.

D'autres travaux abordent la problématique de la détection d'émotions des agents cognitifs avec la logique floue. Dans (Howells et Ertugan, 2017), les tweets sont traités avec la logique floue et classés comme : {très négatif, négatif, neutre, positif, très positif}, en analysant les émoticons, les hashtags et le texte. Arguedas et al. (2018) proposent un classifieur flou pour détecter les états émotionnels des étudiants. Leur modèle analyse des textes rédigés par des étudiants, dont chaque mot est associé automatiquement à une émotion récupérée du dictionnaire ANEW (Bradley et Lang, 1999). Ensuite, en utilisant Emolex (Mohammad et Turney, 2013) le modèle calcule la charge émotionnelle de chaque association. Une association sera considérée comme une caractéristique qui sera envoyée au *Fuzzy Based Classification Model* (FBCM) [12] qui pourra prédire l'état affectif de l'étudiant selon le texte analysé.

Les techniques de logique floue ont été couramment utilisées pour analyser les commentaires sur les produits ou services proposés sur Internet. Vashishtha et Susan (2020) proposent un modèle de classification des produits à base de règles floues. Leur modèle analyse les critiques vidéo des clients. La mesure TF-IDF [13] est utilisée pour pondérer les mots présents. Pour les caractéristique

et trapézoïdales sont les plus couramment utilisées dans le processus de fuzzification (Kayacan et Khanesar, 2016).

10. `https://wordnet.princeton.edu/`

11. `http://sentic.net/`

12. Le modèle FBCM a été mis en œuvre et breveté par Marta Arguedas sous le nom de FUZZYEMOSYS.

13. TF-IDF est une mesure statistique qui combine le nombre d'occurrences et la rareté d'un

phonétiques, les auteurs utilisent le système openSMILE, qui extrait des caractéristiques telles que l'intensité ou la hauteur de la voix. Un total de 6 373 caractéristiques ont été extraites pour chaque audio analysé. Puis, elles ont été analysées par le modèle de logique floue avec une fonction d'appartenance triangulaire pour obtenir le degré de polarité de chaque commentaire. L'approche proposée donne une précision de 82, 6%, un rappel de 82, 3%, et un F-score de 84, 4%.

Nadali et al. (2010) proposent un modèle qui permet de classer les critiques de produits en analysant les commentaires au niveau des phrases et des mots. Leur modèle met l'accent sur les verbes, les adjectifs et les noms mais inclut également les adverbes comme caractéristique auxiliaire. Ils proposent une méthode, basée sur des fonctions d'appartenance floues, conçues expérimentalement pour calculer la force de la subjectivité dans une phrase, en analysant les adverbes.

Nous constatons ainsi que les méthodes floues sont efficaces pour l'analyse des sentiments et des émotions. Dans nos expériences, nous avons fait appel à ces techniques pour analyser la charge émotionnelle véhiculée par les phrases générées par nos systèmes. L'objectif étant de déterminer dans quelle mesure on peut générer des phrases transmettant des émotions proches de celles de l'œuvre d'un auteur, en plus d'être guidées par un contexte. Nos modèles flous et nos expériences seront expliqués au Chapitre 4.

2.4 Corpus pour l'analyse littéraire ou émotionnelle

Les corpus linguistiques ont toujours été utilisés dans les tâches de TAL (Martínez, 2018). Que ce soit pour une recherche exploratoire ou pour une expérimentation, la constitution des corpus est une tâche complexe qui a souvent été sous-estimée. On constate qu'un corpus bien construit et correctement analysé peut être exploité pour améliorer les résultats des recherches en TAL et en particulier en Génération Automatique de Texte.

Par exemple, Yang et Coxhead (2020) ont construit un corpus contenant un vocabulaire classé, extrait de l'analyse des cahiers d'étudiants en anglais. Le classement a été effectué à partir d'un modèle de langue en considérant la fréquence de chaque mot. Ce corpus a mis en évidence l'importance du vocabulaire employé par les étudiants et de l'impact que cela pourrait avoir dans leur formation.

La construction des corpus bilingues permet d'étudier et de comparer les caractéristiques des langues de façon très intéressante. Bourgonje et al. (2017)

mot dans un document (Spärck-Jones, 1972).

Corpus	Langue	Genre	Citations
CQSAC	Anglais	Littérature	3176
PARC 3	Anglais	Journalistique	19712
STOP	Anglais	Fiction, Biographique, Journalistique	13237
ACDS	Anglais	Littérature (Bible)	1245
RWG	Allemand	Narrative, Journalistique	9451

TABLE 2.3 – *Caractéristiques de corpus de citations analysé par Papay et Padó (2020).*

présentent un corpus aligné [14] allemand-italien. Les corpus alignés sont une première étape pour la construction d'un véritable corpus comparables au niveau du discours. Les corpus comparables sont composés d'au moins deux ensembles de textes possédant des caractéristiques communes. Dans les faits, les corpus comparables peuvent relever de la même langue mais nous ne nous y intéresserons que dans le cas où il s'agit d'ensembles de textes rédigés dans au moins deux langues différentes. Ils se distinguent des corpus alignés, car les textes qui les composent ne constituent pas des traductions, ni dans la première, ni dans la seconde langue (Claude, 2004).

Dans le domaine du TAL, on peut trouver une grande variété de corpus linguistiques mais, ils ont été construits en grande partie à partir de textes scientifiques, techniques, journalistiques, de réseaux sociaux, de blogs et d'autres. Cependant, dans le domaine de la CC, de tels corpus ne répondent pas forcément aux besoins requis. Papay et Padó (2020) ont mené une étude comparative entre des corpus littéraires et non littéraires. Ils ont montré que le genre textuel littéraire a été le moins exploité parmi les autres genres. Dans la table 2.3, on observe les corpus analysés, la langue, le genre et le nombre de citations pour chaque corpus.

Lorsque l'on aborde la génération textuelle à partir d'une approche de CC, il est souvent prévu d'analyser des textes littéraires (de la poésie, des romans, des nouvelles, etc.), dans lesquels, le niveau de langue et le style de rédaction peuvent être très différents de ceux que l'on trouve dans les corpus évoqués précédemment. C'est pourquoi nous pensons qu'il est important de consacrer des efforts à la construction des corpus adaptés au domaine littéraire. Nous émettons l'hypothèse que la richesse et la variabilité des styles littéraires peuvent guider, jusqu'à un certain point les algorithmes de génération de texte, tout en évitant les styles trop rigides des documents techniques, ou les stéréotypes du style journalistique. Également, dans cette thèse nous cherchons à éviter de reproduire les usages adoptés dans le cadre des réseaux sociaux. Nous présentons

14. Les corpus alignés réunissent des textes de plusieurs langues dont une partie constitue la traduction de l'autre. Leur réalisation repose sur l'établissement (généralement automatique) de correspondances entre les composantes formelles des textes (Claude, 2004).

ci-après quelques travaux proposant de nouveaux corpus pour l'analyse de documents littéraires.

Papay et Padó (2020) proposent le corpus RiQua composé de structures de citations littéraires en anglais du 19ème siècle. L'intérêt de RiQua est d'étudier les éléments grammaticaux dans un texte, et de voir l'importance de la relation entre le contexte et le contenu. Pour chaque citation, des informations interpersonnelles telles que les noms des locuteurs et des destinataires ont été incluses, fournissant une vue riche de la structure du dialogue [15].

Stymne et Östman (2020) décrivent le corpus SLäNDa, constitué de 44 chapitres de narrations en suédois avec plus de 220K tokens annotés manuellement. L'annotation a permis d'identifier 4 733 occurrences de matériel cité (citations et signes de ponctuation) qui sont séparés du récit principal, et 1 143 correspondances nommées entre locuteur et tour de parole. Ce corpus a été utile pour le développement d'outils informatiques pour analyser la narration littéraire et le discours.

En ce qui concerne les corpus ayant un contenu émotionnel nous présentons quelques exemples intéressants. Le corpus SAB (Navas-Loro et al., 2017) n'est pas un corpus littéraire, il a été construit pour l'analyse et la détection des émotions. Il est composé de tweets en espagnol des critiques concernant 7 types de produits commerciaux (alimentation, automobile, banque, boissons, sports, vente au détail, télécommunications). Ce corpus est composé de 4 548 tweets annotés en utilisant 8 émotions prédéfinies : confiance, satisfaction, bonheur, amour, peur, désaffection, tristesse, colère. L'annotation a été faite en considérant l'émotion perçue dans chaque tweet. Nous trouvons cette étude très intéressante car elle utilise une approche émotionnelle dans l'annotation. La distribution des critiques par émotions est montrée dans la table 2.4.

TASS est un autre corpus composé de tweets (Villena-Román et al., 2013). Il contient environ 70 000 tweets en espagnol récupérés à partir de comptes de 200 célébrités venant de plusieurs domaines (politique, économique, culturel, technologique, musical, sportif, etc.). Le corpus TASS a été annoté manuellement selon deux critères :

— l'émotion perçue dans le tweet : *positive, négative, neutre* ;
— l'intensité de la polarité par rapport à l'émotion remarquée : *très positive, positive, négative, très négative*.

Chen et Skiena (2014) présentent un travail très complet où trois ressources sont proposées. La première est un lexique émotionnel composé de mots issues des 136 langues les plus parlées au monde [16]. La deuxième ressource est un

15. Corpus disponible dans le site officiel de l'Université Stuttgart `https://www.ims.uni-stuttgart.de/en/research/resources/corpora/riqua/`
16. `https://sites.google.com/site/datascienceslab/projects/`

Secteur	COL	TRIS	PEUR	DESS	SATI	CONF	BONH	AMR
ALIMENTA	1,36	1,09	0	7,63	41,69	40,60	13,62	11,99
AUTOMOB	0	0,18	0,91	2,0	5,99	3,09	1,09	0,91
BANQUE	4,49	0,84	12,92	20,51	1,12	0,42	0	0
BOISSON	2,19	1,17	0,73	19,10	44,02	32,80	7,43	7,73
SPORT	2,45	2,60	0,31	13,32	18,84	11,94	4,90	11,33
DÉTAIL	2,03	0,75	0,98	7,53	8,96	8,89	2,33	2,11
TÉLÉCOM	12,85	0,80	0	29,32	8,43	6,02	3,21	1,20

TABLE 2.4 – *Pourcentage de critiques par secteur et par émotion (Navas-Loro et al., 2017). COL=Colère, TRIS=Tristesse, PEUR=Peur, DESS=Désaffection, SATI=Satisfaction, CONF=Confiance, BONH=Bonheur et AMR=Amour.*

graphe de connaissances qui comprend 7 millions de mots, avec environ 131 millions de liens sémantiques inter-lingues. Finalement, les auteurs ont mené une analyse pour étudier la cohérence émotionnelle exprimée dans Wikipédia en 30 langues, concernant des personnages historiques.

L'étude et la composition de corpus est un sujet qui gagne du terrain dans plusieurs domaines scientifiques, grâce à l'augmentation de l'usage des méthodes d'apprentissage profond. Étant donné la nécessité d'une grande quantité de données, et le manque de corpus littéraires dans des langues autres que l'anglais, nous avons décidé de dédier une partie de notre travail de recherche au développement et la diffusion de ressources linguistiques et littéraires en langues romanes. Ces ressources seront détaillées dans le chapitre suivant.

2.5 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art, basé sur des travaux dans trois domaines directement liés à nos objectifs : la génération automatique de textes, l'analyse des émotions et la construction de corpus. Ces travaux décrivent plusieurs problèmes, tels que la complexité de l'établissement d'un protocole d'évaluation automatique dans les modèles basés sur la CC, ou l'absence de corpus linguistiques littéraires dans des langues autres que l'anglais. Ces corpus sont très utiles pour les tâches d'apprentissage profond. Dans ces travaux sont également expliqués les méthodes qui nous ont inspiré et servi de base pour relever les défis de nos objectifs. Certains articles présentent des modèles basés sur des réseaux neuronaux profonds, ce que nous avons décidé d'éviter, avec la motivation de préserver autant de contrôle que possible pendant tout le déroulement de nos expériences. Dans les deux chapitres suivants, nous décrivons la méthodologie utilisée pour résoudre les points cruciaux de notre projet de recherche, sur la base de quelques éléments remarquables de l'état de l'art.

Chapitre 3

Construction de corpus littéraires

Sommaire

3.1 Description des corpus MegaLite 36
3.1.1 Corpus MegaLite-Es 36
3.1.2 Corpus MegaLite-Fr 37
3.1.3 Corpus MegaLite-Pt 37
3.1.4 Prétraitements du corpus MegaLite 38
3.2 Versions alternatives 39
3.2.1 Modèle de représentation continue (Word2vec) 40
3.2.2 Version étiquettes grammaticales (POS) et version lemmatisée 42
3.2.3 Modèle de langue *n*-grammes 44
3.3 Corpus littéraire d'émotions LiSSS 45
3.3.1 Annotation d'émotions 46
3.3.2 Stratégie de vote démocratique 48
3.4 Application des corpus littéraires : une analyse stylistique statistique 49
3.4.1 Divergence de distribution de probabilités appliquée aux styles 51
3.5 Conclusion 52

Dans le domaine de la Créativité computationnelle (CC) de nombreuse recherches ont abordé la génération d'artefacts artistiques. De nos jours on dispose d'algorithmes capables de générer de la peinture, de la musique et des œuvres littéraires de genres telles que la poésie et la narrative (Minu et al., 2022; Ramesh et al., 2022). Cependant, un problème très fréquent de la CC est l'absence de corpus suffisamment riches, utiles pour les processus d'entraînement et de test des modèles proposés. L'étude et la constitution de corpus littéraires

sont très importantes pour le développement et l'évaluation de modèles de CC. Or, ces études ont été systématiquement laissées de côté principalement en raison du niveau de la complexité du discours littéraire, de la subjectivité et de l'ambiguïté présentes dans ces textes. Ainsi, l'utilisation de corpus constitués de documents encyclopédiques (principalement Wikipédia), journalistiques (journaux, magazines) ou de documents spécialisés (textes juridiques, scientifiques ou techniques) a été longtemps privilégiée (Torres-Moreno, 2014; da Cunha et al., 2011; Martínez, 2018). Pour palier ce problème, dans ce chapitre nous décrivons les différents corpus que nous avons construits pour l'entraînement et la validation de nos modèles.

3.1 Description des corpus MegaLite

Dans le cadre de cette thèse et afin d'entraîner et de tester les modèles proposés, nous avons créé des corpus contenant des milliers de documents littéraires en espagnol, français et portugais. Nous nous sommes focalisés sur ces trois langues pour deux raisons : d'abord car les langues romanes possèdent une grande richesse lexicale, une expressivité importante et un haut niveau de complexité morphologique. Également, nous pensons que beaucoup d'études et des ressources ciblent l'anglais. De plus, nous considérons que certaines caractéristiques comme la richesse lexicale, la morphologie et l'expressivité, sont présentes en moindre proportion en anglais. En effet, cette langue a été très exploitée dans les recherches en TAL et en CC à cause –entre autres– de sa moindre complexité grammaticale. Or malgré l'origine commune du français, du portugais et de l'espagnol en tant que langues romanes (ce qui explique en partie leur proximité), leurs structures linguistiques sont très différentes, surtout lorsqu'elles sont analysées au niveau littéraire (voir Section 3.4). Ceci est un autre point intéressant pour les étudier.

3.1.1 Corpus MegaLite-Es

Nous avons constitué le corpus littéraire en espagnol **MegaLite-Es**, constitué de 5 075 documents littéraires (principalement des livres) en espagnol, venant de 1 336 auteurs hispanophones et de traductions officielles d'auteurs dans de langues autres que l'espagnol. Quelques statistiques de ce corpus sont rapportées dans la table 3.1. D'autres informations supplémentaires sont montrées dans l'annexe A.

Les genres et leur distribution sont présentés dans la table 3.2. Le genre narratif est le genre le plus populaire du corpus ($\approx$ 92% des œuvres).

	Documents	Phrases	Mots	Auteurs
MegaLite-Es	5 075	14,7 M	211 M	1 336
Moyenne par document	-	3 K	41,8 K	-

TABLE 3.1 – *Caractéristiques du corpus **MegaLite-Es**.*

MegaLite	Théâtre	Poésie	Narratif
Espagnol	247 (4,9%)	138 (2,7%)	4 690 (92,4%)

TABLE 3.2 – *Distribution des documents par genres du corpus **MegaLite-Es**.*

3.1.2 Corpus MegaLite-Fr

Dans le même esprit que la version en espagnol, le corpus **MegaLite-Fr** est constitué de 2 690 documents littéraires en français venant de 620 auteurs. Dans la plupart des cas, les documents ont été écrits originellement en français. Une partie importante de ce corpus vient du site Bibebook [1], disponible sous la licence du domaine public *Creative Commons BY-SA* [2]. Quelques statistiques de **MegaLite-Fr** sont présentées dans la table 3.3. La répartition des œuvres par genre est présentée dans la table 3.4. D'autres informations supplémentaire sont également montrées dans l'annexe A.

	Documents	Phrases	Mots	Auteurs
MegaLite-Fr	2 690	9,7 M	182 M	620
Moyenne par document	-	3,6 K	67,9 K	-

TABLE 3.3 – *Caractéristiques du corpus **MegaLite-Fr**.*

MegaLite	Théâtre	Poésie	Narratif
Français	97 (3,6%)	55 (2,0%)	2 538 (94,3%)

TABLE 3.4 – *Distribution des documents par genres du corpus **MegaLite-Fr**.*

3.1.3 Corpus MegaLite-Pt

Finalement, et en suivant la même philosophie que les corpus **MegaLite-Es** et **MegaLite-Fr**, nous avons constitué une version portugaise. **MegaLite-Pt** est

1. `http://www.bibebook.com`
2. `http://creativecommons.org/licenses/by-sa/3.0/fr`

constitué de 4 311 documents littéraires en portugais venant de 1 419 auteurs. Quelques statistiques de **MegaLite-Pt** sont présentées dans les tables 3.5 et 3.6. Des statistiques plus détaillées de **MegaLite-Pt** sont présentées dans les tables de l'Annexe A. [3]

	Documents	**Phrases**	**Mots**	**Auteurs**
MegaLite-Pt	4 311	21 M	252 M	1 419
Moyenne par document	-	4,9 K	58.5 K	-

TABLE *3.5* – *Caractéristiques du corpus **MegaLite-Pt**.*

MegaLite	**Théâtre**	**Poésie**	**Narratif**
Portugais	352 (8,1%)	1218 (28,2%)	2 739 (63,5%)

TABLE *3.6* – *Distribution des documents par genres du corpus **MegaLite-Pt**.*

Le nombre de mots du corpus **MegaLite** trilingue, est de plus de 645 millions de mots et de 45,4 millions de phrases. Cette taille lui rend particulièrement intéressant pour des tâches d'apprentissage automatique, comme on verra dans les chapitres suivants.

3.1.4 Prétraitements du corpus MegaLite

Les textes originaux venant de plusieurs formats hétérogènes [4] ont été prétraités afin de créer un ensemble de documents standardisés dans un format texte Unicode *utf8*. Ces documents contiennent une grande quantité d'erreurs (mots tronqués, concaténés ou scindés, des symboles étranges et disposition inhabituelle des paragraphes). Ces erreurs assez fréquentes se retrouvent généralement dans des corpus de dimensions et des sources similaires.

Afin de minimiser ces inconvénients, un processus de segmentation et de filtrage minutieux a été effectué pour découper les textes en unités textuelles adéquates (les phrases). Pour cela, nous avons développé un outil en PERL 5.0 avec des expressions régulières appropriées. Ce traitement élimine également un certain nombre de phrases et de tokens indésirables non informatifs, tels que les tampons éditoriaux, les remarques, les tables des matières, les résumés, les copyrights, les numéros de page et de chapitre, entre autres. Ces processus

3. Le corpus en portugais ne sera pas utilisé dans les expériences de génération de phrases dans le cadre de cette thèse.

4. Principalement des documents en formats pdf, txt, html, doc, docx et odt, parmi d'autres.

sont difficilement réalisables avec des outils d'apprentissage automatique, c'est pourquoi nous nous sommes repliés sur des méthodes à base de règles heuristiques (Eensoo et Valette, 2015).

Les corpus **MegaLite** ont l'avantage d'être d'une taille considérable et de convenir ainsi aux développement des modèles TAL ; soit pour l'apprentissage automatique, soit pour la génération automatique de textes soit pour d'autres tâches. Ils conviennent également pour tester la qualité et les performances de tels algorithmes. **MegaLite** présent toutefois l'inconvénient que les phrases ne sont pas toutes forcement considérées comme des *phrases littéraires*. Beaucoup d'entre elles sont des phrases auxiliaires contenant souvent un vocabulaire général. Tel est le cas des dialogues, par exemple. Cependant, les phrases auxiliaires permettent une lecture fluide et fournissent les liens nécessaires entre les idées exprimées parmi les phrases littéraires. Nous avons donc décidé de les garder.

Afin de faciliter la gestion du corpus **MegaLite**, nous avons établi le renommage des documents selon la convention suivante :

`Nom_d'Auteur,_Prénom-Titre_de_l'oeuvre=GENRE.txt`

La balise `GENRE` peut prendre les valeurs suivantes :
— `TEATRO/THÉÂTRE`
— `POESIA/POÉSIE`
— `NARRATIVA/NARRATIF`

3.2 Versions alternatives

En raison des droits d'auteur de plusieurs œuvres, le corpus **MegaLite** ne peut pas être entièrement distribué tel quel. Néanmoins, pour éviter cet inconvénient, nous les diffusons sous plusieurs formats alternatifs [5]. Les différentes versions du corpus **MegaLite** sont brièvement expliquées ci-dessous et plus en détail dans la Section 3.4.

— Version I (plongements) : des tableaux des plongements, utiles pour l'analyse sémantique, obtenus avec l'entraînement du modèle Word2vec.
— Version II (POS) : les mots lexicaux, c'est-à-dire, les adjectifs, les noms, les verbes et les adverbes, qui sont sont les éléments les plus représentatifs d'un texte (Bracewell et al., 2005), ont été remplacés par leurs étiquettes grammaticales (POS) correspondantes.
— Version III (lemmes) : les mots lexicaux sont remplacés par leurs lemmes [6]

5. Corpus disponibles dans le site : `http://juanmanuel.torres.free.fr/corpus/megalite`
6. Forme canonisée d'un mot.

correspondants. Cette version et la version II sont très utiles pour des analyses morphologiques.
— Version IV (*n*-grammes) : un ensemble de tableaux contenants les fréquences des *n*-grammes pour chaque document a été construit. Pour générer ces tableaux, chaque corpus a été traité séparément. Les *n*-grammes calculés sont : unigrammes, bigrammes, et SU4-bigrammes (Lin, 2004; Cabrera-Diego et Torres-Moreno, 2018).

3.2.1 Modèle de représentation continue (Word2vec)

Un plongement vise à quantifier et à catégoriser les proximités sémantiques entre les éléments linguistiques (les mots) sur la base de leurs propriétés d'occurrence et de distribution dans des grands corpus de données linguistiques. Nous avons évité à tout prix d'utiliser des plongements pré-entraînés, car rien ne garantit que les corpus utilisés pour leur apprentissage soient littéraires. Nous avons donc décidé de produire nos propres plongements.

Pour générer les plongements à partir du corpus **MegaLite**, nous avons entraîné un modèle Word2vec, avec l'architecture *Skip-gram* (Mikolov et al., 2013c) en utilisant la bibliothèque de Python Gensim (Řehůřek et Sojka, 2010). Comme résultat de ce processus, nous avons produit deux tables contenant 420 757 plongements du corpus **MegaLite-Es**, et 171 544 plongements du corpus **MegaLite-Fr**.

Les paramètres utilisés pour entraîner le modèle Word2vec sont décrits ci-dessous. Les valeurs configurées pour ces paramètres sont affichées dans la table 3.7.
— *Iterations* : désigne le nombre d'époques d'apprentissage ;
— *Minimal count* : indique le nombre minimal d'occurrences qu'un mot doit avoir dans le corpus pour être inclus dans le vocabulaire du modèle ;
— *Vector size* : spécifie la dimension des plongements lexicaux ;
— *Window size* : représente la taille de la fenêtre des mots qui seront liés au mot analysé, lors de l'apprentissage du modèle.

Paramètres	**Valeurs**
Iterations	1
Minimal count	3
Vector size	60
Window size	5

TABLE 3.7 – *Paramètres d'entraînement du modèle Word2vec.*

Les valeurs pour les paramètres listés ci-dessus ont été déterminées en respectant l'étude effectuée par Mikolov et al. (2013a). Les chercheurs ont trouvé

que pour les corpus textuels de dimensions supérieures à 2Go, les paramètres comme *vector size* ou *windows size* peuvent être fixés jusqu'à 200 et 10 respectivement, ceci afin de capter le plus possible la charge sémantique dans le texte.

Dans la table 3.8, on peut observer trois requêtes en espagnol (en gras) et leurs mots les plus proches, selon les similarités cosinus[7] calculées par notre modèle Word2vec à partir des plongements. Dans la première colonne, on peut remarquer que la requête **bleu** retourne des mots tels que vert, violet, jaune (*verde, violeta, amarillo*) parmi d'autres couleurs. Étant donné que les plongements sont des vecteurs numériques, diverses opérations mathématiques peuvent être effectuées entre eux, par exemple : [*'bleu'*] + [*'océan'*]) - [*'violet'*]. Le résultat est un vecteur numérique, à partir duquel – en calculant la similarité cosinus – on peut récupérer les plongements les plus proches.

Ces propriétés calculatoires des plongements sont très puissantes, car avec une analyse mathématique appropriée, différents champs sémantiques peuvent être déduits. La bibliothèque Gensim implémente déjà plusieurs fonctions pour effectuer ce type d'opérations mathématiques entre les plongements. En continuant avec l'exemple précèdent, nous avons :

(([*'bleu'*] + [*'océan'*]) - [*'violet'*]) $\rightarrow$ {*mer, Pacifique, Atlantique, atoll, lac, récif, Méditerranée, Arctique, Océan, marécage*}.

Azul (*bleu*)	similarité cosinus	**Mujer** (*femme*)	similarité cosinus	**Amor** (*amour*)	similarité cosinus
verde *vert*	0,934	muchacha *jeune fille*	0,930	anhelo *désir*	0,818
violeta *violet*	0,930	niña *fille*	0,915	cariño *cœur*	0,805
amarillo *jaune*	0,923	muchachita *fillete*	0,912	goce *plaisir*	0,801
púrpura *pourpre*	0,923	chica *jeune fille*	0,909	pasión *passion*	0,794
carmesí *cramoisi*	0,918	anciana *vielle dame*	0,900	afecto *affection*	0,789

TABLE 3.8 – *Exemples des résultats retournés par le modèle Word2vec, entraîné sur le corpus **MegaLite-Es**.*

7. Généralement, l'angle entre deux vecteurs est utilisé comme mesure de proximité géométrique entre les vecteurs, et le cosinus de l'angle est utilisé comme mesure de la similarité numérique (puisque le cosinus a la propriété intéressante d'être égal à 1 pour des vecteurs identiques et à 0 pour des vecteurs orthogonaux (Singhal et al., 2001)).

La table 3.9 montre les plongements calculés à partir du même groupe de requêtes de la table 3.8 mais en utilisant le corpus **MegaLite-Fr**. Si l'on compare ces dix plongements, les plus proches à chaque requête, on observe qu'ils génèrent des valeur de similarité légèrement différentes. Cela peut s'expliquer car l'apprentissage est influencé par les dimensions et la différence du contenu entre les corpus.

Bleu	similarité cosinus	**Femme**	similarité cosinus	**Amour**	similarité cosinus
foncé	0,915	fille	0,944	âme	0,829
vert	0,912	maîtresse	0,857	orgueil	0,822
blanc	0,886	dame	0,856	espoir	0,813
gris	0,884	fillette	0,837	hymen	0,812
pailleté	0,884	demoiselle	0,830	bonheur	0,802
piqueté	0,867	créature	0,826	idéal	0,790
noir	0,865	tricoteuse	0,824	amant	0,786
outremer	0,861	mademoiselle	0,822	égoïsme	0,784
nacré	0,860	duègne	0,813	amitié	0,782
transparent	0,849	mère	0,796	cœur	0,780

TABLE 3.9 – *Exemples des résultats retournés par le modèle Word2vec, entraîné sur le corpus **MegaLite-Fr**.*

3.2.2 Version étiquettes grammaticales (POS) et version lemmatisée

Ces deux versions ont été développées afin de mettre à disposition de la communauté des outils adéquats pour effectuer des analyses morphosyntaxiques plus complètes.

Étiquettes grammaticales

Pour cette version, une analyse syntaxique avec l'outil Freeling [8] (Padró et Stanilovsky, 2012) a été effectuée sur les documents de chaque corpus. Par la suite, les mots ont été remplacés par leur étiquette grammaticale *Part-of-Speech, POS* correspondante. Les corpus **MegaLite-Es/Fr/Pt** ont ainsi été traités séparément.

Par exemple, pour la phrase :

s=« *En la muerte dejarán sufrimientos y penas.* » [9],

8. Freeling peut être téléchargé à l'adresse : `https://nlp.lsi.upc.edu/freeling`
9. Traduction approximative : Dans la mort, ils laisseront des souffrances et des chagrins.

Freeling effectue l'analyse suivante (`Token` - **POS tag** = label) :

— `En` - **SP** = préposition ;
— `la` - **DA0FS0** = article féminin singulier ;
— `muerte` - **NCFS000** = nom féminin singulier ;
— `dejarán` - **VMIF3P0** = verbe principal conjugué au futur ;
— `sufrimientos` - **NCMP000** = nom commun masculin pluriel ;
— `y` - **CC** = connecteur ;
— `penas` - **NCFP000** = nom commun féminin au pluriel ;
— `.` - **Fp** = ponctuation.

Version lemmatisée

Pour cette version, les documents ont été analysés également avec le système Freeling pour détecter les trois principaux types de mots lexicaux : verbes, noms et adjectifs. Ensuite, ces éléments ont été remplacés par leur forme canonique correspondante (*lemme*) : les verbes sont ramenés à l'infinitif ; les noms au singulier et au genre masculin ; les adjectifs sont singularisés ; les articles, adverbes, prépositions, les chiffres et d'autres symboles non lexicaux ont été supprimés.

Par exemple, pour la phrase précédemment analysée :

s= « *En la muerte dejarán sufrimientos y penas.* »

la version lemmatisée est :

s=« `MUERTE DEJAR SUFRIMIENTO PENA.` »[10]

La table 3.10 montre quelques exemples de phrases récupérées des corpus **MegaLite-Es/Fr**, ainsi que leurs versions POS et lemmes.

10. Traduction approximative : MORT LAISSER SOUFFRANCE CHAGRIN.

Exemples récupérés de la pièce de théâtre *Manfredo* de Lord Byron (MegaLite-Es)		
Phrase	**POS**	**Lemmes**
Mi lámpara va a apagarse ; por más que quiera reanimar su luz moribunda.	DP1CSS VMIP3S0 VMIP3S0 SP VMN0000 PP3CN00 Fx SP CC PR0CN00 VMSP3S0 VMN0000 DP3CSN NCFS000 AQ0FS00	LAMPARA IR APAGAR QUERER REANIMAR LUZ MORIBUNDO
Si parece que duermo, no es el sueno el que embarga mis sentidos	CS VMIP3S0 CS VMIP1S0 Fc RN VSIP3S0 DA0MS0 VMIP1S0 DA0MS0 PR0CN00 VMIP3S0 DP1CPS NCMP000	PARECER DORMIR SONAR EMBARGAR SENTIDO
Exemples récupérés de la pièce de théâtre *Le chasseur d'ours* de Charles Buet (MegaLite-Fr)		
Le visage respire la bonté, la franchise, la simplicité, j'oserai même dire la candeur.	DA0MS0 NCMS000 VMIP3S0 DA0FS0 NCFS000 Fc DA0FS0 NCFS000 Fc DA0FS0 NCFS000 Fc VMIP3S0 Frc VMIF1S0 RG VM N0000 DA0FS0 NCFS000 Fp	VISAGE RESPIRER BONTÉ, FRANCHISE, SIMPLICITÉ, J' OSER DIRE CANDEUR.
Si mes souvenirs ne me trahissent point, la salle à manger et la bibliothèque n'étaient point indignes du salon.	CS DP1CPS NCMP000 RN PP1CS00 VMIP3P0 RN Fc DA0FS0 NCFS000 SP VMN0000 CC DA0FS0 NCFS000 AQ0MS0 Frc VMIP 3P0 RN AQ0CP00 SP DA0MS0 NCMS000 Fp	SOUVENIR TRAHIR, SALLE MANGER BIBLIOTHÈQUE N' ÉTAYER INDIGNE SALON.

TABLE *3.10* – *Phrases issues des corpus **MegaLite-Es/Fr** et leurs versions POS et lemmes.*

3.2.3 Modèle de langue *n*-grammes

Afin d'avoir plus d'information statistique utile pour une analyse plus complète, nous avons calculé également des *n*-grammes (n=1, 2 et SU4-bigrammes) pour construire un modèle de langue (ML). Les unigrammes et les bigrammes ont été calculés de manière classique, en utilisant une fenêtre de 1 et 2 mots respectivement.

Pour les SU4-bigrammes (Lin, 2004), nous avons considéré des paires de mots dans une fenêtre de taille variable. À ce fin, à partir d'un mot *i* nous consi-

dérons les mots dans les positions $i+1, i+2, i+3, i+4$, en composant une paire différente pour chaque mot i. Par exemple, pour la phrase **s**=« *Les enfants jouent dans le jardin* », nous pouvons calculer les 14 SU4-bigrammes : [*Les enfants, Les jouent, Les dans, Les le, enfants jouent, enfants dans, enfants le, enfants jardin, jouent dans, jouent le, jouent jardin, dans le, dans jardin, le jardin*]. La table 3.11 affiche les cinq premiers n-grammes de mots avec un nombre important d'occurrences venant des corpus espagnol et français.

Unigrammes	fréquence	**Bigrammes**	fréquence	**SU4-bigrammes**	fréquence
***La señorita de Travelez* de Carlos Arniches** extrait de **MegaLite-Es**					
don	613	don gonzalo	362	picavea don	43
gonzalo	402	don marcelino	207	marcelino gonzalo	33
numeriano	238	don arístides	27	don arístides	27
marcelino	233	gonzalo don	26	don numeriano	26
picavea	196	por dios	14	marcelino numeriano	25
***Juvenilia* de Miguel Cane** extrait de **MegaLite-Es**					
colegio	65	miguel cané	47	san ignacio	6
miguel	48	colegio nacional	8	tres meses	5
cané	47	buenos aires	7	cané colegio	5
juvenilia	46	doctor agüero	7	cinco años	4
jacques	44	san ignacio	6	entrar colerio	4
***Le curé du village* de Honoré de Balzac** extrait de **MegaLite-Fr**					
curé	403	curé village	224	jeune homme	27
graslin	336	madame graslin	217	procureur général	27
madame	317	monsieur bonnet	98	monsieur grossetête	26
véronique	286	monsieur curé	33	l'abbé gabriel	24
monsieur	283	jean françois	30	jeune abbé	22
***La sève immortelle* de Laure Conan** extrait de **MegaLite-Fr**					
jean	141	sève immortelle	86	madame tilly	17
tilly	106	jean tilly	37	pauvre enfant	12
d'autrée	90	madame d'autrée	35	jeune fille	11
immortelle	89	monsieur tilly	26	nouvelle france	11
sève	87	colonel d'autrée	19	monsieur laycraft	11

TABLE *3.11* – *Fréquence de n-grammes des corpus **MegaLite-Es/Fr**.*

3.3 Corpus littéraire d'émotions LiSSS

Dans cette section nous introduisons le corpus **LiSSS**, un corpus de petite taille composé de phrases littéraires en espagnol, soigneusement sélectionnées et annoté manuellement en émotions. Le corpus **LiSSS** possède le double avantage d'être homogène en termes de genre, ne contenant que des phrases considérées comme littéraires, et hétérogène en termes de classes d'émotions. Dans d'autres corpus littéraires, comme **MegaLite**, de grande taille, les textes sont

surchargés de phrases de transition qui donnent une fluidité à la lecture et fournissent les relations nécessaires entre les idées exprimées dans les phrases littéraires. Les phrases de transition peuvent être par exemple : «Dans ces conditions», «de sorte que» ou «c'est pourquoi». **LiSSS** contourne aussi quelques inconvénients comme la présence du bruit (des symboles, des caractères spéciaux, des phrases coupées, des mots collés, une syntaxe incorrecte, etc.). Ceci peut-être un inconvénient si l'on cherche à analyser de façon plus fine et formelle des textes littéraires.

Puisque, le corpus **LiSSS** possède un nombre réduit de phrases, il n'est pas adapté à l'entraînement d'algorithmes basés sur l'apprentissage automatique. Or, l'objectif de ce corpus n'est pas d'être employé dans des processus d'apprentissage mais, il a été conçu pour tester la qualité et la performance des algorithmes d'analyse littéraire ou des émotions.

3.3.1 Annotation d'émotions

Le corpus **LiSSS** a été constitué manuellement à partir de textes littéraires en espagnol venant d'environ 200 auteurs. Nous incluons des auteurs hispanophones et non hispanophones (en conservant uniquement les traductions officielles) afin d'enrichir le contenu émotionnel, le lexique et le sens expressif du corpus. Les phrases de transition (en langue générale) ainsi que celles trop courtes ($\leq$ 3 mots) ou trop longues ($\geq$ 50 mots) ne sont pas incluses. Cette exclusion a pour but de concevoir un corpus avec un lexique complexe, esthétique ou ayant des figures littéraires comme l'anaphore ou la métaphore, en plus de la charge émotionnelle.

Chaque élément de ce corpus est un objet linguistique complexe composé à son tour d'une ou plusieurs phrases, expressions ou paragraphes. Les phrases ont été extraites de citations, récits, romans, nouvelles et poèmes. Les objets linguistiques ont été annotés en cinq catégories d'émotions :

— Haine (**A**),
— Amour (**L**),
— Peur (**F**),
— Bonheur (**H**),
— Tristesse (**S**)

Pour l'annotation, 12 personnes hispanophones natives ayant une formation de niveau universitaire ont participé. Puisque les phrases peuvent appartenir à plusieurs émotions, les annotateurs pouvaient choisir des étiquettes appartenant à plusieurs émotions. Nous avons ainsi créé des fichiers (en formats texte et XML) annotés avec des émotions. Dans la version texte, chaque fichier contient une ligne par phrase, avec des informations structurées en trois champs séparés par une tabulation :

```
ID      Phrase      # Auteur
```

Le champ ID est composé d'un numéro entier (1,2,3,...) suivi d'un code (A, L, F, H, S) correspondant aux émotions perçues. Dans la version XML, la même structure est conservée en utilisant des balises XML appropriées. Une phrase multi-émotionnelles aura autant de codes que de catégories auxquelles elle appartient. Les phrases ont été sélectionnées afin de garder un certain équilibre entre les catégories mais cela n'est pas toujours garanti.

Comme indiqué, quelques éléments du corpus **LiSSS** sont composés de plusieurs phrases courtes, donnant lieu à des mini-paragraphes. Cela permet de respecter au maximum la cohérence et les émotions correspondantes. Par exemple, la phrase de J.P. Sartre dans la catégorie **Peur (F)** :

« *Todos los hombres tienen miedo. El que no tiene miedo no es normal. No tiene nada que ver con el coraje.* »

est un paragraphe de trois phrases :

1. *Todos los hombres tienen miedo* (Tous les hommes ont peur).
2. *El que no tiene miedo no es normal* (Celui qui n'a pas peur n'est pas normal).
3. *No tiene nada que ver con el coraje* (Cela n'a rien à voir avec le courage).

Les caractéristiques d'une première version, avec 500 phrases du corpus **LiSSS**, sont montrées dans la première ligne de la table 3.12. Toutes les expériences d'annotation expliquées ci-après ont été effectuées sur cette version. Sur la deuxième ligne sont montrées les caractéristiques d'une deuxième version contenant 2 000 phrases avec une annotation manuelle effectuée par une seule personne. Les deux versions du corpus **LiSSS** peuvent être téléchargées dans le site : `http://juanmanuel.torres.free.fr/corpus/lisss`.

Nombre de phrases	**Mots par phrase**	**Total de mots**	**Auteurs hispano-phones**	**Auteurs non hispano-phones**	**Annotateurs**
500 (version 1)	≈ 19	9 400	37	164	12
2 000 (version 2)	≈ 21	42 217	N/A	N/A	1

TABLE *3.12* – *Caractéristiques du corpus **LiSSS**.*

Nous avons constaté que certaines phrases étaient étiquetées avec des émotions opposées. Ce phénomène, lié à l'ambiguïté, est couramment observé dans les textes du genre littéraire. Il devient donc une tâche difficile pour la classification automatique et même manuelle. Par exemple, Léon Tolstoï est connu pour son style fortement émotionnel. Il a souvent écrit sur le chevauchement

et la frontière floue entre l'amour et la mort. Pour mieux interpréter cette ambiguïté, les classifications effectuées par les annotateurs ont été intégrées dans le corpus **LiSSS** par une stratégie de vote démocratique afin de produire une classification pondérée.

3.3.2 Stratégie de vote démocratique

Tout d'abord nous avons fixé un seuil $t = 0,5$ (équivalant à 50% des émotions). Pour chaque phrase du corpus **LiSSS**, nous gardons les catégories d'émotions sélectionnées par, au moins, une fraction $p(e)$ de n annotateurs, tel que $p(e) \geq t$.

- Nous calculons la probabilité $p(e) = Count(e)/12; e \in \{A, L, F, H, S\}$ sur tous les annotateurs, où $Count(e)$ correspond au nombre d'annotateurs qui ont attribué l'émotion e à la phrase analysée.
- **SI**, il y a une ou plusieurs émotions e dont $p(e) \geq t$, le processus est terminé et la sortie est la concaténation des émotions ayant $p(e) > 0,5$.
- **SINON**, le seuil est réduit à $t = 0,3$ et la sortie est recalculée.
- Finalement, s'il n'y a pas d'émotions au-dessus de ce seuil, nous diminuons $t = 0,2$ et nous répétons le processus jusqu'à sélectionner les émotions les plus pertinentes.

$t = 0,2$ semble être un seuil approprié dans la condition hypothétique où une personne a annoté une phrase avec toutes les émotions possibles. Dans ce cas, chaque émotion e a au moins une probabilité $p(e) = 0,2$. En utilisant un vote démocratique, la sortie peut être multi-étiquetée.

Il y a plusieurs phrases dont les émotions se chevauchent. Elles ont été marquées par l'algorithme de vote, en traitant toutes les classifications des annotateurs. Dans la table 3.13 les colonnes **A/x, F/x, H/x, L/x, S/x** représentent la proportion de phrases contenant qu'une seule émotion (mono-classe), ainsi que la proportion des phrases contenant des émotions superposées (multi-classes). Pour calculer cette proportion, nous avons divisé le nombre de phrases classifiées pour chaque émotion par le nombre d'annotateurs. Par exemple, les votants ont marqué 89,5 phrases uniquement comme **L** et 48,5 phrases combinant **L** avec d'autres émotions **x**. Le degré de chevauchement est obtenu en considérant, $a =$ la moyenne des phrases mono-classe, et $b =$ la moyenne des phrases multi-classes, nous calculons donc $b/(a + b)$. Nous répétons ce processus pour chaque émotion, et au final on calcule la moyenne de toutes les valeurs obtenues. Un exemple d'une phrase multi-émotions pourrait être la suivante, classée aux émotions Haine **A** et Amour **L** :

«Del amor al odio, solo hay mas amor.» # *Mario Benedetti*

(De l'amour à la haine, il n'y a que plus d'amour.)

A/x	L/x	F/x	H/x	S/x	Chevauchement
74,1/31.3	89,5/48,5	103,2/31,4	92,4/32,4	115,3/64,8	33,5%

TABLE *3.13* – *Quantité de phrases uni-émotion vs multi-émotions du corpus **LiSSS**.*

La table 3.14 montre de manière détaillée la distribution des classifications des phrases par paires d'émotions, ainsi que leurs degrés de chevauchement. Nous observons un degré de chevauchement élevé dans les paires d'émotions « Bonheur–Tristesse » et « Amour–Tristesse », avec une valeur de **HS**=18,7 et **LS**=19,9 des phrases multi-émotions, respectivement. L'ambiguïté émotionnelle représente un défi supplémentaire pour les algorithmes de classification automatique. Nous précisons que ces valeurs correspondent à la moyenne des phrases chevauchées entre deux émotions, (calculée à partir du retour des annotateurs). C'est la raison pour laquelle, nous avons des valeur supérieures à 100 comme pour l'émotion Peur (**F**).

Emotion	A	L	F	H	S	Chevauchement %
A	74,1	12,1	7,9	2,3	9,0	29,7
L		89,5	5,7	10,8	19,9	27,7
F			103,2	0,6	17,2	35,5
H				92,4	18,7	38,9
S					115,3	35,9

TABLE *3.14* – *Distribution moyenne des phrases multi-émotions du corpus **LiSSS**.*

3.4 Application des corpus littéraires : une analyse stylistique statistique

Nous avons vu que l'analyse des textes littéraires est très intéressante car on peut y observer des structures linguistiques assez complexes, qui sont peu fréquentes dans les textes non littéraires (Rosso et al., 2009). Or, une question nous est venu à l'esprit, pourrait-on mettre en évidence ces structures au moyen d'outils statistiques en franchissant les barrières de la langue et du lexique ? Pour tenter d'y répondre, nous avons réalisé une étude comparative statistique de textes littéraires d'auteurs francophones et hispanophones. L'objectif étant de déterminer s'il est possible de détecter un style littéraire de documents écrits dans deux langues, sans analyser le lexique mais dans un niveau d'abstraction purement morphosyntaxique.

Notre étude comparative a été réalisée en utilisant les versions POS (Section 3.2.2) des corpus **MegaLite-Es/Fr** avec une mesure de divergence de distribu-

	Documents	Phrases	POS
Références linguistiques, P dans l'équation (3.3)			
Ref-Es	391	1,3 M	23,0 M
Ref-Fr	314	1,3 M	32,0 M
Auteurs hispanophones, Q_{es} dans l'équation (3.3)			
Gabriel García Márquez	31	40,0 K	1,0 M
Jorge Luis Borges	37	10,0 K	200,0 K
Juan José Benítez López	11	123,0 K	1,7 M
Juan-Manuel Torres-Moreno	51	6,4 K	106,0 K
Julio Cortázar	42	35,8 K	942,0 K
Miguel de Cervantes Saavedra	24	44,0 K	1,1 M
Rubén Darío	15	9,6 K	91,6 K
Auteurs francophones, Q_{fr} dans l'équation (3.3)			
Charles de Montesquieu	17	11,0 K	323,0 K
Denis Diderot	12	19,0 K	492,0 K
Gustave Aimard	19	70,0 K	1,8 M
Paul Henri Corentin Féval	22	126,0 K	2,6 M
Victor Hugo	24	148,0 K	3,0 M

TABLE *3.15* – *Caractéristiques de documents utilisés pour l'étude de la divergence de Jensen-Shannon.*

tion de probabilités adéquate. En effet, les étiquettes grammaticales permettent une comparaison de documents dans un niveau d'abstraction autre que le niveau lexical. On évite ainsi les différences évidentes (dichotomiques, culturelles ou autres) du lexique entre les deux langues. Les étiquettes POS ont été traitées comme des évènements ayant des probabilités d'occurrences et les documents ont été représentés par leur distribution de probabilités.

À partir des version POS des corpus **MegaLite-Es** et **MegaLite-Fr**, nous avons tiré au hasard deux sous-ensembles statistiquement représentatifs de documents. Ces sous-ensembles seront appelés corpus *Ref-Es* et corpus *Ref-Fr*, et ils seront considérés comme de références linguistiques dans cette expérience. À partir de ces deux corpus nous avons calculé leurs distributions de probabilités respectives, à savoir P_{es} et P_{fr}. D'un autre côté, nous avons établi une liste d'auteurs hispanophones et une autre d'auteurs francophones à étudier. Un certain nombre d'œuvres de chaque auteur a été extrait également du corpus **MegaLite-Es/Fr** respectivement. Le sous-ensemble d'œuvres en espagnol permet de calculer une distribution de probabilités Q_{es} (respectivement Q_{fr} en français) Les caractéristiques (documents, phrases, POS) des œuvres analysées et des références linguistiques sont affichées dans la table 3.15.

La question concernant le style littéraire peut se poser de la façon suivante : étant donné l'œuvre d'un auteur (soit hispanophone, soit francophone), comment se situe-t-elle vis-à-vis des corpus de référence espagnol et français ? Dans d'autres termes, nous voulons mesurer la proximité du style des auteurs par rapport au style global de la littérature contenue dans **MegaLite** (dans les deux langues). De façon intuitive la réponse est que les auteurs hispanophones (respectivement francophones) devraient être plus proches du style exprimé dans **MegaLite-Es** (respectivement **MegaLite-Fr**). La question intéressante est de savoir si l'on peut justifier ces réponses intuitives *sans utiliser le lexique*, qui est d'ailleurs très différent parmi les auteurs étudiés. Nous sommes maintenant en mesure de répondre à ces questions à l'aide d'une divergence de distribution des probabilités entre P et Q. En particulier, étant donné leur propriétés symétriques (de P vers Q et vice-versa), nous avons opté pour utiliser la divergence de Jensen-Shannon.

3.4.1 Divergence de distribution de probabilités appliquée aux styles

La divergence de Kullback-Leibler $\mathcal{D}_{\mathcal{KL}}$ (Kullback et Leibler, 1951) calcule la divergence entre deux distributions de probabilité. Pour deux distributions de probabilité discrètes P et Q, la divergence $\mathcal{D}_{\mathcal{KL}}$ de Q en relation avec P est :

$$\mathcal{D}_{\mathcal{KL}}(P||Q) = \frac{1}{2}\sum_{w\in P} P_w \log_2 \frac{P_w}{Q_w}, \tag{3.1}$$

où P_w est la distribution de probabilité des mots w dans le document D_P et Q_w la distribution de probabilité des mots w dans le document D_Q, dans lequel $D_Q \subset D_P$. Les distributions de probabilités P_w et Q_w sont obtenues en calculant :

$$P_w = C_w^P/|P|; \qquad Q_w = \begin{cases} C_w^Q/|Q|\,, & \text{si } w \in Q \\ \delta & \text{ailleurs} \end{cases}, \tag{3.2}$$

dans lequel $C_w^{P|Q}$ est le nombre d'occurrences du mot w dans la distribution P ou Q ; $|P|$ la taille du document D_P en terme de mots, $|Q|$ la taille de document D_Q en nombre de mots, et $\delta > 0$ un facteur de lissage. La divergence de Jensen-Shannon ($\mathcal{D}_{\mathcal{JS}}$) est la version symétrisée de la divergence $\mathcal{D}_{\mathcal{KL}}$:

$$\begin{aligned} \mathcal{D}_{\mathcal{JS}}(P||Q) &= \frac{1}{2}\mathcal{D}_{\mathcal{KL}}(P||M) + \frac{1}{2}\mathcal{D}_{\mathcal{KL}}(Q||M); \quad M = \frac{1}{2}(P+Q) \\ &= \frac{1}{2}\sum_w P_w \log_2 \frac{2P_w}{P_w+Q_w} + Q_w \log_2 \frac{2Q_w}{P_w+Q_w}. \end{aligned} \tag{3.3}$$

Les valeurs de divergence normalisées peuvent être utilisées pour mesurer la proximité informative entre des paires de documents.

Les représentations des textes employées dans nos calculs sont les n-grammes d'étiquettes POS ($n = 1, 2$ et SU4). En effet, les n-grammes de mots détectent des séquences prototypiques des phrases. Ces séquences ont un rapport au style et aux tournures de la langue des auteurs. Ceci est statistiquement vrai si les séquences sont calculées sur un corpus de taille adéquate. Nous pensons que cet idée reste valable dans le cas des n-grammes d'étiquettes POS, où ces séquences représentent un niveau d'abstraction morphosyntaxique lié au style.

La figure 3.1 montre les résultats de nos expériences. Les traits courts représentent les œuvres de chaque auteur (distribution Q). Les auteurs sont placés en fonction de leur divergence $\mathcal{D}_{\mathcal{JS}}$ calculée à partir des corpus de références linguistiques (distribution P) : *Ref-Es* (lignes bleues) et *Ref-Fr* (lignes rouges). Les cinq premiers auteurs de la figure sont francophones. On observe que leurs divergences par rapport à *Ref-Es* sont les plus élevées, à l'opposé des auteurs hispanophones dont leurs divergences sont plus faibles. Un comportement opposé est observé avec les divergences obtenues par rapport à l'ensemble *Ref-Fr*. Comme prévu, les auteurs hispanophones ont obtenu des divergences plus élevées que les auteurs francophones.

La figure 3.2 montre la moyenne et l'écart-type des divergences. Dans cette figure, on peut apprécier un chevauchement important entre les écarts types des divergences correspondantes de cinq auteurs hispanophones : Julio Cortázar, José Luis Borges, Rubén Darío et Juan-Manuel Torres-Moreno. Or, les biographies de ces auteurs montrent qu'ils ont séjourné plusieurs années en France ou en Belgique. Cela nous laisse supposer que leur style d'écriture a été influencé, d'une certaine manière, par leur contact avec un environnement francophone (culture, langue, etc.). Ce point pourrait donc expliquer le chevauchement des écarts-types obtenus.

3.5 Conclusion

Dans ce chapitre, nous présentons les différents corpus littéraires construits. Le corpus **MegaLite** est composé de documents littéraires en trois langues, dont la taille convient aux tâches d'apprentissage automatique. Le corpus **LiSSS** de taille réduite, mais bien contrôle, idéal pour les tâches d'évaluation. La production ainsi que l'étude des corpus sont des tâches extrêmement importantes dans différents domaines du TAL, comme on a vu au long de ce chapitre. Concernant les corpus littéraires, nous remarquons que les caractéristiques des textes littéraires demandent un niveau d'analyse fine pour leur compréhension et traitement. Les différents corpus construits et présentés dans ce chapitre constituent

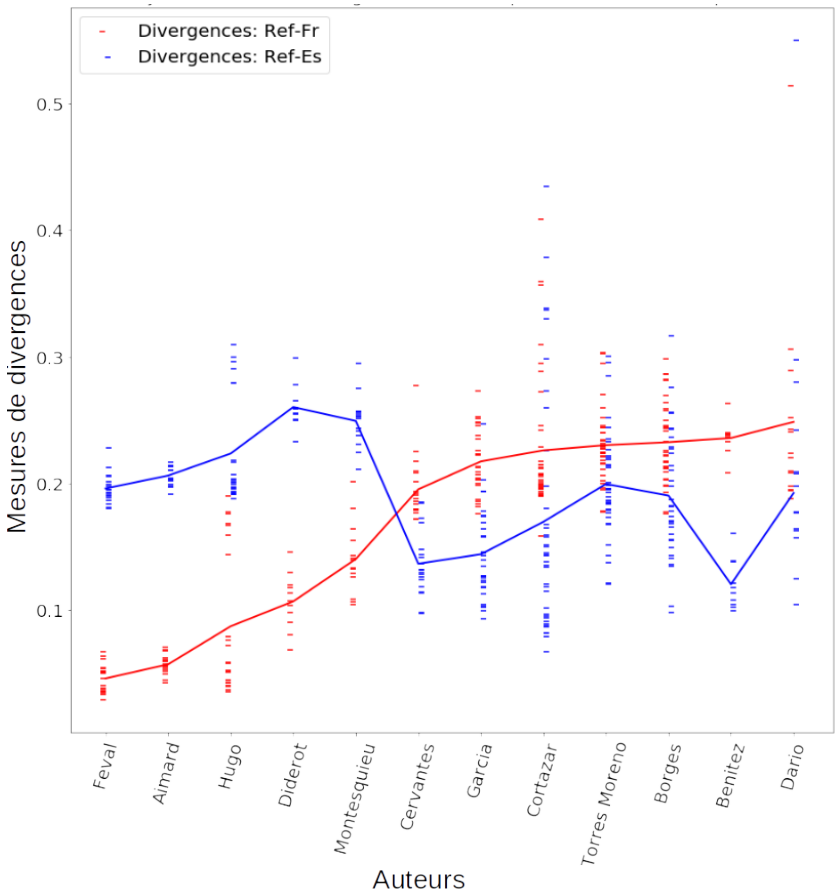


FIGURE 3.1 – *Diagramme de divergences Jensen-Shannon des corpus **MegaLite-Es/Fr**.*

une contribution à l'étude et l'analyse de textes littéraires, ainsi qu'au développement d'algorithmes pour la Créativité computationnelle dans le domaine du calculable, comme on le verra dans les chapitres suivants. Une dernière contribution au corpus **MegaLite** a été la constitution de la version en portugais. Même si nous n'avons pas réalisé des expériences de génération littéraire, ce corpus représente une contribution importante dans le domaine au niveau de ressources spécialisées (Morgado et al., 2022).

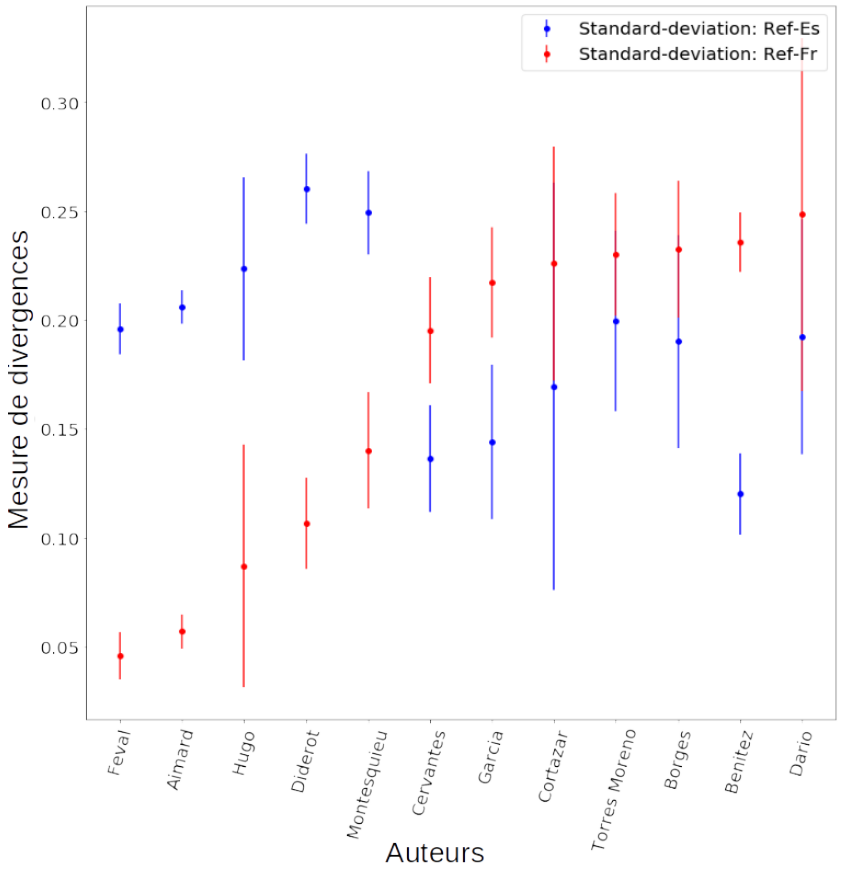


FIGURE 3.2 – *Diagramme d'écart-type calculé des divergences Jensen-Shannon.*

Chapitre 4

Modèles pour la génération de texte littéraire

Sommaire

4.1 Génération de texte : une approche stochastique (Stoch) 57
4.1.1 Première étape : étude stochastique de la langue 57
4.1.2 Deuxième étape : étude sémantique de la langue 59
4.2 Génération de texte au moyen de *canned text* 61
4.2.1 Génération des Structures Grammaticales partiellement-Vides 61
4.2.2 Modèle d'analyse pertinent avec Word2vec (CaP) 62
4.2.3 Modèle de composition de vecteurs (CaV) 64
4.2.4 Modèle basé sur des traits psychologiques (CaT) 67
4.3 Génération de rimes sémantiques (CaR) 69
4.3.1 Rime sémantique 70
4.3.2 Production de rimes avec similarité sémantique 70
4.3.3 Première étape : *canned text* 72
4.3.4 Deuxième étape : sélection du vocabulaire 72
4.4 Conclusion 76

La génération automatique de texte (GAT) est une tâche largement abordée par les chercheurs dans le domaine du TAL (Sridhara et al., 2010b; Szymanski et Ciota, 2002; Pérez y Pérez, 2015; Torres-Moreno, 2014). Actuellement, le champ d'application de la GAT s'élargit et de nombreux travaux récents visent à générer des textes liés à différents domaines spécifiques, comme le domaine littéraire (Oliveira et Cardoso, 2015; Molins et Lapalme, 2015). C'est dans ce domaine que nos algorithmes vont se focaliser dans ce chapitre.

La génération de texte littéraire est particulièrement intéressante, par rapport à d'autres types de GAT qui posent moins des difficultés aux systèmes TAL. Cependant, les textes littéraires ont été, dans la plupart des travaux, délaissés. Ceci est normal, car d'abord, le discours littéraire est plus complexe que d'autres genres. Les documents littéraires font souvent référence à des mondes ou à des situations imaginaires ou allégoriques, au contraire des autres genres traitant plutôt de situations factuelles. Ces caractéristiques et d'autres comme l'occurrence de mots peu utilisés dans le langage commun et les figures littéraires comme la métaphore, rendent extrêmement complexe la tâche d'analyse automatique de ce genre de documents.

En outre, les textes littéraires ne sont toujours pas perçus de la même manière chez les personnes. Cette perception peut varier en fonction de la sensibilité ou l'état émotionnel du lecteur. On peut donc voir que la perception littéraire est subjective et, de ce point de vue, il est difficile de garantir que le texte généré par un algorithme sera perçu comme littéraire.

Pour diminuer la subjectivité, nous avons basé notre étude sur la génération de phrases littéraires sur certaines caractéristiques linguistiques telles qu'un vocabulaire étendu, éloigné de celui de la langue générale et chargé d'émotions, ainsi que des constructions syntaxiques complexes comme la rime. Selon Sharples (1996), l'écriture est un processus créatif où l'écrivain peut exprimer une idée de différentes manières. C'est à l'écrivain de jouer avec les ressources linguistiques nécessaires pour donner à cette idée la structure souhaitée. Sur la base de l'analyse de Sharples et les caractéristiques linguistiques que nous avons remarquées, nous avons introduit notre propre définition de phrase littéraire (voir Section 1.2.2).

On peut penser que cette définition pourrait guider partiellement nos analyses pour concentrer nos efforts dans l'étude de ces caractéristiques. Or, on constate que cette définition est, comme toutes les définitions de texte littéraires, pas assez utile et très vague du point de vue informatique. Elle ne permet pas de transposer avec certitude l'idée de base dans un algorithme de création de phrases artificielles littéraires. Dans son livre, *La notion de Littérature*, Todorov et al. (1973) conçoit la littérature comme un phénomène encore en évolution. Il s'interroge donc à savoir si nous sommes véritablement en mesure de répondre à la question, «qu'est la littérature ?». Nous avons donc opté d'introduire, dans le cadre de cette thèse, une définition opérationnelle et restreinte de phrase artificielle littéraire.

Définition 2 : *Une phrase artificielle littéraire est un objet linguistique qui respecte une structure grammaticale construite à partir d'un corpus accepté comme littéraire. Cet objet doit se composer d'un vocabulaire sélectionné selon un en-*

semble de contraintes lexicales et sémantiques adéquates.

Cette définition nous permettra d'implémenter plus aisément nos algorithmes dans le domaine du calculable pour la génération de texte littéraire. Dans ce chapitre, nous expliquons les différentes méthodes que nous avons employées pour aborder la problématique menée pour la génération de phrases littéraires. Ensuite, dans le but de produire des paragraphes, nous proposons une méthode pour la génération des paires des phrases rimées et sémantiquement liées.

4.1 Génération de texte : une approche stochastique (Stoch)

Dans cette section, nous expliquons notre modèle **Stoch**, basé sur une méthode stochastique. Ce modèle est composé de deux étapes décrites comme suit.

1. La première étape est basée sur l'algorithme de Viterbi lors qu'il est utilisé pour l'analyse stochastique et grammaticale du texte (Manning et Schütze, 1999). L'algorithme de Viterbi est un algorithme très efficace pour déterminer le chemin le plus probable pour parcourir une séquence d'états dans un graphe de Markov (Viterbi, 2006). Dans notre cas, un état de la séquence déterminée par l'algorithme de Viterbi correspond à une séquence de mots qui composent une phrase en cours de génération par notre méthode.
2. La deuxième étape, utile pour l'analyse sémantique, implémente un modèle Word2vec pour la génération des plongements (représentations vectorielles). Ces plongements sont générés à partir du vocabulaire d'un corpus d'entraînement (voir Section 2.1). Par la suite, on utilise Word2vec pour trouver le plongement d'un mot Q, après on calcule l'ensemble des plongements les plus proches de Q en fonction de mesures de distances appropriées (équations 4.4 et 4.5), et on choisi un mot parmi cet ensemble des plongements.

4.1.1 Première étape : étude stochastique de la langue

L'objectif de la mise en œuvre d'un modèle stochastique est d'analyser un ensemble de séquences d'étiquettes POS pour trouver la distribution de probabilité d'occurrence des séquences. Par la suite, nous construirons des nouvelles séquences d'étiquettes POS à partir de l'étude probabiliste effectuée sur les séquences d'entraînement. Pour générer les séquences d'étiquettes POS d'entraînement ou *séquence d'états observés*, nous avons construit le corpus **8KF** avec 7 679 phrases sélectionnées du corpus **MegaLite-Es** (voir table 4.1).

	Phrases	Tokens	Caractères
8KF	7 679	114 K	652 K
Moyenne par phrase	–	15	85

TABLE 4.1 – *Caractéristiques du corpus **8KF**.*

La sélection des phrases a été effectuée manuellement pour mieux contrôler nos expériences. Nous avons retenu des phrases contenant une ou deux idées au maximum avec une longueur moyenne de 10 mots. Un prétraitement classique a été effectué sur ce corpus, à savoir : normalisation, tokenisation et l'élimination des *tokens* indésirables (chiffres, acronymes, heures et dates). Le corpus filtré a été ensuite analysé à l'aide du système FreeLing pour remplacer chaque mot de la phrase par son étiquette POS correspondante. À la fin de cette analyse, un nouveau corpus **8KPOS** a été obtenu avec $s = 7\,679$ séquences d'étiquettes POS. Les s séquences de **8KPOS** ont été traitées avec l'algorithme de Viterbi, qui calcule la matrice de transition $P[s \times s]$. Cette matrice sera utilisée pour créer des nouvelles séquences POS n'existant pas dans le corpus **8KPOS**.

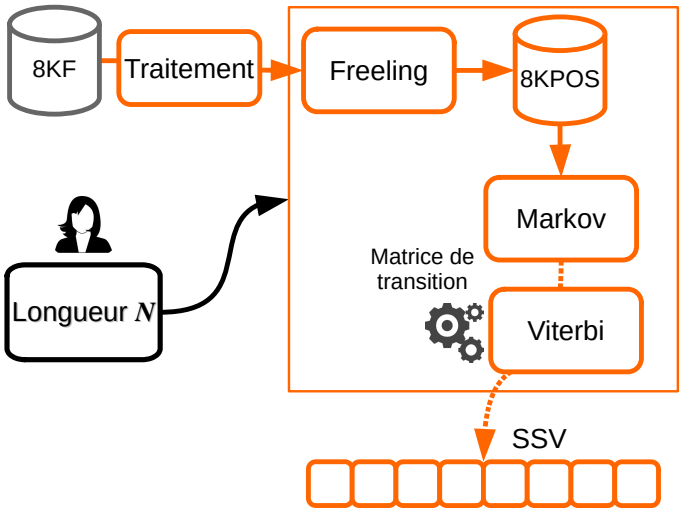


FIGURE 4.1 – *Schéma du modèle **Stoch** pour la génération de Structures Stochastiques Vides (SSV).*

Dans cet algorithme, X_g représente l'état d'une étape de la création d'une phrase, à la $g^{ième}$ itération, qui correspond à une séquence d'étiquettes POS. Suivant la procédure de Markov, dans l'itération g on sélectionne l'étiquette suivante POS_{g+1}, avec une probabilité d'occurrence maximale, étant donné l'étiquette actuelle POS_g de l'état X_g. L'étiquette POS_{g+1} sera ajoutée après l'état

X_g pour générer l'état X_{g+1}. $P(X_{g+1} = Y|X_g = Z)$ est la probabilité de transition d'un état X_g à un autre X_{g+1}, obtenue avec l'algorithme de Viterbi. Les transitions sont répétées, jusqu'à ce que la longueur N souhaitée soit atteinte. Nous appelons le résultat, X_N, *Structure Stochastique Vide* (SSV). Dans la figure 4.1 chaque casse vide représente une étiquette POS qui sera remplacée par un mot, lors de l'étape finale de génération de la nouvelle phrase (qui sera détaillée dans la section suivante).

4.1.2 Deuxième étape : étude sémantique de la langue

Dans cette étape, les étiquettes POS des SSV générées précédemment seront analysées et remplacées, afin de générer de nouvelles phrases $f(Q, N)$, où Q correspond à la requête (contexte) fournie par l'utilisateur et N est le nombre de mots. À cette fin, nous introduisons un algorithme d'interprétation sémantique à l'aide d'un modèle Word2vec.

Pour l'entraînement de Word2vec nous avons utilisé le corpus **MegaLite-Es** et l'outil Gensim [1] pour sa mise en œuvre. Gensim est une ressource libre écrite en Python, qui permet le développement des modèles d'apprentissage tels que **Word2vec, FastText, Latent Semantic Indexing** et autres.

Les paramètres d'entraînement décrits dans la Section 3.2.1 ont été re-configurés de la manière suivante :

— Nous considérons uniquement les mots ayant plus de 5 occurrences dans le corpus ;
— la fenêtre contextuelle possède une longueur de $c = 10$ mots ;
— le nombre de dimension pour les représentations vectorielles a été configuré à 60 ;
— l'architecture d'apprentissage a été *Skip-gram*.

Quand le processus d'entraînement a été finalisé, le modèle est capable de renvoyer un ensemble des plongements associés à une requête Q. La requête correspond à un mot donné par l'utilisateur. Autrement dit, Word2vec reçoit une requête Q et renvoie un lexique $L(Q) = (w_1, w_2, ..., w_m)$, qui représente un ensemble de $m = 10$ mots sémantiquement proches de Q. La valeur de m a été définie empiriquement car nous avons constaté que, plus sa valeur est incrémentée, plus les mots de $L(Q)$ perdent leur rapport sémantique vis-à-vis de Q.

1. Disponible en : `https://pypi.org/project/gensim/`

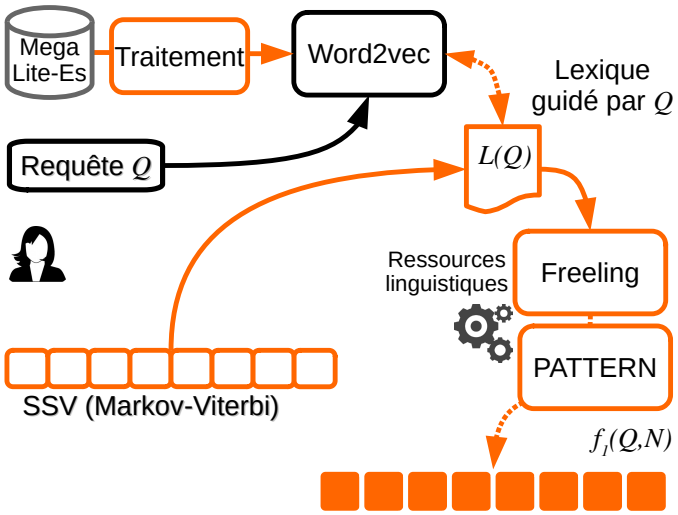


FIGURE 4.2 – *Schéma du modèle **Stoch** basé sur des méthodes de Markov et Word2vec.*

Nous analyserons maintenant la SSV. Tout d'abord, il faut identifier et classer les étiquettes POS. À cette fin, nous avons considéré deux catégories : les étiquettes complètes (POS_λ) qui concernent les mots lexicaux et les étiquettes fonctionnelles (POS_Φ) qui correspondent à la ponctuation et aux mots fonctionnels. Les mots fonctionnels (des articles, adverbes, prépositions, etc.) sont des mots qui jouent un rôle à l'intérieur de la langue elle-même : ils fournissent des informations sur les relations syntaxiques entre mots ou ensembles de mots. Les étiquettes POS_Φ seront remplacées par des mots qui peuvent être récupérés à partir de ressources linguistiques. En particulier, nous avons utilisé les dictionnaires de Freeling, où chaque étiquette POS_Φ est associée à une liste de mots ou de signes de ponctuation qui respectent la catégorie signalée par l'étiquette.

Les étiquettes POS_λ seront remplacées par les mots renvoyés par Word2vec à partir de $L(Q)$. Si aucun des mots de $L(Q)$ ne possède l'inflexion syntaxique requise par POS_λ, nous nous replions sur le système PATTERN [2], pour effectuer les conjugaisons ou les conversions en genre et/ou en nombre afin de remplacer correctement l'étiquette POS_λ en question. Si l'ensemble des mots $L(Q)$ ne contient aucun type de mot qui soit approprié ou qui puisse être manipulé avec PATTERN pour remplacer l'étiquette POS_λ, un mot $w_i \in L(Q)$, aussi proche de Q que possible (en fonction de la distance calculée par Word2vec) sera considéré. Une nouvelle requête $Q^* = w_i$ sera définie et utilisée pour générer un nouvel ensemble de mots $L(Q^*)$.

2. PATTERN est une bibliothèque en Python qui peut être utilisée pour mettre en œuvre des tâches de TAL. (`https://www.analyticsvidhya.com/blog/2021/11/pattern-library-for-natural-language-processing-in-python`)

Cette procédure sera itérée, jusqu'à ce que $L(Q^*)$ contienne un mot pouvant remplacer POS_λ. Une fois que toutes les étiquettes dans SSV ont été remplacées, le résultat sera une nouvelle phrase f qui n'existe pas ni dans les corpus **MegaLite-Es** ni dans **8KF**. La figure 4.2 montre le processus décrit ci-dessus.

4.2 Génération de texte au moyen de *canned text*

Notre modèle **Stoch** est capable de reproduire certaines structures linguistiques, grâce à l'analyse du corpus **MegaLite-Es**. Ce modèle fonctionne relativement bien pour la génération de phrases courtes, ne dépassant pas cinq mots. Cependant, lorsque l'on veut étendre la longueur des phrases à $N > 5$ mots, nous avons constaté une perte de cohérence et de lisibilité. Une analyse manuelle a montré que la contrainte venait de la SSV elle même, une structure que limitait la sélection d'un lexique adéquat. Ainsi, plus la taille de la structure s'élargissait, plus la grammaticalité se dégradait. Nous avons décidé alors de mettre en œuvre une méthode de texte en boîte pour la génération de texte.

La méthode de texte en boîte (aussi couramment appelée *canned text*) est basée sur la construction de structures grammaticales. *Canned text* est employée très fréquemment dans des travaux de recherche visant la génération de dialogues (McRoy et al., 2003; van Deemter et al., 2005) ou des poèmes (Oliveira, 2017). Cependant, nous pensons que cette méthode possède un potentiel pas assez exploité pour générer des textes bien plus complexes. Molins et Lapalme (2015) considèrent que l'utilisation de ces structures permet de gagner du temps dans l'analyse syntaxique et de se concentrer directement sur le vocabulaire.

Nous avons donc décidé d'employer la méthode *canned text* pour la génération de nouvelles structures grammaticales qui devraient servir à la production de phrases littéraires. Pour cela, le corpus **LiSSS** (Section 3.3) a été employé. En effet, ce corpus est composé de phrases littéraires portant des caractéristiques linguistiques facilement adaptables pour créer une structure ou gabarit adéquat et générer ainsi de nouvelles phrases. Nous décrivons ci-après le processus pour la génération des nouvelles structures.

4.2.1 Génération des Structures Grammaticales partiellement-Vides

Nous détaillons ici le processus de génération des Structures Grammaticales partiellement-Vides (SGV) pour la génération de phrases littéraires. Le processus commence par la sélection aléatoire d'une phrase f_o venant du corpus **LiSSS**, d'une longueur $|f_o| = N$. f_o sera analysée avec FreeLing pour identifier ses mots lexicaux et les remplacer par leurs étiquettes POS correspondantes.

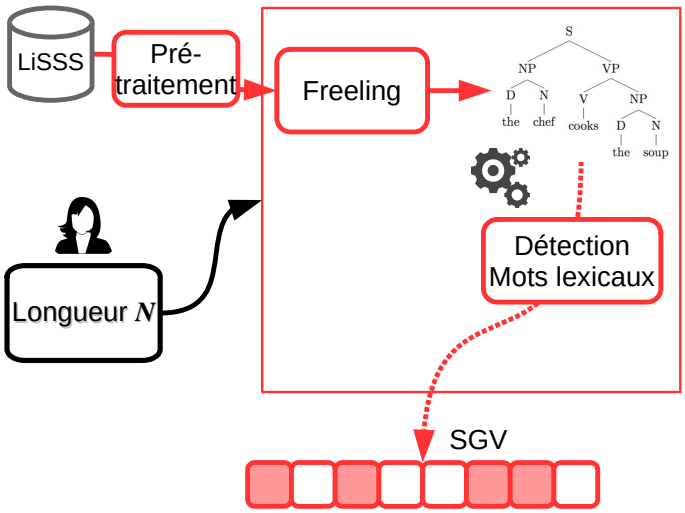


FIGURE 4.3 – *Schéma du processus basé sur canned text pour la génération de Structures Grammaticales partiellement-Vides (SGV).*

Les mots fonctionnels sont préservés sous leur forme d'origine afin de garder la cohérence et le style littéraire. Comme résultat de ce processus, une SGV sera générée. La SGV pourra ensuite être manipulée afin de créer une nouvelle phrase. L'hypothèse sous-jacente est de produire des nouvelles phrases via l'homosyntaxe, c'est-à-dire, possédant une sémantique différente mais ayant la même structure syntaxique.

Le résultat de ce processus est une structure SGV, avec des mots fonctionnels fournissant un support grammatical et des étiquettes POS qui plus tard seront remplacées par un nouveau lexique. Grâce à cette méthode, nous avons réduit la complexité en temps et en ressources tout en améliorant la cohérence syntaxique par rapport à la méthode **Stoch**. L'architecture générale de ce modèle est illustrée dans la figure 4.3. Les cases remplis représentent les mots fonctionnels et les cases vides représentent les étiquettes POS à remplacer. Dans les sections suivantes nous décrivons les différentes analyses sémantiques employées pour la génération des phrases littéraires à partir des SGV produites.

4.2.2 Modèle d'analyse pertinent avec Word2vec (CaP)

Dans cette section nous allons expliquer notre modèle **CaP**. Nous l'avons nommé de cette manière pour indiquer qu'il s'agit d'un modèle basé sur la méthode **ca***nned text* combiné à un modèle **p**ertinent du type Word2vec. **CaP** est basé sur l'analyse sémantique pour le remplacement des étiquettes POS dans les SGV. Ce remplacement est effectué à l'aide du modèle Word2vec entraîné sur le corpus **MegaLite-Es**. L'objectif de cette implémentation est de dépasser la limitation grammaticale des SSV générées par le modèle **Stoch** et de pouvoir

exploiter la richesse du lexique du corpus **MegaLite-Es**. Nous cherchons également à réduire les itérations nécessaires lorsque les étiquettes POS de la SSV ne peuvent pas être remplacées par le lexique $L(Q)$.

Tout d'abord, nous avons créé les ressources linguistiques nécessaires à notre modèle. La première ressource, nommée table associative (TA) a été créée à partir du corpus **MegaLite-Es**. Nous avons analysé ce corpus à l'aide du système Freeling. De cette manière, le vocabulaire complet du corpus a été regroupé en catégories grammaticales. La TA contient des entrées de couples (POS_k, liste de mots associés) qu'on peut représenter comme :

$$\text{POS}_k \rightarrow \vec{V_k} = (v_{k,1}, ..., v_{k,i}, ...),$$

où chaque $v_{k,i}$ représente un mot issu du corpus **MegaLite-Es** appartenant à la catégorie grammatical indiquée par POS_k. Comme indiqué précédemment, dans la SGV les mots fonctionnels et les signes de ponctuation sont conservés. Pour générer une nouvelle phrase, chaque étiquette $\text{POS}_k \in \text{SGV}$, $k = 1, 2, ...$ est remplacée par un nouveau mot. Pour chaque étiquette POS_k, on récupère le vocabulaire associé $v_{k,i}$ à partir de la TA.

Ensuite, on utilise les plongements retournés par Word2vec à partir de la requête Q, pour calculer les distances en cosinus $d(Q, v_{k,i})$ entre chaque mot, $v_{k,i}$, et Q (voir équations (4.4) et (4.5)). Le vocabulaire $v_{k,i}$ est ensuite trié par ordre décroissant en fonction des valeurs calculées par $d(Q, v_{k,i})$. Finalement, l'un des trois premiers éléments est choisi de façon aléatoire (par une distribution uniforme) pour remplacer l'étiquette POS_k de la SGV. Nous lançons l'hypothèse que le hasard peut jouer aussi un rôle important dans la créativité textuelle.

Le résultat est une nouvelle phrase $f_2(Q, N)$ qui n'existe pas dans les corpus **MegaLite-Es** et **LiSSS**. Le processus est illustré dans la figure 4.4. Les résultats de ce modèle seront discutés dans le Chapitre 5. Or, nous avons constaté que le fait d'effectuer des remplacements en se guidant uniquement sur les distances calculées entre Q et $v_{k,i}$ implique parfois des limitations importantes. Ceci peut être plus flagrant quand on utilise directement les valeurs numériques des plongements, sans prendre en compte les contraintes que nous nous sommes imposées. Dans la section suivante, nous proposons une solution à ce problème.

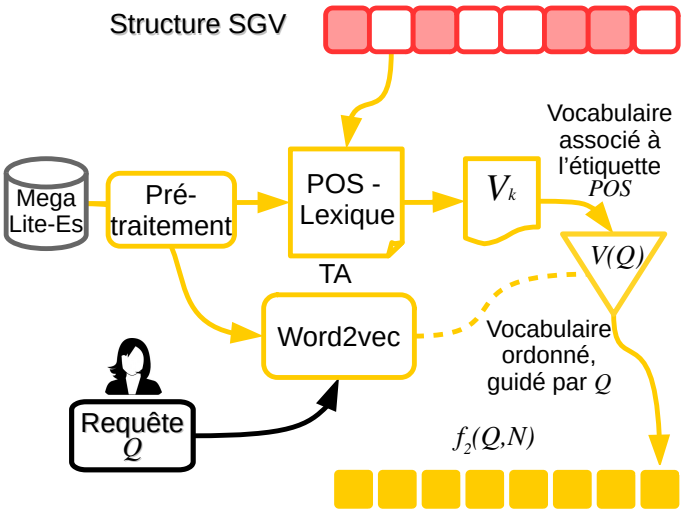


FIGURE 4.4 – *Schéma du modèle **CaP** basé sur une analyse préliminaire de Word2vec.*

4.2.3 Modèle de composition de vecteurs (CaV)

Comme déjà mentionné, l'utilisation directe des valeurs des plongements sans aucun post-traitement peut parfois avoir des limites pour le remplacement des étiquettes POS. Pour l'éviter, nous présentons un deuxième modèle basé sur une nouvelle analyse sémantique pour le remplacement des étiquettes POS. Cette alternative vise surtout l'introduction d'une méthode pour la composition de nouveaux vecteurs à partir des valeurs des plongements calculées par Word2vec. Nous réutilisons plusieurs ressources précédentes : la table associative (TA) et les Structures Grammaticales partiellement-Vides (SGV). Nous avons nommé ce modèle **CaV**, dérivé de « ***ca**nned text* » et de l'idée de la composition des « **v**ecteurs » que nous proposons.

Dans le modèle **CaP**, pour récupérer les mots plus proches du contexte, on calcule les distances entre le vocabulaire produit par Word2vec et Q en utilisant directement les valeurs des plongements. Ensuite, les étiquettes POS de la SGV sont remplacées par les mots ayant les distances les plus courtes par rapport à Q. Pour le modèle **CaV**, nous avons deux points qui serviront de référence pour calculer les nouvelles distances. L'un concerne toujours le contexte Q souhaité dans la phrase ; et l'autre indique le mot dont la sémantique doit être évitée dans la phrase. L'algorithme est décrit ci-dessous, où le processus est itéré pour chaque étiquette POS_k, $k = 1, 2, ... \in$ SGV.

D'abord, un vecteur est construit pour chacun des trois mots suivants :

— o est le mot k de la phrase f_o (Section 4.2.1), qui correspond à l'étiquette POS_k. Ce mot permet de recréer un contexte dont la nouvelle phrase doit s'éloigner, évitant ainsi la production d'une paraphrase ;

— Q définit le contexte fourni par l'utilisateur ;
— w est le mot candidat qui pourrait remplacer l'étiquette POS_k, $w \in V_k$. Le vocabulaire, récupéré de la TA, a une taille $|V_k| = m$ composé par les mots associés à l'étiquette POS_k.

Les 10 mots o_i les plus proches de o, les 10 mots Q_i les plus proches de Q et les 10 mots w_i les plus proches de w (dans cet ordre et obtenus avec Word2vec), sont concaténés et représentés dans un vecteur symbolique $\vec{U}$ de 30 dimensions. Le nombre de dimensions a été fixé à 30 de manière empirique, comme un compromis raisonnable entre diversité lexicale et temps d'exécution. Le vecteur $\vec{U}$ peut être défini comme :

$$\vec{U} = (u_1, u_2, ..., u_{10}, u_{11}, u_{12}, ..., u_{20}, u_{21}, u_{22}, ..., u_{30}) , \tag{4.1}$$

dans lequel les éléments $u_j, j = 1, 2, ..., 10$, représentent les 10 mots les plus proches de o ; $u_j, j = 11, 12, ..., 20$, représentent les mots proches de Q ; et $u_j, j = 21, 22, ..., 30$, sont les mots proches de w. $\vec{U}$ peut alors être réécrit comme suit :

$$\vec{U} = (o_1, ..., o_{10}, Q_{11}, ..., Q_{20}, w_{21}, ..., w_{30}) . \tag{4.2}$$

o, Q et w génèrent respectivement trois vecteurs numériques à 30 dimensions :

$$\begin{aligned} o : \vec{X} &= (x_1, x_2, ..., x_{30}) , \\ Q : \vec{Q} &= (q_1, q_2, ..., q_{30}) , \\ w : \vec{W} &= (w_1, w_2, ..., w_{30}) , \end{aligned} \tag{4.3}$$

où les valeurs de $\vec{X}$ sont obtenues de la distance entre le mot o et chaque mot $u_j \in \vec{U}, j = 1, ..., 30$. La distance, $x_j = d(o, u_j)$ est calculée à partir du modèle Word2vec, dont $x_j \in [0, 1]$. Évidemment, le mot o sera plus proche des 10 premiers mots u_j que des autres.

La même procédure est effectuée pour obtenir les valeurs de $\vec{Q}$ et $\vec{W}$ à partir de Q et w, respectivement. Dans ces deux cas, la requête Q sera plus proche des mots u_j dans les positions $j = 11, ..., 20$ et le mot candidat w sera plus proche des mots u_j dans les positions $j = 21, ...30$.

Ensuite, on calcule les similarités cosinus entre $\vec{Q}$ et $\vec{W}$ (équation 4.4) et entre $\vec{X}$ et $\vec{W}$ (équation 4.5),

$$\theta = \cos(\vec{Q}, \vec{W}) = \frac{\vec{Q} \cdot \vec{W}}{|\vec{Q}||\vec{W}|} \tag{4.4}$$

$$\beta = \cos(\vec{X}, \vec{W}) = \frac{\vec{X} \cdot \vec{W}}{|\vec{X}||\vec{W}|} \tag{4.5}$$

Les valeurs de θ et β sont normalisées entre [0,1]. Le processus est répété pour tous les mots w du lexique V_k. À chaque nouvelle itération, un nouvel ensemble de vecteurs $\vec{X}, \vec{Q}$ et $\vec{W}$ est généré, et les similarités doivent être recalculées. Finalement, on obtient m valeurs de similarité θ_i et β_i, $i = 1, ..., m$.

Les valeurs moyennes $\langle\theta\rangle$ et $\langle\beta\rangle$ sont maintenant calculées. Le quotient normalisé $\left(\frac{\langle\theta\rangle}{\theta_i}\right)$ indique l'importance de la similarité de θ_i par rapport à la moyenne $\langle\theta\rangle$, que l'on souhaite maximiser ; c'est-à-dire, la proximité du mot candidat w par rapport à la requête Q devrait être minimal. Le quotient normalisé $\left(\frac{\langle\beta\rangle}{\beta_i}\right)$ indique la relation entre β_i et $\langle\beta\rangle$, que l'on souhaite minimiser ; c'est-à-dire, la distance qui sépare le mot candidat w du mot o dans f_o doit être maximal. Ces fractions sont obtenues par chaque paire (θ_i, β_i) et sont combinées pour calculer un score S_i, selon l'équation (4.6) :

$$S_i = \left(\frac{\langle\theta\rangle}{\theta_i}\right) \cdot \left(\frac{\beta_i}{\langle\beta\rangle}\right) \tag{4.6}$$

Plus la valeur S_i est élevée, plus elle obéit à l'objectif recherché, qui consiste à se rapprocher de Q et à s'éloigner de la sémantique originale de f_o.

Enfin, la liste des valeurs de S_i est triée par ordre décroissant et le mot candidat w qui remplacera l'étiquette POS_k en question est choisi au hasard parmi les 3 premiers. Le résultat est une nouvelle phrase $f_3(Q, N)$ qui n'existe pas dans les corpus employés pour l'entraînement du modèle. La figure 4.5 montre l'architecture du modèle **CaV**.

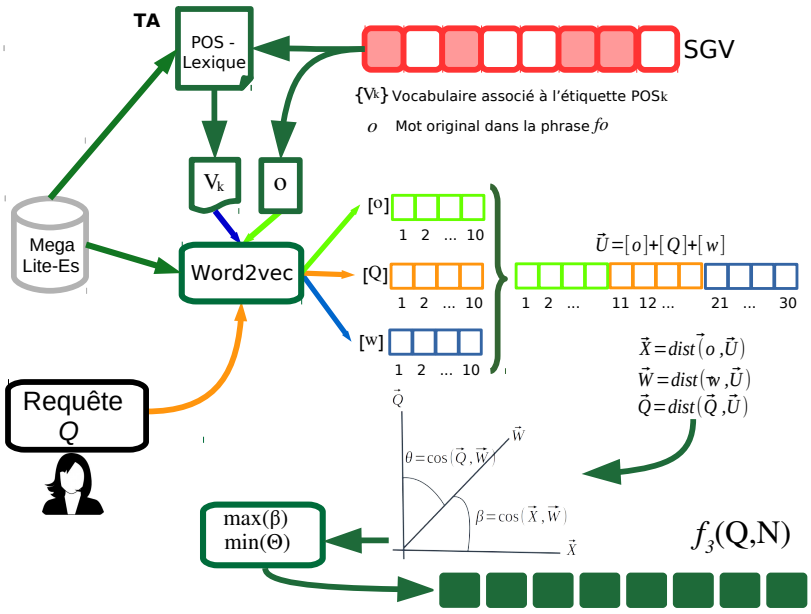


FIGURE 4.5 – *Schéma du modèle **CaV** basé sur Word2vec et compositions des vecteurs.*

Grâce à l'utilisation de vecteurs plus adaptés à nos objectifs, calculés à partir des plongements, le modèle **CaV** génère un plus grand nombre de phrases cohérentes dans des temps d'exécution plus courts comparés à ceux du modèle **CaP**. En même temps, ce modèle permet de traiter les plongements et de les réinterpréter en nouveaux vecteurs afin d'établir des champs sémantiques mieux définis. Dans la section suivante, nous expliquons un modèle basé sur **CaV** pour la production des phrases littéraires guidées par une analyse de sentiments.

4.2.4 Modèle basé sur des traits psychologiques (CaT)

Les résultats obtenus lors des expériences effectuées avec le modèle **CaV** nous ont encouragés à aller plus loin. Nous présentons, dans cette section, le modèle **CaT**. Ce modèle a été conçu pour générer une phrase avec une nouvelle sémantique tout en préservant les émotions exprimées par l'auteur de la phrase originale f_o utilisée pour générer la structure SGV. Nous avons l'hypothèse que les traits psychologiques exprimés dans un texte qui sont associés aux émotions et aux discours d'une personne peuvent être récupérés via une analyse des mots lexicaux. Pour ce modèle, les ressources utilisées dans les modèles des Sections 4.2.2 et 4.2.3 ont été réutilisées : la table associative (TA), l'implémentation du modèle Word2vec et les structures SGV générées par de la méthode *canned text* (Section 4.2.1).

L'algorithme remplace les étiquettes POS correspondant aux substantifs par un vocabulaire proche de la requête Q, tandis que les étiquettes correspondant aux verbes et aux adjectifs sont remplacées par un vocabulaire dont le sens est plus proche des termes originaux de f_o. L'idée est de préserver le style mais aussi le contenu émotionnel exprimé par l'auteur.

Nous illustrons le mécanisme implémenté par l'algorithme **CaT**, avec un cas étudié à partir de l'œuvre de Johann Wolfgang von Goethe. À cet effet, nous avons utilisé des phrases de la version traduite en français du roman *Les souffrances du jeune Werther* pour générer les structures SGV. Nous avons respecté les règles décrites dans la Section 4.2.1 pour choisir les phrases. La TA a été reconstruite en utilisant le mini-corpus **cGoethe** avec des œuvres importantes de Goethe : *Fausto, Le Serpent vers : Conte symbolique, La fiancée de Corinthe, Les souffrances du jeune Werther* et quelques poésies. Les caractéristiques de ce corpus sont montrés dans la table 4.2. Cela nous permettra de générer des nouvelles phrases en utilisant le lexique employé par Goethe.

Nous rappelons que la TA est composée d'entrées du type $\text{POS}_k \rightarrow$ liste de mots $v_{k,i}$, avec la même information grammaticale donné par l'étiquette POS_k. Pour générer une nouvelle phrase, chaque étiquette $\text{POS}_k \in$ SGV est remplacée par un mot sélectionné du lexique $\vec{V}_k$ donné par la TA. Ensuite, pour choisir le mot qui remplacera l'étiquette POS, nous calculons θ et β selon les équations

	Phrases	**Tokens**	**Caractères**
cGoethe	19 519	340 K	2 M
Moyenne par phrase	–	17	103

TABLE 4.2 – *Caractéristiques du corpus **cGoethe**.*

(4.4) et (4.5). Pour cela, nous utilisons les vecteurs définis par (4.2) et (4.3). Ce processus est répété r fois, une fois pour chaque mot $w = v_{k,i}$ dans V_k, et les similarités θ_i et β_i, $i = 1, ..., r$, sont obtenues pour chaque $v_{k,i}$, ainsi que les moyennes $\langle\theta\rangle = \frac{1}{r}\sum\theta_i$ et $\langle\beta\rangle = \frac{1}{r}\sum\beta_i$.

Un score Sn_i est obtenu pour chaque paire (θ_i, β_i) de la manière suivante :

$$Sn_i = \left(\frac{\langle\theta\rangle}{\theta_i}\right) \cdot \left(\frac{\beta_i}{\langle\beta\rangle}\right) \tag{4.7}$$

Plus la valeur de Sn_i est élevée, plus le candidat, $w = v_{k,i}$ se rapproche de Q et se éloigne de la sémantique de f_o. Ce but vise à obtenir le candidat $v_{k,i}$ le plus proche de Q, tout en considérant la sémantique de f_o. Nous utilisons les candidats avec de grandes valeurs de Sn_i pour remplacer les noms.

Pour remplacer les verbes et les adjectifs, nous avons adapté l'équation (4.7) afin d'obtenir le candidat $w = v_{k,i}$ le plus proche de f_o. Nous choisissons donc parmi les candidats ayant les plus grandes valeurs de $S\text{va}_i$, donné par l'équation (4.8) :

$$S\text{va}_i = \left(\frac{\theta_i}{\langle\theta\rangle}\right) \cdot \left(\frac{\langle\beta\rangle}{\beta_i}\right) \tag{4.8}$$

Enfin, nous trions les valeurs de Sn_i (noms) ou $S\text{va}_i$ (verbes et adjectifs) par ordre décroissant et choisissons, de façon aléatoire, parmi les trois valeurs les plus élevées, le candidat $v_{k,i}$ qui remplacera l'étiquette POS_k. Le résultat est une nouvelle phrase générée $f_4(Q, N)$ qui n'existe pas dans les corpus mais qui conserve le contenu émotionnel de f_o. Le modèle est présenté dans la figure 4.6.

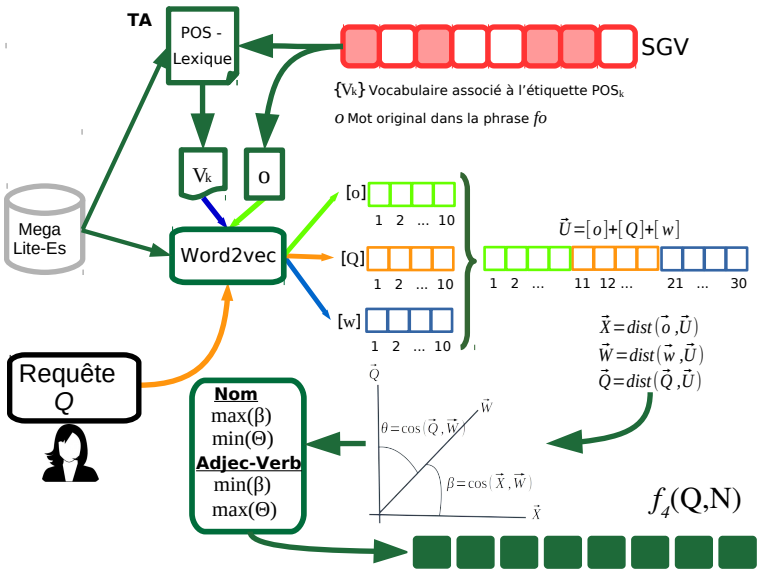


FIGURE 4.6 – *Schéma du modèle **CaT** basé sur l'analyse de traits physiologiques.*

Grâce à l'utilisation minutieuse d'un vocabulaire tiré des œuvres de Goethe et à l'analyse linguistique réalisée par le modèle proposé, nous avons obtenu des résultats très intéressants. Nous pouvons générer des phrases exprimant une idée cohérente dans un nouveau contexte, et en conservant la charge émotionnelle mélancolique observable dans *Les souffrances du jeune Werther*. Les résultats de l'évaluation de ce modèle seront discutés au Chapitre 5.

4.3 Génération de rimes sémantiques (CaR)

Dans cette section nous présentons le modèle **CaR**, pour la génération des phrases contenant la rime comme figure littéraire. L'analyse sémantique suit la même logique du modèle **CaT** avec quelques adaptations importantes. Nous réutilisons les SGV pour représenter les structures syntaxiques et l'implémentation Word2vec pour l'analyse sémantique. Quelques processus ont été simplifiés afin de réduire le temps d'exécution. Par exemple, la table associative n'est plus utilisée, pour pouvoir exploiter tout le vocabulaire fourni par Word2vec. De cette manière, nous avons revisité l'analyse sémantique par composition de vecteurs décrite dans les Sections 4.2.3 et 4.2.4. L'intention est de concevoir un module d'analyse sémantique plus efficace afin de préserver la cohérence au-delà des frontières des phrases.

Pour cette tâche nous avons fait appel au système RIMAX (Urrea et Torres-Moreno, 2019). RIMAX est un système de génération de rime sémantique en espagnol. Il utilise comme ressources le dictionnaire de l'espagnol du Mexique

(DEM) [3] et le dictionnaire de rimes (REM) [4] (Medina Urrea, 2018). Étant donné un terme et son acception (choisis par l'utilisateur), RIMAX produit une liste ordonnée de rimes assonantes et consonnes, ainsi que leurs définitions. La liste est triée en fonction de la similarité sémantique des rimes avec l'acception du terme en entrée. L'algorithme utilisé par RIMAX sera décrit dans la suite.

4.3.1 Rime sémantique

Les dictionnaires de rimes rassemblent les mots selon des patrons de rimes. Les rimes *consonnes* partagent des séquences terminales de sons vocaliques et consonantiques et les rimes *assonantes* partagent des sons vocaliques similaires. Ces deux classes sont donc basées sur des caractéristiques de prononciation, et non sur des modèles d'écriture. De plus, comme la consonance et l'assonance dépendent de la syllabe accentuée, les mots qui se terminent par une syllabe accentuée sont regroupés, ceux dont la syllabe accentuée est l'avant-dernière apparaissent ensemble, et ainsi de suite [5].

Dans l'optique de génération de rimes, a été conçu RIMAX, le premier système automatique de génération de rimes sémantiques en espagnol. Il est constitué des éléments suivants :

— le dictionnaire de l'espagnol du Mexique (DEM) (El Colegio de México, 2022) contenant dans sa version 2021 environ 60 000 entrées ;
— le dictionnaire de rimes de l'espagnol du Mexique (REM), issu du DEM ;
— un algorithme pour mesurer la proximité sémantique écrit en PERL 6.0.

Une procédure similaire pourrait être appliquée à d'autres langues mais dans notre étude nous nous sommes limités à l'espagnol du Mexique.

4.3.2 Production de rimes avec similarité sémantique

Les dictionnaires électroniques offrent des vastes ressources linguistiques exploitables. Des techniques de recherche d'information et de TAL peuvent alors être utilisées pour calculer la similarité sémantique entre les acceptions des termes au moyen des représentations numériques de leurs définitions. Ceci est l'idée de base de RIMAX pour obtenir une similarité sémantique entre les rimes (voir figure 4.7).

À partir d'un dictionnaire D, nous considérons l'ensemble de termes suivant $\{z_1, z_2, z_3, \ldots\}$, et l'ensemble constitué par la définition $\vec{d_j}(z_j)$, de chaque terme

3. Dictionnaire de l'espagnol du Mexique, `https://dem.colmex.mx/`.
4. Dictionnaire de rimes assonantes et consonnes d'espagnol du Mexique.
5. Par exemple, les avant-dernières syllabes des mots espagnols suivants sont les syllabes accentuées : *angula, chula, mula, chamula*. Ces mots devraient donc figurer ensemble dans un dictionnaire de rimes.

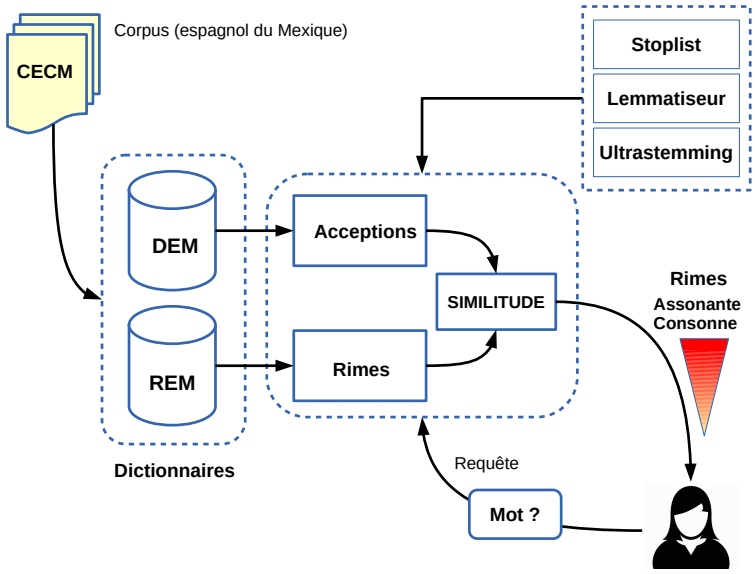


FIGURE 4.7 – *Schéma du système RIMAX.*

z_j, représenté par $\{\vec{d}_1(z_1), \vec{d}_2(z_1), \vec{d}_3(z_3)\ldots\}$. Nous utilisons la notation simplifiée $\vec{d}_j = \vec{d}_j(z_j)$, étant donné que $\vec{d}_j$ est un vecteur dont ses éléments sont les mots qui constituent la définition de z_j. Chaque définition est, donc, représentée par un vecteur $\vec{d}_j = \vec{d}_j(z_j)$ [6].

RIMAX produit un ensemble $\{r_1, r_2, \ldots r_k\}$ de k rimes (assonantes et/ou consonnes) à partir d'un terme z_j. On considère une définition $\vec{d}_i(r_i)$ pour chaque rime r_i, qui peut également être représentée par un vecteur $\vec{d}_i = \vec{d}_i(r_i)$. La similarité entre deux définitions $\vec{d}_j$ et $\vec{d}_i$ peut alors être mesurée à l'aide de la similarité cosinus classique $\cos(\vec{d}_j, \vec{d}_i)$, calculée par les équations (4.4 e (4.5) (Manning et Schütze, 1999). Dans le cas de RIMAX, ce qui est intéressant est de trouver la rime r_j avec la similarité maximale entre les définitions des k rimes et la définition $\vec{d}_j$ du terme z_j, donnée par :

$$\forall z_j, r_j = \max_{1 \leq i \leq k} \{\cos(\vec{d}_j, \vec{d}_i)\} \; ; \; i = 1, 2, \ldots, k \; . \tag{4.9}$$

Ceci garanti la production d'une rime r_j ayant une proximité sémantique raisonnable avec le terme original z_j.

Nous décrivons ci-après l'adaptation non-triviale de RIMAX que nous avons réalisé, pour l'intégrer dans **CaR**, un modèle de génération de phrases rimées. Notre modèle a été implémenté en deux étapes. Dans la première étape, deux

6. Les définitions ont été prétraités et filtrées des *stopwords* et également ultrastemisées (Torres-Moreno, 2012).

structures SGV appropriées sont générées au moyen de *canned text* (voir Section 4.2). Dans l'étape suivante, les SGV seront traitées pour générer une paire de phrases ayant un vocabulaire (rimes comprises) appropriée au contexte.

4.3.3 Première étape : *canned text*

Nous utilisons la méthode *canned text* pour la génération des Structures Grammaticales partiellement-Vides (SGV). Les SGV sont créées à partir d'un ensemble, appelé *TempSet*, composé de phrases sélectionnées manuellement du corpus **MegaLite-Es**, selon les règles suivantes :

— chaque phrase doit exprimer un message clair, sans avoir besoin d'un contexte préalable ;
— chaque phrase doit avoir une longueur de N mots, tel que $5 \leq N \leq 10$;
— chaque phrase doit contenir au moins trois mots lexicaux.

Pour la génération des rimes, le processus commence par la sélection de deux phrases $f1$ et $f2$ de *TempSet*. Ces phrases doivent respecter les conditions suivantes :

— les deux phrases doivent finir par un mot lexical ;
— les mots lexicaux terminant les phrases doivent avoir la même inflexion grammaticale.

$f1$ et $f2$ sont alors analysées avec FreeLing pour détecter les mots lexicaux.

Dans la figure 4.8, nous montrons une illustration du modèle proposé. Les cases remplies représentent les mots fonctionnels et les cases vides représentent les mots lexicaux qui seront remplacés par des étiquettes POS. Une fois que la paire de phrases a été transformée en une SGV, elles seront traitées par la procédure de la deuxième étape.

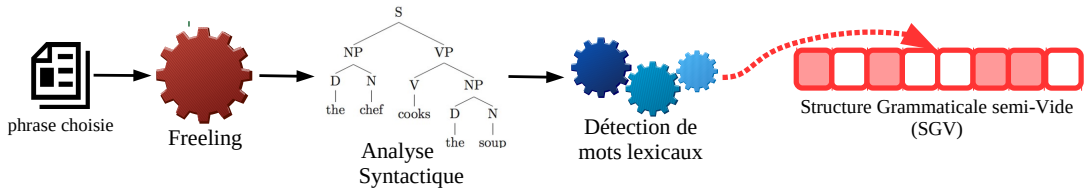


FIGURE 4.8 – *Schéma du modèle **CaR** pour la génération des SGV avec canned text.*

4.3.4 Deuxième étape : sélection du vocabulaire

Dans cette étape, les étiquettes POS des SGV sont remplacées par un vocabulaire produit par le modèle Word2vec.

Analyse sémantique avec Word2vec

Pour le remplacement, nous avons utilisé le raisonnement analogique 3CosAdd introduit par (Drozd et al., 2016). Elle consiste à considérer la relation entre des mots, par exemple « France », « Paris », « Espagne » et un mot manquant x. Supposons que « France », « Paris » et « Espagne » sont des mots qui appartiennent au vocabulaire d'un corpus **CorpA** qui a été utilisé pour entraîner Word2vec. En conséquence, $\overrightarrow{Paris}$, $\overrightarrow{France}$, et $\overrightarrow{Espagne}$ sont les vecteurs correspondants associés à ces mots après l'entraînement.

Le mot x est ensuite déterminé en trouvant un vecteur $\vec{x}$ associé à un mot dans **CorpA**, de telle sorte que $\vec{x}$ soit le plus proche à $\vec{y} = \overrightarrow{Paris} - \overrightarrow{France} + \overrightarrow{Espagne}$, selon la similarité cosinus entre $\vec{y}$ et $\vec{x}$ (équation (4.11)). On considère que la réponse à cet exemple spécifique est correcte si $\vec{x}$ correspond à « Madrid » dans le vocabulaire de **CorpA**. Nous considérons les mots Q, o et A, où :

— Q est le contexte ;
— o est le mot original dans $f1$ ou $f2$ qui est remplacé par l'étiquette POS ;
— A est le mot précédant o, dans la phrase $f1$ ou $f2$, s'il existe.

Ces mots sont représentés par les plongements $\vec{Q}$, $\vec{o}$ et $\vec{A}$, qui sont utilisés pour calculer :

$$\vec{y} = \vec{A} - \vec{o} + \vec{Q}\,, \tag{4.10}$$

où le vecteur $\vec{y}$ possède des caractéristiques enrichies de $\vec{A}$ et $\vec{Q}$ et des caractéristiques de $\vec{o}$ diminuées, de sorte qu'il est plus éloigné de $\vec{o}$. Ensuite, nous gardons les premiers $M = 4\,000$ plongements, les plus proches de $\vec{y}$ selon les distances calculées par le modèle Word2vec, dans une liste $\mathcal{L}$. Plus précisément nous prenons les 4 000 premières sorties de Word2vec lorsque $\vec{y}$ est donné en entrée. $\mathcal{L}$ est donc une liste ordonnée de 4 000 entrées, où chaque ligne, j, correspond à le plongement d'un mot, w_j, associé à $\vec{y}$. La valeur de M a été établie comme un compromis entre le temps d'exécution et la qualité des résultats des expériences conduites (voir Chapitre 5). La tâche suivante consistait à classer les M plongements dans $\mathcal{L}$, en calculant les similarités en cosinus entre le $j^{ième}$ plongement dans $\mathcal{L}$, $\vec{L_j}$, et $\vec{y}$ comme suit :

$$\theta_j = \cos(\vec{L_j}, \vec{y}) = \frac{\vec{L_j} \cdot \vec{y}}{||\vec{L_j}|| \cdot ||\vec{y}||}\ ;\ 1 \leq j \leq M. \tag{4.11}$$

$\mathcal{L}$ est classé par ordre décroissant selon les θ_j.

Si nous remplaçons la première étiquette POS, alors $A = None$, donc nous calculons uniquement $\vec{y} = \vec{o} + \vec{Q}$. Par exemple, pour $Q = amour$ et la phrase $f =$ « *Je joue de la guitare* », nous allons remplacer le verbe fléchi *joue*. Nous calculons alors $\vec{y} = \vec{joue} + \vec{amour}$ pour obtenir la liste classée $\mathcal{L}$. Voici quelques exemples des plongements retournés : *aimer, jouer, enchanter, abandonner*. Cette

liste est ensuite mélangée avec les mots adjacent à o de la phrase f et analysée avec un modèle de langue basé sur des bigrammes, le processus est détaillé à continuation.

Analyse avec un modèle de langue de bigrammes

Une caractéristique importante à considérer lorsque l'on choisit le mot pour remplacer la POS tag est la cohérence. Pour la préserver, nous avons implémenté une analyse de bigrammes, en estimant la probabilité conditionnelle de la présence du $n^{ième}$ mot, w_n, dans une phrase, étant donné qu'un mot, w_{n-1}, est présent.

La probabilité conditionnelle de l'équation (4.12) correspond à la fréquence d'occurrence de chaque bigramme dans **MegaLite-Es**, obtenue à partir de la procédure de détection des n-grammes utilisée lors de la construction de ce corpus (Section 3.2.3). Parmi les bigrammes de **MegaLite-Es**, nous avons considéré seulement les bigrammes formés par des mots lexicaux et fonctionnels (la ponctuation, les chiffres et les symboles ont été ignorés) pour créer la liste LB, utilisée pour calculer les fréquences

$$P(w_n|w_{n-1}) = \frac{P(w_n \wedge w_{n-1})}{P(w_{n-1})}. \tag{4.12}$$

Pour chaque $\vec{L}_j \in \mathcal{L}$, nous formons deux bigrammes, $b1_j$ et $b2_j$, où $b1_j$ est formé par le mot adjacent à gauche de o dans f concaténé avec le mot w_j. Ensuite, $b2_j$ est formé par w_j concaténé avec le mot adjacent à droite de o dans f (voir figure 4.9).

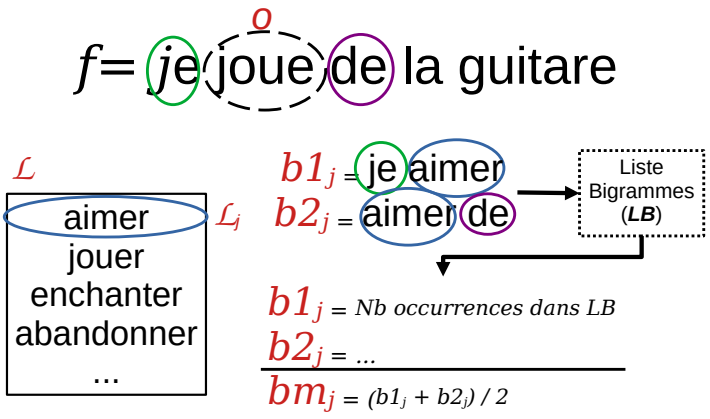


FIGURE 4.9 – *Schéma de la génération de bigrammes.*

Nous calculons ensuite la moyenne arithmétique, bm_j, des fréquences d'occurrence de $b1_j$ et $b2_j$ dans LB. Si o est le dernier mot de f, bm_j est simplement

la fréquence de $b1_j$. La valeur bm_j de chaque $\vec{L}_j$ est ensuite combinée avec la similarité cosinus obtenue avec l'équation (4.11), et la liste $\mathcal{L}$ est reclassée dans l'ordre décroissant selon les nouvelles valeurs

$$\theta_j = \frac{\theta_j + bm_j}{2}, \quad 1 \leq j \leq M. \tag{4.13}$$

Finalement, nous prenons le premier plongement dans $\mathcal{L}$ pour remplacer o. L'idée est de sélectionner le mot sémantiquement le plus proche de $\vec{y}$, basé sur l'analyse effectuée avec Word2vec, tout en conservant la cohérence du texte généré à l'aide de l'analyse linguistique du modèle de langue. Le processus est répété pour chaque mot dans $f1$ et $f2$, sauf lorsqu'on remplace le dernier mot $f2_L$ de $f2$.

Pour remplacer $f2_L$, le mot $f1_L$ en dernière position de $f1$, est envoyé comme entrée au système RIMAX. Il renvoie une liste triée LR ayant des rimes, R_k, consonnes et assonantes liées à $f1_L$. Un score SR_k est attribué à chaque mot R_r dans LR, correspondant à une mesure de similarité sémantique, qui résulte de l'analyse sémantique-phonétique (voir Section 4.3.1). Les scores SR_k sont normalisés dans l'intervalle $[0, 1]$.

Pour chaque $\vec{L}_j \in \mathcal{L}$, on prend le mot w_j correspondant au plongement $\vec{L}_j$, et pour ce w_j, on parcourt la liste LR pour vérifier si $R_k = w_j$ pour chaque $R_k \in LR$. Si $R_k = w_j$ existe, on calcule

$$\theta_j = \frac{\theta_j + SR_k}{2}, \tag{4.14}$$

autrement on calcule

$$\theta_j = \frac{\theta_j}{2}. \tag{4.15}$$

Pour chaque élément $R_k \in LR$ qui n'est pas représenté par un plongement $\vec{L}_j \in \mathcal{L}$ (qui n'est pas égal à un mot w_j), on additionne un nouveau élément, $\theta_i, i > M$, au vecteur $\vec{\theta}$, de sorte que

$$\theta_i = \frac{SR_k}{2}, \tag{4.16}$$

qui fait de θ_i un nouveau score de R_k.

Alors nous prenons l'élément θ_{max} avec le score le plus grand dans $\vec{\theta}$, qui correspond à la meilleure rime w_{max} en considérant la sémantique et la cohérence. Enfin, une analyse morphologique est effectuée avec FreeLing, afin de transformer le mot sélectionné, w_{max}, selon l'inflexion correcte spécifiée par l'étiquette POS correspondant à $f2_L$. Pour cela, nous effectuons des conjugaisons et des conversions de genre ou de nombre.

Le résultat est une nouvelle paire de phrases qui n'existent pas dans le corpus **MegaLite-Es**, où $f2_L$ doit rimer avec $f1_L$. Le modèle est illustré dans la figure 4.10,

4.4 Conclusion

Les modèles expliqués dans ce chapitre ont été développés de manière séquentielle. Chacun des modèles est le produit des points d'amélioration détectés lors de nos expériences. Nous avons abordé les différents aspects que nous nous étions fixés. De la génération par des méthodes stochastiques à l'utilisation de différentes architectures pour l'analyse des émotions et l'incorporation de rimes sémantiques. Pour le modèle stochastique, nous avons perçu la contrainte sémantique pour la sélection du vocabulaire. Pour répondre à cette contrainte, nous avons implémenté un modèle *Word2vec*, avec lequel nous avons mis en évidence le potentiel des plongements lexicaux, avec lequel nous avons proposé un modèle basé sur la composition vectorielle et un autre basé sur l'analogie 3CosAdd. La flexibilité de l'architecture de ces modèles nous a permis d'intégrer la rime sémantique dans la production de phrases littéraires.

En dehors de *Word2vec*, d'autres modèles comme *CamemBERT* (Martin et al., 2020) ont été superficiellement testés. Nous avons constaté que l'avantage de posséder une vaste collection de données pour l'entraînement permet à ces modèles une meilleure analyse sémantique, cependant, les associations sémantiques que l'on trouve souvent dans les textes littéraires était à peine perceptible. Les résultats de nos expériences sont expliqués dans le chapitre suivant.

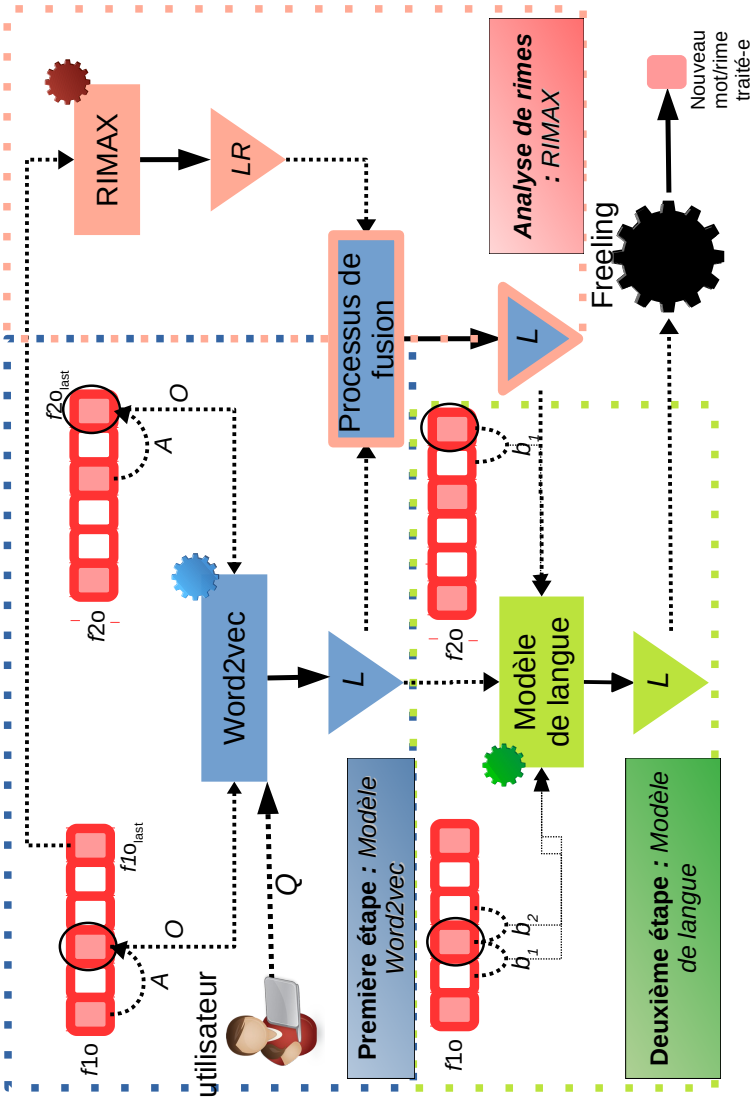


FIGURE 4.10 – *Schéma du modèle **CaR** pour la génération de phrases littéraires rimées.*

Chapitre 5

Expériences de nos modèles génératifs

Sommaire

5.1 Description des expériences 79
5.1.1 Modèle Stoch 79
5.1.2 Modèle CaP 80
5.1.3 Modèle CaV 81
5.1.4 Modèle CaT 82
5.2 Protocoles d'évaluation 83
5.2.1 Évaluation linguistique 83
5.2.2 Évaluation littéraire et émotionnelle 86
5.3 Modèle CaR 88
5.3.1 Évaluation de la rime en espagnol 88
5.3.2 Réalisation des phrases en français 89

Dans ce chapitre, nous présentons les expériences et les protocoles d'évaluations effectuées aux modèles décrits précédemment. Nous montrons des exemples et nous expliquons les ressources employés pour chaque expérience. Ensuite on décrit deux protocoles d'évaluation et on présente les résultats. Le premier protocole a été conçu pour évaluer et comparer les modèles **Stoch, CaP** et **CaV**, car ils ont été développés en suivant les mêmes objectifs (grammaire, cohérence et rapport au contexte). Le deuxième protocole a été configuré pour évaluer les résultats des expériences exécutées par le biais du modèle **CaT** qui a pour objectif l'assimilation et reproduction émotionnelle d'un texte. Finalement, nous montrons des exemples et des résultats de l'évaluation effectuée sur le modèle **CaR**. Nous avons décidé de séparer le modèle **CaR** des autres pour mieux expliquer notre protocole d'évaluation. Ce protocole consiste en deux

étapes, une pour évaluer la production de rimes en espagnol et une autre pour évaluer la production de phrases en français (sans rime).

5.1 Description des expériences

Nous présentons d'abord les ressources employées pour les différents expériences ainsi que des exemples de phrases en espagnol générées pour nos modèles **Stoch, CaP** et **CaV**.

5.1.1 Modèle Stoch

Le modèle **Stoch** a besoin en entrée d'une requête et d'une longueur de la phrase à générer. Le modèle génère une Structure Stochastique Vide (SSV) à l'aide d'une analyse basée sur des chaînes de *Markov*. Ensuite, la SSV sera traitée dans le module d'analyse sémantique guidé par notre implémentation Word2vec afin de produire une phrase f_1.

Ressources employées

Pour la génération des SSV, nous avons utilisé le corpus de phrases en espagnol **8KF** (voir Section 4.1.1). L'entraînement de Word2vec a été effectué sur le corpus **MegaLite-Es** (voir Section 3.1). Nous présentons ci-après quelques exemples des phrases générées dans le format :

$f(Q, N)$ = phrase générée en espagnol (*traduction approximative en français*),

où Q est la requête et N représente la longueur en nombre de mots. Les résultats ont été générés un utilisant les requêtes : **guerre** et **soleil** (*guerra et sol* en espagnol).

Exemples

1. f_1(`GUERRE,12`) = El ejército conquista mediante el enemigo. La batalla es la guerra desde... (*L'armée conquiert à travers l'ennemi. La bataille est la guerre dès...*)

---

2. f_1(`GUERRE,13`) = Toda batalla en rebelión es la guerra contra el ejército en el combate. (*Toute bataille en rébellion est une guerre contre l'armée au combat.*)

---

3. f_1(`SOLEIL,12`) = La luna salvo la lluvia sobre el ocaso hacia el cielo brilla. (*La lune à l'exception de la pluie sur le coucher du soleil vers le ciel brille.*)

4. f_1(SOLEIL, 13) = Cuántos naveguen salvo iluminar para el cielo hacia la aurora es la luna. (*Combien naviguent sauf éclairer par le ciel vers vers l'aurore est la lune.*)

Dans ces exemples on peut observer des phrases plus ou moins cohérentes. Cependant, nous remarquons la première phrase, composée de deux phrases, dont la deuxième donne l'impression d'être incomplète. Cela arrive quand le modèle est incapable de déterminer la fin de la phrase.

5.1.2 Modèle CaP

Ce modèle génère une Structure Grammaticale partiellement-Vide (SGV) à l'aide de la méthode texte en boîte. Il reçoit comme paramètre la longueur de la phrase et le contexte qui sera traité par notre implémentation Word2vec afin de produire une phrase f_2.

Ressources employées

Pour l'entraînement du modèle Word2vec le corpus **MegaLite-Es** a été utilisé, et pour la génération des SGV, nous avons utilisé le corpus **LiSSS** (voir Section 3.3).

En ce qui concerne l'analyse sémantique, nous avons utilisé Word2vec et la Table Associative (TA). La TA contient le vocabulaire du corpus **MegaLite-Es** regroupé selon son inflexion grammaticale (voir Section 4.2.2). Le vocabulaire à utiliser pour générer les phrases est obtenu à partir de cette TA et pondéré en fonction des valeurs de proximité par rapport à la requête donné par l'utilisateur et calculées par Word2vec. Quelques exemples obtenus lors de cette expérience sont affichés ci-dessous.

Exemples

1. f_2(GUERRE, 9) = El incivil comportamiento para la magnificencia es la dicha. (*Un comportement incivil pour la magnificence est la joie.*)

2. f_2(GUERRE, 10) = La cultura es la religión de dogmatizar los bienes caducos. (*La culture est la religion de dogmatiser les biens périmés.*)

3. f_2(SOLEIL, 11) = Brilla que contener siempre. Nunca se es dominado de el todo. (*Il brille pour contenir toujours. Il n'est jamais totalement maîtrisé.*)

4. f_2(SOLEIL, 10) = El rocío exhala el bosque después de haberlo fatigado. (*La rosée exhale la forêt après l'avoir fatiguée.*)

On peut observer que la troisième phrase est une phrase composée. Cela est tout à fait normal, car pour générer les SGV, nous avons employé le corpus **LiSSS**. Ce corpus est composé de phrases individuelles ou de mini paragraphes, contenant deux ou trois phrases.

5.1.3 Modèle CaV

Le modèle **CaV** a été également basé sur la méthode texte en boîte pour la génération des SGV. Cependant, il utilise une ré-interprétation des valeurs calculées par l'implémentation de Word2vec, dont le but est de générer de nouveaux vecteurs. Ces vecteurs sont capables de produire des espaces multi sémantiques contenant plus d'un contexte. Cette approche nous a servit pour générer des phrases s'éloignant d'un contexte tout en se rapprochant à un autre.

Plus spécifiquement, l'intention a été de générer une phrase f_3 qui soit proche de la requête indiquée par l'utilisateur, et loin du contexte produit par la sémantique de la phrase originale. Pour ce modèle, les expériences ont été effectuées avec les mêmes ressources employées que par le modèle **CaP**.

Exemples

1. f_3(GUERRE, 9) = Existe demasiada innovacion en torno a muy pocos sucesos. (*Il y a trop d'innovation autour de trop peu d'événements.*)

---

2. f_3(GUERRE, 9) = En la pelea todo debe motivo, menos la retirada. (*Dans le combat, tout doit [être] un motif, sauf la retraite.*)

---

3. f_3(SOLEIL, 11) = Con rapidez, los monógamos impedimentos buscan para iluminar nos la luz. (*Rapidement, les empêchements monogames cherchent pour nous illuminer la lumière.*)

---

4. f_3(SOLEIL, 10) = Incluso los luceros ingratos son comilones, y por tanto antiguos. (*Même les étoiles ingrates sont gloutonnes, et donc anciennes.*)

Dans ces exemples, nous apprécions des phrases grammaticales et cohérentes. Néanmoins, quelques éléments pourraient être remplacés pour donner une meilleure lisibilité à la phrase. Par exemple, la première partie de la phrase 2 *«En la pelea todo debe motivo»*, on pourrait remplacer le nom *motivo* par le verbe *motivar* (motif -> motiver). Cependant, la sélection du vocabulaire et ses inflexions est limitée pour l'information grammaticale de Freeling via la SGV.

5.1.4 Modèle CaT

Le modèle **CaT** réutilise en principe la même logique que le modèle **CaV**, sauf quelques adaptations qu'ont été expliqués dans la Section 4.2.4. Le but de ce modèle est de générer une nouvelle phrase f_4 avec une sémantique différente mais en préservant la charge émotionnelle de la phrase original.

Ressources employés

Pour l'entraînement du Word2vec le corpus **MegaLite-Es** a été employé. Pour générer la SGV, nous avons utilisé les phrases du roman *Les souffrances du jeune Werther* (von Goethe, 1774) de Johann Wolfgang von Goethe. Nous avons choisi ce roman car il possède une charge émotionnelle important et facile à percevoir. Cela nous a permis de comparer les émotions portées par les phrases produites par notre modèle et celles extraites du roman. Ensuite, le corpus **cGoethe** (voir la table 4.2), composé de plusieurs œuvres du même auteur, a été utilisé pour générer la TA. Les requêtes utilisées pour générer les phrases ont été $Q \in$ {ODIO, AMOR, SOL, LUNA} (en français {HAINE, AMOUR, SOLEIL, LUNE}). Nous montrons quelques exemples de phrases générées en espagnol et une version approximative en français.

Exemples

1. f_4(AMOUR) = Guardando mi deseo, decidí a intentar el sentimiento del placer. (*Gardant mon désir, j'ai décidé d'essayer la sensation de plaisir.*)
2. f_4(AMOUR) = El deseo caía en sus motivos, y los bienes castañeteaban. (*Le désir tombait dans ses motifs, et les marchandises ont claquaient.*)

---

3. f_4(HAINE) = Levantó, y recogió lejos la antipatía con un sentimiento que no traté explicar. (*Il a levé, et a évacué l'antipathie avec un sentiment que je n'ai pas essayé d'expliquer.*)
4. f_4(HAINE) = Muy pocos sentimientos tienen el perfecto odio de pensamientos. (*Très peu de sentiments ont la haine parfaite des pensées.*)

---

5. f_4(SOLEIL) = Muy pocos linderos tienen el perfecto sol de cielos. (*Très peu de frontières ont un parfait soleil de ciels.*)
6. f_4(SOLEIL) = El cierto o desdichado horizonte no atreve a nuestra sombra. (*L'horizon certain ou malheureux n'ose pas notre ombre.*)

---

7. f_4(LUNE) = Tres colores de la sombra principal estaban aun criados en esta luna. (*Trois couleurs de l'ombre principale étaient encore élevées sur cette lune.*)

8. f_4(LUNE) = ¡Tanto dan estas tinieblas mi noble horizonte ! (*Tant d'obscurité donnent à mon noble horizon !*)

Nous pouvons observer que les phrases énumérées ci-dessus contiennent en effet une sémantique associée au contexte établi par l'utilisateur. Même si parfois les phrases sont générées à partir de contextes opposés, par exemple AMOUR et HAINE, nous apprécions une charge émotionnelle similaire, qui obéit à la charge émotionnelle des phrases originales.

5.2 Protocoles d'évaluation

Deux protocoles d'évaluation manuels pour vérifier la qualité des phrases générées en espagnol sont présentés dans cette section. Le premier d'entre eux a été conçu pour effectuer une évaluation des trois modèles **Stoch**, **CaP** et **CaV**. Le deuxième protocole est dédié à l'évaluation des résultats du modèle **CaT**. Nous avons effectué cette séparation car le modèle **CaT** est un cas d'étude du modèle **CaV** avec des conditions particulières de substitution de mots et, par conséquent, les critères d'évaluation ne sont pas les mêmes et n'ont pas été notés avec le même barème.

5.2.1 Évaluation linguistique

Pour chacun des trois modèles, 15 phrases ont été générées, cinq phrases par requête, $Q = \{$AMOUR, GUERRE, SOLEIL$\}$. Au total, nous avons obtenu 45 phrases. Avant d'être présentées aux évaluateurs, les phrases ont été regroupées par requête, en évitant que l'évaluateur puisse savoir de quel modèle vient chaque phrase.

Nous avons demandé à 7 personnes de lire attentivement les 45 phrases (15 phrases par requête). Tous les évaluateurs possèdent des études universitaires et ils sont des hispanophones natifs. Il leur a été demandé de noter sur une échelle de [0,1,2] (où 0=mauvais, 1=bien et 2=très bien) les critères suivants.

— **Grammaticalité :** Est-ce que l'orthographe, les conjugaisons et l'accord du genre et du nombre sont correctes ?
— **Cohérence :** Est-ce que la lisibilité est correcte, et la perception d'une idée générale est claire ?
— **Contexte :** Est-ce que le rapport de la phrase générée avec la requête est acceptable ?

Également, une adaptation du test de Turing a été effectuée. Pour cela, les évaluateurs ont été amenés d'annoter les phrases qui, selon eux, étaient générées par des personnes avec un 0 et celles qui étaient générées par des algo-

rithmes avec un 1. Les résultats de l'évaluation des trois premiers critères sont présentés dans la figure 5.1, où chaque barre représente un critère évalué.

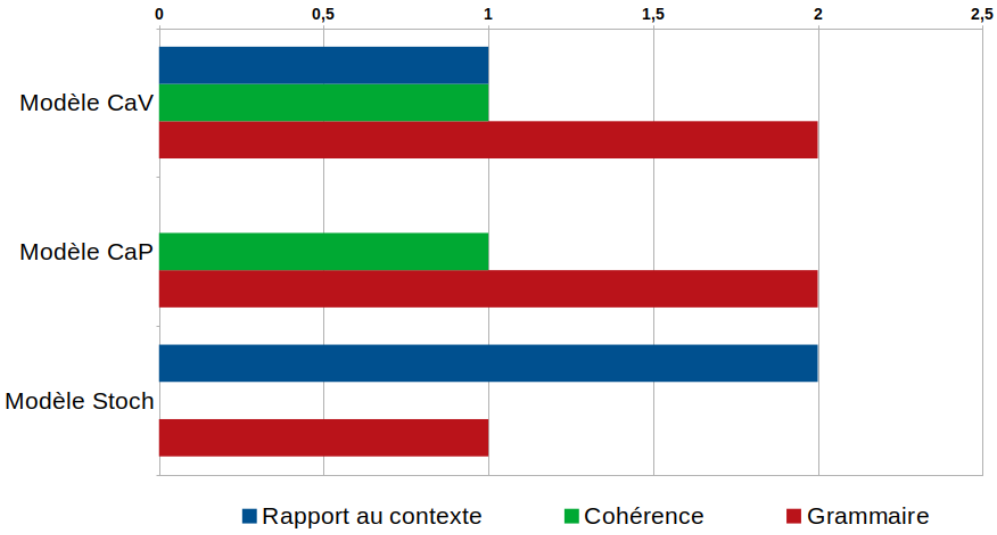


FIGURE 5.1 – *Résultats de l'évaluation d'éléments syntaxiques.*

La partie inférieure de la figure 5.1 montre l'évaluation des phrases générées par le modèle **Stoch**. Elle illustre la perception des évaluateurs vis-à-vis le contexte (barre bleue) et une grammaire acceptable (barre rouge). Cependant, pour ce modèle, les évaluateurs ne perçoivent nullement les phrases comme cohérentes. Nous considérons que la relation avec le contexte s'explique par l'important degré de liberté donné par la SSV. En effet, la SSV permet de remplacer tous les éléments de la structure par un lexique guidé par la requête donné par l'utilisateur.

Dans les résultats du modèle **CaP**, les évaluateurs perçoivent les phrases comme raisonnablement cohérentes et grammaticalement correctes. Cependant, les évaluateurs n'ont pas perçu de relation forte entre le contexte des phrases générées et la requête. En effet, les phrases générées rapportent, pour la plupart, le même contexte ou la même idée que la phrase originale, ce que pourrait être interprété comme une paraphrase élémentaire, ce qui nous cherchons à éviter à tout prix. En revanche, une meilleure grammaire grâce aux SGV générées au moyen de la méthode texte en boîte a été perçue, ainsi qu'une meilleure cohérence.

Enfin, le modèle **CaV**, d'après les évaluateurs, produit des phrases cohérentes, grammaticalement correctes et plus étroitement liées au contexte. Il s'agit du modèle où les trois critères évalués ont eu une perception au moins accep-

table. On y parvient avec la mise en place de l'homosyntaxe, une intuition opposée à la paraphrase où nous cherchons à préserver la structure grammaticale de la phrase originale, tout en générant une sémantique complètement différente. Nous constatons que contrairement aux deux premières modèles, où seuls 2 critères sur 3 étaient clairement perçus, le modèle **CaV** est le seul à avoir obtenu des résultats intéressants dans les trois critères. La mise en place de la méthode texte en boîte et l'approche basée sur le remplacement des éléments grammaticaux au moyen des nouveaux vecteurs calculés à partir des plongements donnés, ont produit du texte original et grammaticalement correct mais éloigné de la sémantique d'origine.

Finalement, dans la figure 5.2 nous montrons les résultats obtenus lors du test de Turing. Le test de Turing a été très fréquemment utilisé dans les recherches d'intelligence artificielle et en Génération Automatique de Texte (GAT) (Turing, 2009). Ce test sert à évaluer dans quel mesure la machine est capable de tromper à l'humain lors d'une comparaison entre des artefacts. Les artefacts comparés sont produits d'un côté par un modèle artificiel et d'un autre par l'humain. La tâche consiste en demander à une personne d'identifier lequel a été produit par qui.

En bleue, on observe les pourcentages des phrases perçues comme étant artificielles, c'est-à-dire, des phrases perçues comme écrites par un algorithme. Tandis qu'en rouge, on observe les pourcentages des phrases perçues comme générées par des humains. Globalement, 44% des phrases générées par le modèle **CaV** ont été perçues comme étant générées par un humain. Il s'agit du meilleur résultat parmi les trois modèles présentés. L'évaluation du modèle **CaT** sera montrée dans la section suivante.

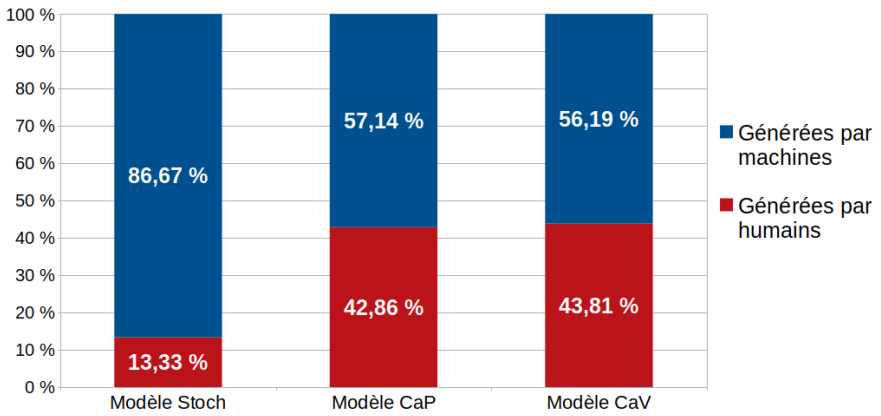


FIGURE 5.2 – *Résultat des phrases générées, évaluées avec le test de Turing.*

5.2.2 Évaluation littéraire et émotionnelle

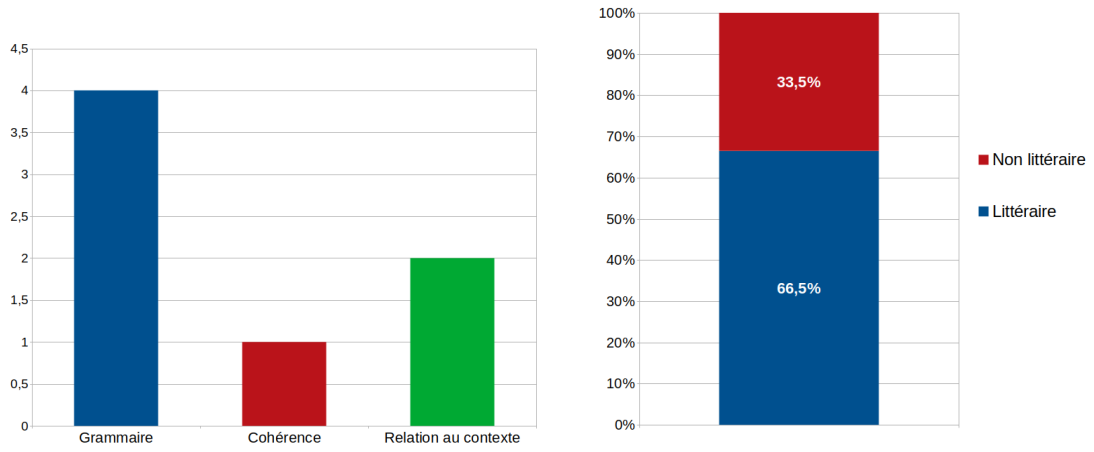
Dans le cadre des expériences de modèle **CaT**, nous proposons une évaluation consistant à produire 5 phrases pour les requêtes $Q = \{$`HAINE, AMOUR, SOLEIL, LUNE`$\}$. Nous avons demandé à 5 évaluateurs de lire attentivement et d'évaluer un total de 20 phrases. Pareil que dans le protocole précèdent, tous les évaluateurs possèdent des études universitaires et sont des hispanophones natifs. Nous leur avons demandé de noter chaque phrase sur une échelle allant de 0 à 4, dont 0 = très mauvais, 1 = mauvais, 2 = acceptable, 3 = bien et 4 = très bien. Les critères employés pour cette évaluation sont les suivants :

— **Grammaire :** orthographe, conjugaisons et accord entre le genre et le nombre ;
— **Cohérence :** lisibilité, perception d'une idée générale ;
— **Contexte :** relation entre la phrase et la requête.

Les résultats sont montrés dans la figure 5.3a. On observe que les deux caractéristiques les mieux notées sont la grammaticalité, qui est en effet le meilleur critère évalué, suivie par la relation au contexte. Pour ce dernier, bien que l'on puisse le considérer comme une caractéristique améliorable, il est important de noter que notre objectif n'était pas seulement d'approcher la sémantique des phrases au contexte mais de garder la charge émotionnelle exprimée dans les textes d'origine. En dernière position, on trouve la cohérence. Pour ce critère, bien qu'il s'agisse d'un résultat défavorable, nous pouvons en déduire que les évaluateurs attendaient des phrases cohérentes et logiques. Cependant, la cohérence en littérature est une caractéristique très subjective, que ne sera pas toujours appréciée de la même manière chez tous les lecteurs.

L'adaptation du test de Turing, effectuée lors du protocole précédent a été remplacé pour un nouveau critère. Cette fois-ci, nous avons demandé aux évaluateurs d'indiquer s'ils considéraient les phrases comme appartenant au genre littéraire. Dans la figure 5.3b nous observons que 66.5% des phrases ont été perçues comme littéraires. Malgré la subjectivité de ce critère, la majorité des évaluateurs ont détecté un dégrée de littérarité dans nos phrases artificielles.

Enfin, nous voulions constater si les phrases générées transmettaient une émotion chez le lecteur et si cette émotion correspondait à ce que l'auteur voulait transmettre dans ses phrases originelles. Pour cela, nous avons demandé aux évaluateurs d'associer une émotion parmi : **Peur, Tristesse, Espoir, Amour,** et **Bonheur**. Dans la figure 5.4, nous pouvons remarquer que les émotions appréciées par les évaluateurs dans les phrases générées sont principalement **Tristesse**, **Espoir** et **Amour**. Rappelons que ces phrases ont été produites à partir des textes du roman *Les souffrances du jeune Werther*, qui possède un contenu émotionnel très proche à ce qui a été produit dans les phrases artificielles.



(a) *Évaluation de critères syntaxiques.* ***(b)*** *Perception de la littérarité.*

FIGURE 5.3 – *Résultats de l'évaluation du modèle* ***CaT****.*

Après une analyse des résultats obtenus, nous considérons que le modèle **CaT** est capable de générer des phrases artificielles avec une bonne perception littéraire, qui même si parfois ne semblent pas cohérentes, elles sont des phrases qui portent des éléments grammaticales corrects et des caractéristiques propres au genre littéraire. Ainsi, ce modèle est capable de gérer une sémantique autour d'un contexte spécifique, tout en gardant comme socle une émotion de base.

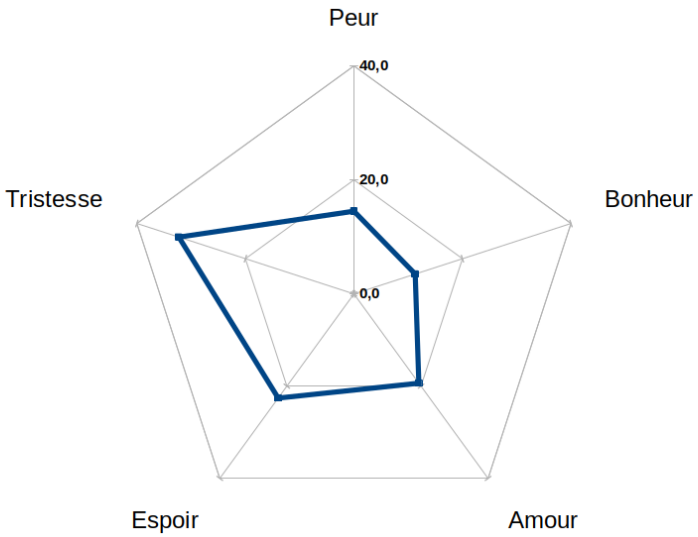


FIGURE 5.4 – *Évaluation de perception émotionnelle du modèle* ***CaT****.*

5.3 Modèle CaR

Pour le modèle **CaR** nous avons implémenté deux protocoles d'évaluation. Le premier protocole a été conçu pour évaluer les rimes en espagnol générées par notre modèle. Étant donné que **CaR** est capable de produire de phrases individuelles, nous avons décidé de générer des phrases en français (sans rime). Un deuxième protocole a été donc configuré afin d'évaluer ces dernières phrases.

5.3.1 Évaluation de la rime en espagnol

Pour cette évaluation nous avons généré 44 couples de phrases rimées, générées à l'aide de SGVs tirées de **MegaLite-Es**. Les SGVs respectent les règles spécifiées dans la Section 4.3. Les contextes ont été définis par les requêtes suivantes : **amour, haine, tristesse, joie, soleil, lune, homme, femme, forêt, désert** et **mer** (*amor, odio, tristeza, alegria, sol, luna, hombre, mujer, bosque, desierto* et *mar*).

Exemples de rimes

Quelques exemples de phrases générées sont présentés ci-dessous. Nous montrons les requêtes et les phrases générées en espagnol en **gras**, et en *italiques* leur traduction. On peut observer que les deux premières phrases du contexte *soleil* ne riment pas. Nous faisons exprès de montrer ces phrases, car les SGVs ne sont pas toujours traitables pour générer des rimes selon nos conditions.

— $Q =$**amour**

1. Jamás he sido más afectuoso ; yo era ya un ofrecimiento. (*Je n'ai jamais été aussi affectueux ; j'étais déjà une proposition.*)
2. Corría el sol rápidamente hacia las cumbres que anteponían el firmamento. (*Le soleil se dirigeait rapidement vers les sommets qui surplombaient le firmament.*)
3. Inspiré pasión por averiguar de qué himnos se atormentaba. (*J'ai inspiré la passion pour enquêter de quels hymnes se torturait.*)
4. Yo volvía y iba sin encontrar jamás lo que hallaba. (*J'allais et retournais sans jamais retrouver ce que je cherchais.*)

— $Q =$**tristesse**

1. El sol de mediodía encapota sobre la inexpresable niebla de mi bosque. (*Le soleil de midi recouvre la brume inexprimable de ma forêt.*)
2. Subía el sol rápidamente hacia las desolaciones que limitaban el zopilote. (*Le soleil montait rapidement vers les désolations qui limitaient le vautour.*)

3. No se apena decir más en menos sílabas. (*Il n'est pas désolé de dire plus en moins de syllabes.*)
4. Zozobré ocultar la angustia que me causaban estas amarguras. (*J'ai hésité à cacher l'angoisse que me causaient ces amertumes.*)

— Q =**soleil**

1. El sol de mediodía brilla sobre la impenetrable aurora de mi boscaje. (*Le soleil de midi brille sur l'aube impénétrable de mon bocage.*)
2. Ascendía el sol rápidamente hacia las estrellas que limitaban el sol. (*Le soleil se levait rapidement vers les étoiles qui limitaient le soleil.*)
3. Bajaba el sol rápidamente hacia las estrellas que limitaban el sol. (*Le soleil se couchait rapidement vers les étoiles qui limitaient le soleil .*)
4. Jamás he sido más ligero ; yo era ya un mirasol. (*Je n'ai jamais été plus léger ; j'étais déjà un tournesol.*)

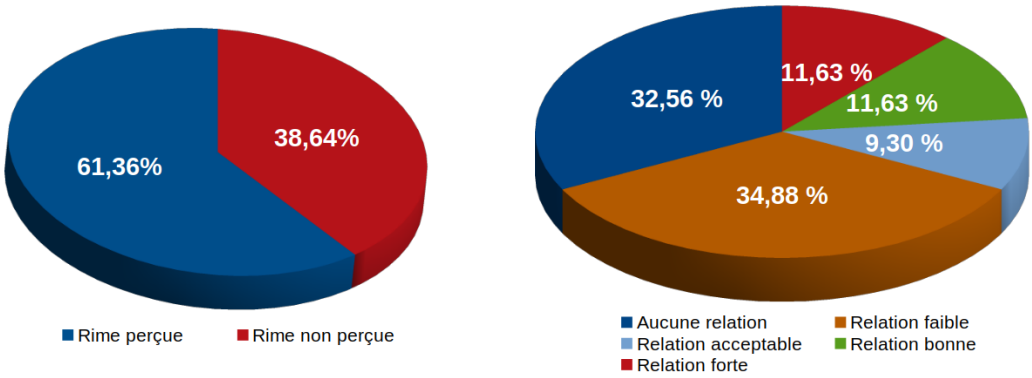
Protocole d'évaluation

Nous avons demandé à 6 évaluateurs, tous diplômés en études littéraires et hispanophones, d'évaluer les rimes et leurs relations sémantiques générées par notre algorithme. Dans nos précédents modèles, des critères tels que la cohérence et la composition grammaticale ont été évalués. Dans cette partie, nous nous sommes focalisé sur l'évaluation de la rime. À cet effet, nous avons demandé aux évaluateurs d'indiquer s'ils percevaient une rime entre les derniers mots de chaque paire de phrases et de préciser également leurs perception concernant la relation sémantique entre les mots rimés. La note d'évaluation pourrait être : **aucune relation**, **relation faible**, **relation acceptable**, **relation bonne** et **relation forte**.

Dans cette évaluation, nous avons obtenu des résultats encourageants avec une perception des rimes dans 61% des paires de phrases évalués (figure 5.5a). Cependant, la relation sémantique entre chaque paire des phrases rimées a été évaluée comme **faible** $= 34.88\%$ et **aucune relation** $= 32.56\%$. Ce résultat n'est pas si étonnant car, lorsque le modèle recherche le deuxième mot en rime, l'analyse sémantique est effectuée en tenant compte non seulement le mot en rime (celui placé à la fin de la première phrase) mais aussi le contexte (donné par la requête) et le mot adjacent à gauche. Pour cette raison, dans certains cas, la relation sémantique entre les deux mots en rimes ne peut pas être garantie.

5.3.2 Réalisation des phrases en français

Nous présentons des exemples et l'évaluation des phrases en français produites par le modèle **CaR**. Nous avons utilisé le corpus **MegaLite-Fr** pour l'en-



(a) *Perception de rimes.* ***(b)*** *Relation sémantique entre des mots rimés.*

FIGURE *5.5* – *Résultat de l'évaluation des phrases rimées en espagnol du modèle* ***CaR****.*

traînement du Word2vec, la génération des SGV et le modèle de langue de bigrammes pour ajouter la grammaire (voir Section 4.3).

Exemples de phrases en français

Quelques exemples de phrases, générées avec trois SGVs différentes, sont présentés ci-dessous sous le format :

[f(`requête`) = phrase].

Dans ces exemples on peut observer des phrases plutôt cohérentes, avec des mots appartenant au même champ sémantique. Quelques erreurs superficiels de syntaxe peuvent y se percevoir. Ces erreurs se sont produits à cause du module de tokenization de Freeling. Une solution au moyen d'une analyse fine à posteriori basée sur des expressions régulières est envisagée.

1. f(`AMOUR`) = Il n'y a pas de passion sans impulsif.
2. f(`AMOUR`) = Il n'y a pas d'affection sans estime.
3. f(`AMOUR`) = Il n'y a ni aimante ni mélancolie en sérénité.
4. f(`AMOUR`) = Il n'y a ni fraternelle ni inquiétude en anxiété

---

5. f(`TRISTESSE`) = En solitude, la première tendresse est la plus forte
6. f(`TRISTESSE`) = Il n'y a pas de confusion sans amour.
7. f(`TRISTESSE`) = Il n'y a pas de liaison sans ordre.
8. f(`TRISTESSE`) = En douleur, la première mélancolie est la plus grande.

---

9. f(`AMITIÉ`) = En union, la première sollicitude est la plus belle.

10. f(AMITIÉ) = Il n'y a ni fraternelle ni faiblesse en impuissance.
11. f(AMITIÉ) = Il n'y a pas de sympathie sans émoi
12. f(AMITIÉ) = Il n'y a ni amie ni honte en peur

Test de Turing

Nous présentons les résultats d'un protocole d'évaluation incluant un test de Turing. Nous nous sommes concentrés dans l'évaluation des phrases individuelles, sans rime ; guidées par un contexte. Quatre critères ont été évalués : grammaire, rapport avec le contexte, littérarité et test de Turing. Nous avons comparé les phrases générées par notre modèle (phrases artificielles) avec des phrases générées par des humains (phrases humaines). Les phrases humaines ont été rédigées en suivant le même processus conçu pour notre modèle, notamment le traitement d'une SGV préalablement générée, puis le remplacement des mots lexicaux guidé par un contexte déterminé.

Le protocole de génération-évaluation a consisté d'abord dans la génération automatique de $p = 70$ phrases artificielles en suivant les contextes : **tristesse, amitié** et **amour**. D'un autre côté, un ensemble de $p' = 230$ phrases ont été rédigées par 18 personnes francophones natives, toutes ayant un niveau d'étude de master. Pour la rédaction de ces phrases, chaque personne a traité un total de 6 SGV (les mêmes structures employées pour notre modèle artificiel).

Nous avons demandé aux humains de rédiger donc p' phrases littéraires par remplacement des éléments grammaticaux dans les SGV. Finalement, les p phrases artificielles ont été mélangées aléatoirement avec les p' phrases manuellement créées et ainsi, un total de $P = 300$ phrases ont été évaluées. Nous avons veillé à ce que les annotateurs n'évaluent pas leurs propres phrases. Pour la grammaire et la littérarité, les évaluateurs devaient indiquer : 0 = mauvais, 1 = bien, selon leurs perception. Le rapport avec le contexte a été évalué dans l'échelle : 0 = mauvais rapport, 1 = bon rapport, 2 = très bon rapport. Finalement, pour le test de Turing les évaluateurs devraient indiquer 0 si la phrase a été générée par une machine, 1 autrement.

Les résultats sont encourageants, avec environ 80% des phrases artificielles aperçues comme grammaticalement correctes. Ceci est un très bon résultat surtout, comparables au 84% obtenu des phrases humaines. L'écart type calculé sur les phrases artificielles (0.22) indique un niveau d'accord acceptable entre les évaluateurs. Pour le critère littérarité, le résultat d'environ 44% des phrases artificielles perçues comme littéraires contre le 73% obtenu par le phrases humaines peut sembler bas mais il est encourageant, étant donné l'ambiguïté de ce qui est perçu comme *littéraire* chez les personnes (voir figure 5.6). Ce résultat indique qu'il y a encore de la marge pour améliorer la littérarité des phrases générées par nos systèmes.

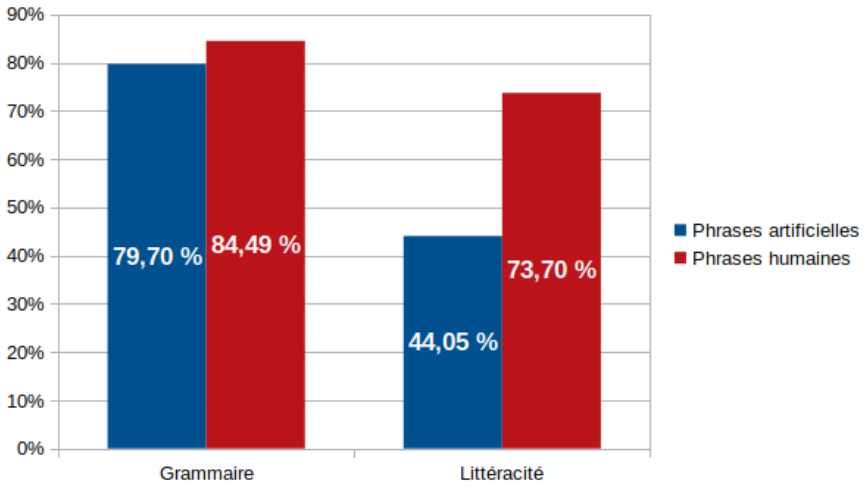


FIGURE 5.6 – *Résultats de l'évaluation des phrases en français du modèle **CaR**.*

Dans la figure 5.7 les résultats du rapport avec le contexte sont affichés. On observe qu'environ 24% des évaluateurs trouvent un mauvais rapport entre les phrases artificielles et leur contexte. En revanche, 36% des évaluateurs ont considéré les phrases ayant un bon rapport et presque 40% un très bon rapport contextuel.

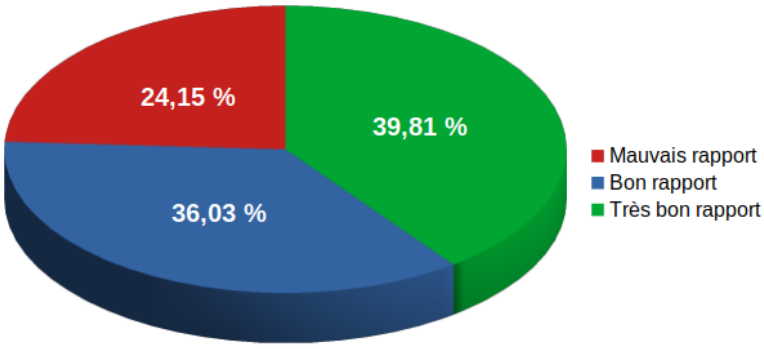


FIGURE 5.7 – *Évaluation du rapport au contexte du modèle **CaR**.*

Finalement, en ce qui concerne le test de Turing, 44% des phrases artificielles ont été perçues comme des phrases générées par des humains. Ceci est un résultat encourageant en considérant que 26% des phrases humaines ont été perçues comme des phrases artificielles (voir figure 5.8). Étant donné que seulement 74% des phrases humaines ont été perçues comme des phrases écrites par des

humains, nous avons 30% de marge d'amélioration pour reproduire la même perception des évaluateurs vis à vis des phrases humaines.

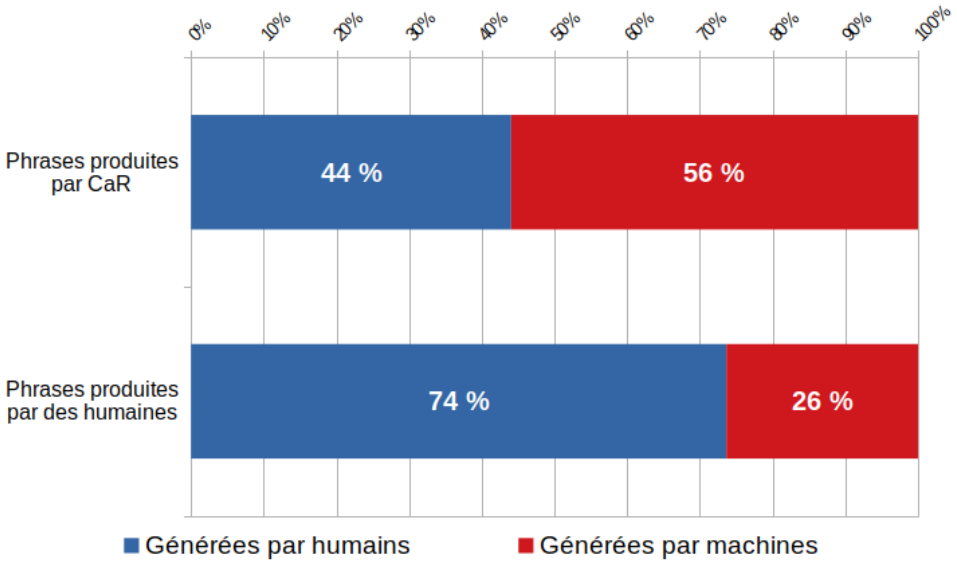


FIGURE 5.8 – *Résultat du test du Turing apliqué au modèle **CaR** (En rouge : phrases perçues comme artificielles. En bleu : phrases perçues comme écrites par des personnes).*

Chapitre 6

Conclusion et perspectives

Dans ce chapitre, nous passerons en revue les principales caractéristiques de nos modèles et les résultats obtenus des expériences, avec des remarques conclusives. Cette explication suivra une ligne chronologique en fonction des expériences et de l'interprétation de nos résultats. Ceci aidera le lecteur à comprendre comment s'est déroulé notre processus de réflexion pour modéliser nos différentes propositions. Enfin, nous esquissons quelques perspectives afin d'améliorer les résultats et d'étendre la portée des objectifs fixés au début de cette thèse.

Conclusions

Lorsqu'on aborde la Créativité computationnelle (CC) pour la modélisation du processus créatif, il est important de comprendre que sont but n'est pas de remplacer les êtres humains dans les différentes disciplines des beaux arts. L'objectif principal de ce domaine de recherche n'est pas non plus de donner solution à une problématique spécifique. Colton et al. (2012) soutiennent dans le livre *Computational creativity : The final frontier ?*, que la CC est un domaine dont l'objectif principal est de proposer de nouveaux paradigmes pour la production d'artefacts de grande valeur artistique, tels que la musique, la sculpture, la peinture ou la littérature, à partir d'une approche computationnelle.

Nous soutenons cette prémisse et, en conséquence, l'objectif de cette thèse focalise dans la proposition de nouveaux algorithmes pour la production de phrases pouvant être perçues comme littéraires. Au début de nos travaux, nous nous sommes fixés le double objectif de génération de phrases et de paragraphes. Nous n'avons pas atteint complètement l'objectif de génération de paragraphes mais nous nous en sommes rapprochés. Ceci est dû, en grande partie à la complexité du texte littéraire et donc à son analyse mais nous avons fait un pas

en avant avec la génération de mini paragraphes. En effet, les modèles ici proposés s'avèrent être très efficaces pour la production de phrases individuelles, mais nous avons réussi à les modifier afin de produire des couples de phrases rimées et sémantiquement liées, avec des résultats encourageants. Les modèles ici présentés sont le produit des réflexions effectuées pendant notre travail de recherche. Des hypothèses et des percées ont été discutées lors de l'analyse des résultats obtenus dans nos expérimentations (Chapitre 5). Au cours de nos études, une dizaine de modèles ont été développés et testés mais seulement les cinq ayant abouti aux meilleurs résultats, tout en répondant aux objectifs de notre recherche, ont été validés et présentés.

Un aspect que nous avons observé au long de ce projet est l'importance de disposer de données adéquates (non seulement en quantité mais aussi en qualité et en pertinence) pour l'entraînement des algorithmes. Étant donné que notre objectif porte sur la génération de phrases littéraires, l'utilisation de corpus non littéraires, pourrait présenter des inconvénients. Les corpus non littéraires bien que volumineux et disponibles, ne possèdent pas de caractéristiques telles qu'un vocabulaire riche ou des structures syntaxiques trouvés fréquemment dans les textes littéraires. L'omission de ces caractéristiques conduirait à un sacrifice important de la littérarité des phrases produites par nos systèmes. C'est ainsi que, nous avons consacré un effort considérable à la conception et construction de corpus de documents exclusivement littéraires : **MegaLite-Es**, **MegaLite-Fr**, **MegaLite-Pt** et **LiSSS** (Moreno-Jiménez et Torres-Moreno, 2021, 2022; Morgado et al., 2022; Moreno-Jiménez et Torres-Moreno, 2020b).

En ce qui concerne la génération de phrases, lors de la première étape de nos recherches, nous avons conçu l'algorithme **Stoch**. Basé sur l'algorithme de Viterbi (Viterbi, 2006), il a montré une capacité acceptable pour saisir la sémantique étant donné un contexte Q fourni par l'utilisateur. **Stoch** interprète assez bien la sémantique à laquelle le vocabulaire d'une phrase artificielle doit être associée. Par contre, nous avons remarqué que la cohérence des phrases générées était difficilement perceptible par les évaluateurs, en raison de deux facteurs :

1. Le manque de contrôle de la ponctuation et des connecteurs rhétoriques (*et, ou, avec, sans, par contre*, etc.) produit des phrases avec des idées non liées et difficiles à lire.
2. La gestion imprécise des mots fonctionnels. Les mots fonctionnels (*il, nous, être, falloir, avoir,* etc.) contiennent une sémantique neutre qui convient de respecter car ils aident le lecteur à se construire une structure logique des idées contenues dans le texte.

Comme résultat, **Stoch** produit des phrases artificielles où l'on n'observe ni des erreurs de genre, ni de nombre ni de conjugaison. Par contre, la lisibilité ou la cohérence sont compromises. Nous montrons quelques exemples ci-après :

— *f(**Guerre**)* → Su política lucha. la rebelión es otra guerra desde la derrota. (*Sa politique lutte. la rébellion est une autre guerre depuis la défaite.*)
— *f(**Amour**)* → Cuánto bien deseo sin la amistad. el afecto es otra ternura. (*combien de bien je souhaite sans l'amitié. L'affection est une autre tendresse.*)

Afin de corriger les problèmes de lisibilité et de cohérence chez les phrases produites par **Stoch**, nous avons développé l'algorithme **CaP**. Ce modèle, basé sur la méthode de texte en boîte (*canned text*), analyse et préserve la structure syntaxique d'une phrase écrite par un humain, tout en modifiant complètement la sémantique. Il s'agit donc de la production de texte par homosyntaxe. **CaP** produit des phrases artificielles grammaticalement correctes, tout en améliorant la lisibilité et surtout le rapport au contexte.

Dans les phrases listées ci-après, les caractéristiques précédemment évoquées peuvent être perçues.

— *f(**Guerre**)* → Mi anciana : tú felicidad no la alumbra ninguna autoridad. (*Ma vieille dame : ton bonheur n'est éclairé par aucune autorité.*)
— *f(**Amour**)* → Estorbas una amada calle de un amor de fantasías. (*Tu bloques une rue aimée d'un amour de fantaisies.*)

Cependant, malgré l'existence d'une relation légèrement étroite entre les phrases produites et le contexte, la cohésion sémantique entre les mots reste très faible. On considère que la cause se trouve dans l'analyse sémantique trop superficielle que a été effectuée, où la contrainte a été de chercher un vocabulaire pertinent et proche du contexte. En conséquence, les mots dans la phrase générée sont effectivement associés au contexte mais liés entre eux avec peu de cohésion sémantique.

Pour diminuer ce problème de cohésion sémantique des phrases produites par **CaP**, nous avons conçu l'algorithme **CaV**. Il cherche à préserver la cohérence et la grammaticalité déjà acquises mais également à maintenir la cohésion sémantique. **CaV** part d'une analyse sémantique en s'appuyant sur deux prémisses pour sélectionner les mots qui composeront une nouvelle phrase. La première, est le rapport au contexte, considéré comme l'axe prédominant auquel la phrase doit être associée. La seconde, est l'association avec les mots lexicaux de la phrase originale. Cette dernière prémisse représente la sémantique dont la phrase à générer doit s'éloigner. Établir les mots lexicaux de la phrase originale comme une référence (à éviter) a permis la production d'une nouvelle phrase, tout en conservant, dans une certaine mesure, la cohésion sémantique entre ses mots. Les phrases listées ci-après sont un exemple des résultats obtenus par **CaV**.

— *f(**Guerre**)* → La retirada es el vapor remediable de la lucha ilimitada. (*Le recul est la vapeur remédiable de la lutte illimitée.*)
— *f(**Amour**)* → En el aprecio está el cariño forzoso de una simpatía. (*Dans l'estime se trouve l'affection forcée d'une sympathie.*)

Ces phrases sont grammaticalement correctes, cohérentes et, selon les évaluateurs (voir Section 5.1), possèdent une cohésion permettant de mieux appréhender l'idée véhiculée.

Dans le but de produire de phrases littéraires en considérant aussi des aspects émotionnels, nous avons conçu le modèle **CaT**. Pour ce modèle, nous avons mis en œuvre une stratégie qui consiste à garder la charge émotionnelle d'une phrase et de la reproduire dans une nouvelle phrase sous un contexte différent. Les phrases présentées ci-après ont été générées à partir du roman *Les souffrances du jeune Werther* de Johann Wolfgang von Goethe.

— *f(**Amour**)* → Un sombrío afecto y una amistad terrible se apoderaron de mí. (*Une sombre affection et une terrible amitié se sont emparées de moi.*)
— *f(**Haine**)* → Un bello cariño y una admiración insoportable se apoderaron de mí. (*Une belle affection et une admiration insupportable m'ont envahi.*)

Ces phrases produisent une charge émotionnelle pessimiste, voir négative chez le lecteur, ce qui était recherché étant donné la stratégie suivie pour la sélection du vocabulaire (voir Section 4.2.4). Par exemple la première phrase montre des constructions telles que : *sombre affection* et *terrible amitié*. On constate que, « affection » et « amitié » sont des noms proches du contexte **Amour**, et « sombre » et « terrible » sont des adjectifs très éloignés de ce contexte mais proches de la sémantique de la phrase d'origine. Les évaluations ont montré que les lecteurs ont perçu dans les phrases évaluées les émotions amour, tristesse et espoir (voir Section 5.1.4). Justement, ce sont des émotions très présentes dans ce roman de Goethe.

La production de phrases littéraires produisant des résultats consistants, nous avons étendu nos études pour nous affranchir de la frontière de la phrase en explorant la génération de mini paragraphes. C'est ainsi que nous avons développé l'algorithme **CaR**, avec l'objectif de produire des phrases rimées. Pour la production de rimes sémantiques, nous avons utilisé le système RIMAX, que nous avons intégré a notre modèle (voir Section 4.3.1). **CaR** considère deux phrases ayant des structures syntaxiques similaires, puis il génère deux nouvelles phrases artificielles où les derniers mots de chaque phrase sont à la fois rimés et liés sémantiquement. Le modèle **CaR** respecte les critères syntaxiques essentiels comme la cohérence, la grammaticalité et la relation avec le contexte.

Quelques exemples de phrases générées sont présentés ci-après.

— *f(**Forêt**)* →
El sol de mediodía tramonta sobre la impenetrable sombra de mi *bosque* ;
Bajaba el sol rápidamente hacia las colinas que limitaban el *zopilote*. (rime assonante)
(*Le soleil de midi traverse les montagnes sur l'impénétrable ombre de ma forêt ; Le soleil se couchait rapidement vers les collines qui limitaient le vautour.*)
— *f(**Soleil**)* → Bajaba el sol rápidamente hacia las estrellas que volvían el *firmamento* ;
Jamás he sido más brusco yo era ya un *ofrecimiento*. (rime consonne)
(Le soleil se couchait rapidement vers les étoiles qui revenaient le firmament, je n'ai jamais été plus brusque ; j'étais déjà une proposition.)

Finalement nous avons testé la génération de phrases (non rimées) en français avec le modèle **CaR**. Le système a été entraîné avec le corpus **MegaLite-Fr**. Pour cette expérience, nous avons mis en place une évaluation basée sur le test de Turing (Turing, 2009). Les évaluateurs ont indiqué que 44% des phrases produites par **CaR** semblaient être écrites par des humains. À l'inverse, parmi les phrases réellement écrites par les personnes, seulement 73% ont été perçues comme telles. Ceci indique qu'une marge d'amélioration de **CaR** de 29% est encore possible pour atteindre les performances humaines.

Ci-après nous montrons quelques exemples des phrases générées.

— *f(**Amour**)* → Il n'y a pas de liaison sans intérêt.
— *f(**Tristesse**)* → Il n'y a ni fraternelle ni inquiétude en anxiété.
— *f(**Amitié**)* → En affection, la première indulgence est la plus belle.

Pour les critères de grammaticalité, cohérence et rapport au contexte, le modèle **CaR** a aussi obtenu des bons résultats. Ceci montre les potentialités de cet algorithme pour générer des phrases ayant des qualités similaires dans les deux langues. Nous avons également remarqué que la cohérence a obtenu le même résultat que le test de Turing, avec 44% de réussite. Cela suggère que pour les évaluateurs, la cohérence est la caractéristique déterminante pour qu'une phrase soit considérée comme écrite par un être humain.

Perspectives

De nos jours, un grand nombre de recherches dans le domaine de Génération automatique de texte (GAT) utilisent des techniques basées sur les réseaux de neurones profonds comme les transformateurs (*transformers*) (Radford et al., 2019). Bien que ces méthodes ont montré des bonnes performances en plusieurs tâches de TAL, leur architecture basée sur des couches cachées reste empirique et complique la compréhension du processus de génération de texte. Par contre, nos modèles combinant des techniques symboliques (*shallow parsing*) et en prenant en compte la sémantique des phrases générées avec des réseaux de neurones, nous permet un niveau d'interprétation du processus créatif dans la GAT. En effet, nous avons montré que la combinaison d'une représentation dense au moyen des plongements de Word2vec avec une méthode de texte en boîte sont très performantes et peuvent être exploitées pour aborder des tâches complexes de production de texte artificiel.

Nos expériences nous ont conduit au développement du modèle **CaR**, capable de produire des phrases en français ou en espagnol, uniquement en adaptant les ressources linguistiques existantes comme Freeling. Pour nous, cette flexibilité et cette adaptabilité ont été une réussite importante. Nous avons mis en évidence les différences entre le français et l'espagnol avec nos approches. En effet, si l'on compare les textes littéraires dans les deux langues, nous pouvons observer que la façon dont les écrivains se servent des structures linguistiques est propre à la langue, et cela donne lieu à des structures syntaxiques complètement différentes. Dans cet esprit, et afin d'enrichir encore plus la production de nos phrases, on pourrait utiliser des techniques d'extraction de synonymes comme celle de Hazem et Daille (2018) mais transposée dans un contexte littéraire. Ceci permettrait d'avoir une diversification du lexique ou du contenu, toujours souhaitable afin d'augmenter la littérarité de nos phrases.

Bien que les résultats pour les critères évalués soient acceptables, un approfondissement du travail est possible. Nous considérons qu'en augmentant le nombre de documents de nos corpus littéraires, et en leur appliquant des analyses plus fines ou profondes (au niveau de syntagmes nominaux ou verbaux par exemple), nous pourrions obtenir des plongements sémantiquement plus représentatifs. Ceci aiderait à réaliser une analyse sémantique plus précise. On pourrait supposer que les modèles déjà pré-entraînés tels que *BERT* (Devlin et al., 2018) pour l'anglais ou *CamemBERT* (Martin et al., 2020) pour le français pourraient aussi s'avérer utiles, si un travail d'adaptation en amont était effectué. Ces modèles ont l'avantage d'avoir été constitués avec des quantités massives de données. Or, n'étant pas des corpus tout à fait littéraires, les performances de nos systèmes pourraient être dégradées. Par conséquent, l'augmentation manuelle ou automatique de la taille des corpus du type **MegaLite** pourrait être une meilleure solution pour la recherche dans le domaine de la

créativité computationnelle. C'est pourquoi nous avons collecté de documents littéraires en portugais, avec l'intention de réaliser des expériences dans cette langue. Les ressources construites dans le cadre de ce projet de thèse ont été mises à disposition sur le site Ortolang [1], afin qu'elles puissent être utilisées par la communauté scientifique (Morgado et al., 2022).

La génération de rimes en français n'a pas été possible en raison du manque de ressources adéquates. Il convient de mentionner que la constitution d'un dictionnaire de rimes dépasse les objectifs de cette thèse. Cependant, nous pensons que cette ressource spécialisée pourrait s'intégrer aisément dans notre chaîne de traitement. Finalement, concernant la génération des paragraphes, nous pensons que des approches comme l'énergie textuelle (Torres-Moreno et al., 2010) pourraient s'avérer intéressantes. L'idée consiste à extraire un ensemble de plusiuers phrases (et donc de structures grammaticales) possédant une charge sémantique similaire (énergies proches) ; puis à partir de ces phrases et structures, un paragraphe de plus de deux phrases pourrait être généré tout en essayant de préserver sa cohésion sémantique.

1. Site web : `https://www.ortolang.fr/market/corpora/megalite/v1`

Annexe A

Caractérisation du corpus MegaLite

La table A.1 montre un résumé des caractéristiques du corpus **MegaLite**. Ce total prend en compte les documents dans les trois langues étudiés (français, espagnol et portugais). Nous avons constaté que ce corpus possède une taille acceptable pour des tâches de TALN, en particulier celles qui demandent de l'apprentissage automatique. Le corpus **MegaLite** est disponible sur le site Ortolang sous licence GPL [1] (Torres-Moreno et al., 2022).

	Documents	**Auteurs**	**Phrases**	**Mots**
MegaLite-Es	5 075	1 329	14,7M	211M
MegaLite-Fr	2 690	620	9,7M	182M
MegaLite-Pt	4 311	1419	21M	252M
Total	12 076	3 368	45,4M	646M

TABLE *A.1* – *Résumé du corpus **MegaLite** ($M = 10^6$).*

Les tables A.2, A.3 et A.4 montrent la distribution détaillée de la constitution du corpus **MegaLite** pour chaque langue. On peut observer les documents regroupés selon la première lettre du nom de chaque document (section 3.1). La version publiée du corpus **MegaLite** (Moreno-Jiménez et Torres-Moreno, 2022) contient uniquement des documents en français et en espagnol. La constitution du corpus littéraire en portugais a eu lieu dans l'étape finale de cette thèse. D'où l'absence d'expériences dans cette dernière langue.

1. Site web : `https://hdl.handle.net/11403/megalite`

ESPAGNOL				
	Documents	**Auteurs**	**Phrases**	**Mots**
Anonyme	33	1	93,4K	1,6M
A	352	84	1,1M	16,6M
B	702	137	1,5M	21,2M
C	509	135	1,4M	20,9M
D	330	53	1,0M	12,7M
E	75	27	236K	3,1M
F	94	39	325K	4,9M
G	239	74	649K	9,3M
H	222	61	844K	10,9M
I	18	9	43K	733K
J	101	24	368K	5,0M
K	161	29	639K	8,7M
L	332	72	930K	13,3M
M	244	107	633K	9,1M
N	49	19	171K	2,0M
O	32	16	98K	1,4M
P	356	81	992K	16,7M
Q	68	10	52K	702K
R	135	57	353K	5,1M
S	489	150	1,4M	20,2M
T	119	38	456K	6,4M
U	24	3	25K	414K
V	209	52	760K	11,1M
W	123	34	418K	5,8M
Y	13	5	37K	640K
Z	46	12	201K	2,9M
Total	5 075	1329	14,7M	211M

TABLE *A.2* – *Distribution des document du corpus **MegaLite-Es** (Moreno-Jiménez et Torres-Moreno, 2021) ($M = 10^6$ et $K = 10^3$).*

FRANÇAIS				
	Documents	**Auteurs**	**Phrases**	**Mots**
Anonyme	8	1	40K	882K
A	93	31	308K	5,8M
B	268	70	748K	14,8M
C	165	58	548K	10,6M
D	400	57	1,5M	30,6M
E	35	10	115K	2,4M
F	111	26	574K	10,0M
G	146	36	572K	10,6M
H	76	26	300K	5,3M
I	18	2	38K	697K
J	12	8	37K	574K
K	26	10	89K	1,6M
L	242	51	806K	14,5M
M	192	46	632K	11,3M
N	30	11	73K	1,2M
O	26	9	96K	1,7M
P	144	28	413K	7,5M
Q	0	0	0	0
R	137	40	461K	8,2M
S	215	47	830K	17,3M
T	74	17	274K	5,1M
U	0	0	0	0
V	152	22	604K	9,7M
W	45	8	184K	3,0M
Y	0	0	0	0
Z	75	6	514K	8,6M
Total	2 690	620	9,7M	182M

TABLE *A.3* – *Distribution des documents du corpus **MegaLite-Fr** (Moreno-Jiménez et Torres-Moreno, 2022) ($M = 10^6$ et $K = 10^3$).*

PORTUGAIS				
	Documents	**Auteurs**	**Phrases**	**Mots**
Anonyme	6	1	1 525	31K
A	757	94	1,5M	21,2M
B	355	124	1,5M	19,6
C	459	135	2M	25,1M
D	271	53	1M	13M
E	36	27	180K	2,3M
F	115	53	679K	8,5M
G	197	86	1M	13M
H	151	62	1,1	14,9M
I	19	8	163K	2,1M
J	69	35	414K	4,9M
K	83	32	908K	10,2M
L	142	79	812K	10,5M
M	314	150	2M	25,6M
N	62	33	241K	3,2M
O	31	14	108K	1,6M
P	188	95	804K	11,9M
Q	58	9	373K	4,6M
R	278	80	1,7M	20,2M
S	431	126	1,6M	19,8M
T	70	34	513K	7,2M
U	3	2	25K	357K
V	120	36	309K	4,3M
W	68	35	501K	6,1M
Y	2	2	12K	172K
Z	26	14	148K	1,8M
Total	4 311	1419	21M	252M

TABLE *A.4* – *Distribution des documents du corpus **MegaLite-Pt** (Morgado et al., 2022) ($M = 10^6$ et $K = 10^3$).*

Annexe B

Exemples des phrases générées

Nous montrons quelques phrases générées avec les différents modèles proposés dans cette thèse. Étant donné que le modèle **CaR** est bien plus performant, nous l'avons utilisé pour générer un nombre bien plus important des phrases (en français). Pour chaque modèle nous avons regroupé les phrases selon leurs requêtes.

Exemples de phrases en espagnol générées par le modèle **Stoch**
Q=Guerra
Su política lucha. La rebelión es otra guerra desde la derrota.
Toda batalla en rebelión es la guerra contra el ejército en el combate.
El ejército conquista mediante el enemigo. La batalla es la guerra desde.
El enemigo salvo la batalla en el terrorismo mediante el ejército conquista contra.
Su atómica lucha. La guerra desde el combate. La derrota es.
Q=Sol
La luna salvo la lluvia sobre el ocaso hacia el cielo brilla.
Cuántos naveguen salvo iluminar para el cielo hacia la aurora es la luna.
Nuestro cielo es verdaderamente el que luna es la lluvia bajo la aurora.
Cuántos perezcas durante amanecer el ocaso bajo la luna.
El resplandor deshoja bajo la aurora.
Q=Amor
Toda amistad contra compasión por la ternura es la pasión en el afecto.
Todos durante la ternura con el afecto hacia la compasión por la virtud.
El cariño con el afecto hacia el amado aborrece salvo en sentimiento.
La ternura envidia entre el cariño. El amado es demasiada compasión.
Cuánto bien deseo sin la amistad. El afecto es otra ternura.

Exemples de phrases en espagnol générées par le modèle **CaP**.
Q=Guerra
El incivil comportamiento para la magnificencia es la dicha.
Mi anciana : tú felicidad no la alumbra ninguna autoridad.
No hay hipocresía más impopular que la historia simulada.
La cultura es la religión de dogmatizar los bienes caducos.
De el temperamento a la entereza hay una velocidad terrible.
Q=Sol
El color que reanima más es una picardía suprema.
La paz es el suelo artificial de la luz moderna.
En el vocabulario está el bosque mixto de una política.
El rocío exhala el bosque después de haber lo fatigado.
Brilla que contener siempre . nunca se es dominado de el todo.
Q=Amor
Estorbas una amada calle de un amor de fantasías.
Jamás hubo una conquista nueva o una amistad extraña.
Dios dejó la desesperacion para trabajar la y no para desilusionar la.
Abultar se en cualquier mentira , es conveniente que no porfiar nada.
Por culpa , el anhelo no suprime siempre con el deseo.

Exemples de phrases en espagnol générées par le modèle **CaV**.
Q=Guerra
Existe demasiada innovación en torno a muy pocos sucesos.
En la pelea todo debe motivo, menos la retirada.
La nueva pelea se combate cuando se abandona la civilización.
La codicia, siempre adversa, es terrible engendrada contra un desgraciado.
La retirada es el vapor remediable de el lucha ilimitada.
Q=Sol
Si tus dulces fueran amanecer, mis ojos marchitas fueran.
Con rapidez , los monógamos impedimentos buscan para iluminar nos la luz.
Incluso los luceros ingratos son comilones, y por tanto antiguos.
La aurora es el amanecer que ha olvidado la calma.
El ocaso es una extraña frente de la inmortalidad.
Q=Amor
Los cariños no conocen de nada a un respeto loco.
No está la simpatía en las bondades de la envidia.
Si el respeto es felicidad , que oculten los cariños.
Acostumbramos de lamentar aquello que se ha enseñado a comprender.

Exemples de phrases en espagnol générées par le modèle **Q=CaL**
Q=Amor
¿ No estarán buscandonos en el sentimiento la amada de Fortunato y los demás ?
Guardando mi deseo, decidí a intentar el sentimiento del placer.
Alegrías lo que nadie busca en la amistad.
Pero no creo que nadie haya prometido contra el buen deseo.
Me dediqué a abandonar mi admiración.
El cierto o buen amor no debe a nuestra ternura.
El deseo caía en sus motivos, y los bienes castañeteaban.
¡ Pienso salir ! No es amada, es amor.
Un sombrío afecto y una amistad terrible se apoderaron de mí.
Q=Odio
Me dediqué a partir mi desconfianza.
Pero no creo que nadie haya prometido contra el buen miedo.
Un bello cariño y una admiración insoportable se apoderaron de mí.
Levantó, y recogió lejos la antipatía con un sentimiento que no traté explicar.
¡ Tanto hacen estas angustias mi feliz sentimiento !
Les consideré sentimiento, los dispuse, los di en admiración y en odio.
Cuántas angustias hay que dan del afecto !
Muy pocos sentimientos tienen el perfecto odio de pensamientos.
Q=Sol
Muy pocos linderos tienen el perfecto sol de cielos.
Un atractivo horizonte y una sombra terrible se apoderaron de mí.
¡ Tanto dicen estas tinieblas mi noble horizonte !
Sólo cuando falta de cielo la luz es difícil querer a los montes.
Gritó, y cayó lejos la luna con un sol que no esforcé creer.
El horizonte no supone nada.
¡Pienso vivir ! No es luz, es sol.
Delante, dije yo, allí está el cielo.
El cierto o desdichado horizonte no atreve a nuestra sombra.
Continuó el sol, algo insólito.
Q=Luna
¡ Tanto dan estas tinieblas mi noble horizonte !
Desmayó, y precipitó lejos la luz con un horizonte que no quise creer.
No sentí soportar la sombra de buscarle algunos montes.
Les dije horizonte, los hice, los quise en sombra y en sol.
¿Había entonces una viva nieve en mi sombra que quedara sin arreglar ?
Mi luz está desgraciada, y me deseo por ti.
Tres colores de la sombra principal estaban aun criados en esta luna.
El seguro o buen horizonte no considera a nuestra sombra.
No, horizonte mío ; no deseo confesar de sus locos árboles.

Exemples des phrases en espagnol générées par le modèle **CaR** et RIMAX
El sol de mediodía palpita sobre la inviolable melancolía de mi *bosque* Retribuía el sol rápidamente hacia las cumbres que anteponían el *zopilote*
No se cesa decir más en menos *frases* Deseé esconder la pasión que me causaban estas *estrofas*
Corría el sol rápidamente hacia las cumbres que anteponían el *firmamento* Jamás he sido más afectuoso ; yo era ya un *ofrecimiento*
Inspiré pasión por averiguar de qué himnos se *atormentaba* Yo volvía y iba sin encontrar jamás lo que *hallaba*
El sol de mediodía palpita sobre la hostil envidia de mi *prado* Traicionaba el sol rápidamente hacia las llanuras que exasperaban el *peinado*
No se teme decir más en menos *frases* Deseé conseguir la aversión que me producían estas *palabras*
Empujaba el sol rápidamente hacia las selvas que exasperaban el *firmamento* Jamás he sido más inmotivado ; yo era ya un *ofrecimiento*
Inspiré hostilidad por comprender de qué folletos se *asustaba* Yo arriesgaba y iba sin llegar jamás lo que *encontraba*

Exemples des phrases en français générées par le modèle **CaR**
Q=Amitié
Il n'y a pas de considération sans charlatanisme.
Il n'y a pas d'intimité sans amour.
Il n'y a pas de sympathie sans attrait.
Il n'y a pas de liaison sans éréthisme.
Il n'y a pas de sympathie sans dégoût.
Il n'y a pas de sympathie sans écoeurement.
Il n'y a pas d'alliance sans embrassement.
Il n'y a pas de sympathie sans idéal.
Il n'y a pas d'union sans bonheur.
Il n'y a ni filiale ni angoisse en agonie.
Il n'y a ni démonstrative ni douleur en horreur.
Il n'y a ni affectueuse ni amertume en ivresse.
Il n'y a ni démonstrative ni compassion en commisération.
Il n'y a ni fraternelle ni crainte en douleur.
Il n'y a ni fraternelle ni douleur en désespérance.
Il n'y a ni affectueuse ni joie en ivresse.
Il n'y a ni prévenante ni haine en impureté.
Il n'y a ni généreuse ni douleur en agonie.
En union, la première ardeur est la plus grande.
En sympathie, la première ardeur est la plus forte.
En tempérante, la première tendresse est la plus belle.
En lesbienne, la première affection est la plus belle.
En familière, la première étreinte est la plus grande.
En affection, la première indulgence est la plus belle.
Il n'y a pas de correspondance sans journal.
Il n'y a pas d'alliance sans idéal.
Il n'y a pas de liaison sans drame.
Il n'y a pas de correspondance sans journal.
Il n'y a pas de considération sans affectif.
Il n'y a pas d'affection sans attachement.
Il n'y a pas d'union sans embrassement.
Il n'y a ni mutuelle ni mélancolie en ivresse.
Il n'y a ni mutuelle ni mélancolie en fiévreuse.
En tendresse, la première indulgence est la plus belle.
En affection, la première étreinte est la plus belle.
En affection, la première douceur est la plus grande.
En affection, la première amante est la plus grande.
En tempérante, la première amante est la plus grande.
En tempérante, la première indolence est la plus belle.
En épouse, la première douceur est la plus forte.
En intimité, la première ardeur est la plus belle

Exemples des phrases en français générées par le modèle **CaR**
Q=Amour
Il n'y a pas de dissemblance sans positivisme.
Il n'y a pas de liaison sans prosélytisme.
Il n'y a pas de liaison sans conspirateur.
Il n'y a pas de liaison sans prosélytisme.
Il n'y a pas de liaison sans auditeur.
Il n'y a pas d'intuition sans altruisme.
Il n'y a pas de passion sans impulsif.
Il n'y a ni fraternelle ni inquiétude en anxiété.
Il n'y a ni tempérante ni tristesse en lassitude.
Il n'y a pas d'intimité sans entourage.
Il n'y a pas de sympathie sans attachement.
En insouciante, la première douceur est la plus forte.
En épouse, la première volupté est la plus forte.
En épouse, la première ivresse est la plus belle.
En bûcheur, la première voluptueuse est la plus belle.
En métaphore, la première voluptueuse est la plus belle.
En bûcheur, la première voluptueuse est la plus belle.
En amoureuse, la première ivresse est la plus forte.
En amoureuse, la première amante est la plus grande.
En fiancée, la première volupté est la plus grande.
En indécise, la première caresse est la plus belle.
En épouse, la première amante est la plus forte.
En métaphore, la première langueur est la plus grande.
Il n'y a pas d'union sans appui.
Il n'y a pas d'amitié sans appui.
Il n'y a pas d'union sans arrêt.
Il n'y a pas d'union sans profit.
Il n'y a pas d'intimité sans apitoiement.
Il n'y a pas d'intuition sans déterminisme.
Il n'y a pas de sympathie sans attendrissement.
Il n'y a pas de passion sans passionné.
Il n'y a pas d'apparence sans poseur.
Il n'y a pas de liaison sans conspirateur.
Il n'y a pas d'intimité sans apitoiement.
En épouse, la première tendresse est la plus forte.
En métaphore, la première effluve est la plus belle.
En épouse, la première volupté est la plus belle.
En bûcheur, la première douceur est la plus grande.
En épouse, la première ivresse est la plus forte.
En amante, la première voluptueuse est la plus belle.
En insipidité, la première ivresse est la plus belle.

Exemples des phrases en français générées par le modèle **CaR**
Q=Tristesse
Il n'y a pas d'amertume sans hymen.
Il n'y a pas de liaison sans ordre.
Il n'y a pas d'affliction sans hymen.
Il n'y a pas de liaison sans intérêt.
Il n'y a ni exultante ni terreur en hécatombe.
Il n'y a pas de déception sans acte.
En miséreuse, la première langueur est la plus forte.
En papelardise, la première langueur est la plus belle.
En miséreuse, la première volupté est la plus forte.
En papelardise, la première douleur est la plus grande.
En songerie, la première langueur est la plus forte.
En miséreuse, la première indolence est la plus belle.
En miséreuse, la première langueur est la plus belle.
En miséreuse, la première rancoeur est la plus grande.
En indécise, la première amertume est la plus grande.
Il n'y a pas de mélancolie sans enveloppement.
Il n'y a pas de contradiction sans défaut.
Il n'y a pas de contradiction sans amour.
Il n'y a pas de confusion sans motif.
Il n'y a pas de contradiction sans amour.
Il n'y a pas de correspondance sans exemple.
Il n'y a pas de divagation sans défaut.
Il n'y a pas de divagation sans intérêt.
Il n'y a pas de confusion sans intérêt.
Il n'y a pas de contradiction sans esprit.
Il n'y a pas de déception sans appât.
Il n'y a pas de confusion sans éclat.
Il n'y a pas d'amertume sans idéal.
Il n'y a pas d'affliction sans aveu.
Il n'y a pas de déception sans acte.
Il n'y a ni filiale ni douleur en étreinte.
Il n'y a ni tempérante ni tristesse en innocence.
En songerie, la première rancoeur est la plus belle.
En miséreuse, la première indolence est la plus forte.
En douleur, la première suavité est la plus forte.
En joie, la première tendresse est la plus forte.
En familière, la première tendresse est la plus grande.
En solitude, la première mélancolie est la plus grande.
En joie, la première langueur est la plus grande.
En émotion, la première douceur est la plus grande.

Liste des illustrations

1.1 Exemples des anciennes et nouvelles pièces d'art. 11
1.2 Artefacts artistiques réalisés par des modèles d'IA. 12

2.1 Illustration des implémentations de modèles Word2vec. 20
2.2 Illustration d'une chaîne de Markov. 21
2.3 Schéma pour la génération des résumés descriptifs de code Java. 22

3.1 Diagramme de divergences Jensen-Shannon 53
3.2 Diagramme d'écart-type calculé des divergences Jensen-Shannon. 54

4.1 Schéma du modèle **Stoch** pour la génération de SSV. 58
4.2 Schéma du modèle **Stoch** basé sur *Markov* et Word2vec. 60
4.3 Schéma du processus pour la génération de SGV. 62
4.4 Schéma du modèle **CaP**, une analyse préliminaire de Word2vec. . 64
4.5 Schéma du modèle **CaV**, Word2vec et compositions des vecteurs. 66
4.6 Schéma du modèle **CaT**, une analyse de traits physiologiques. . . 69
4.7 Schéma du système RIMAX. 71
4.8 Schéma du modèle **CaR** pour la génération des SGV. 72
4.9 Schéma de la génération de bigrammes. 74
4.10 Schéma du modèle **CaR**, génération de phrases littéraires rimées. 77

5.1 Résultats de l'évaluation d'éléments syntaxiques. 84
5.2 Résultat des phrases générées, évaluées avec le test de Turing. . . 85
5.3 Résultats de l'évaluation du modèle **CaT**. 87
5.4 Évaluation de perception émotionnelle du modèle **CaT**. 87
5.5 Résultat de l'évaluation des rimes en espagnol du modèle **CaR**. . 90
5.6 Résultats de l'évaluation des phrases en français du modèle **CaR**. 92
5.7 Évaluation du rapport au contexte du modèle **CaR**. 92
5.8 Résultat du test du Turing appliqué au modèle **CaR** 93

Liste des tableaux

2.1 Résultats de l'évaluation du modèle de Zhang et Lapata. 25
2.2 Précision obtenue par le modèle de Majumder et al.. 29
2.3 Caractéristiques de corpus de citations analysé par Papay et Padó. 32
2.4 Critiques par secteur et par émotion (Navas-Loro et al., 2017) . . . 34

3.1 Caractéristiques du corpus **MegaLite-Es**. 37
3.2 Distribution des documents par genres du corpus **MegaLite-Es**. . 37
3.3 Caractéristiques du corpus **MegaLite-Fr**. 37
3.4 Distribution des documents par genres du corpus **MegaLite-Fr**. . 37
3.5 Caractéristiques du corpus **MegaLite-Pt**. 38
3.6 Distribution des documents par genres du corpus **MegaLite-Pt**. . 38
3.7 Paramètres d'entraînement du modèle *Word2vec*. 40
3.8 Exemples des résultats retournés par le modèle Word2vec. 41
3.9 Exemples des résultats retournés par le modèle Word2vec. 42
3.10 Phrases issues du corpus **MegaLite** versions POS et lemmes . . . 44
3.11 Fréquence de *n*-grammes des corpus **MegaLite-Es/Fr**. 45
3.12 Caractéristiques du corpus **LiSSS**. 47
3.13 Phrases uni-émotion vs multi-émotions du corpus **LiSSS**. 49
3.14 Distribution moyenne des phrases multi-émotions du corpus **LiSSS**. 49
3.15 Caractéristiques utilisés pour l'étude de la divergence de J/S. . . 50

4.1 Caractéristiques du corpus **8KF**. 58
4.2 Caractéristiques du corpus **cGoethe** 68

A.1 Résumé du corpus **MegaLite** 101
A.2 Distribution des document du corpus **MegaLite-Es**. 102
A.3 Distribution des documents du corpus **MegaLite-Fr**. 103
A.4 Distribution des documents du corpus **MegaLite-Pt**. 104

Bibliographie

Agirrezabal, M., B. Arrieta, A. Astigarraga, et M. Hulden (2013). POS-tag based poetry generation with WordNet. In *European Workshop on NLG '13*, Sofia, Bulgaria, pp. 162–166. ACL.

Alsayadi, H. A., A. A. Abdelhamid, I. Hegazy, B. Alotaibi, et Z. T. Fayed (2022). Deep Investigation of the Recent Advances in Dialectal Arabic Speech Recognition. *IEEE Access 10*, 57063–57079.

Arguedas, M., F. Xhafa, L. Casillas, T. Daradoumis, A. Peña, et S. Caballé (2018). A Model for Providing Emotion Awareness and Feedback Using Fuzzy Logic in Online Learning. In *Soft Comput*, Volume 22, pp. 963–977.

Aron, T. (1984). *Littérature et littérarité : un essai de mise au point*, Volume 292. Presses Univ. Franche-Comté.

Barrick, M. R. et M. K. Mount (1991). The big five personality dimensions and job performance : a meta-analysis. *Personnel psychology 44*(1), 1–26.

Barros, C., D. Gkatzia, et E. Lloter (2017). Inflection Generation for Spanish Verbs using Supervised Learning. In *First Workshop on Subword and Character Level Models in NLP*, Copenhagen, Denmark, pp. 136–141. ACL.

Batteux, C. (1824). *Les beaux arts réduits à un même principe*, Volume 1. Impr. d'A. Delalain.

Bednarik, R. G. (1996). The cupules on Chief's Rock, Auditorium Cave, Bhimbetka. *Artefact : the Journal of the Archaeological and Anthropological Society of Victoria, The 19*(1996), 63–72.

Bengio, Y. (2008). Neural net language models. *Scholarpedia 3*(1), 3881. revision #140963.

Bengio, Y., A. Courville, et P. Vincent (2013). Representation learning : A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*(8), 1798–1828.

Boden, M. A. (2004). *The creative mind : Myths and Mechanisms*. Routledge.

Bourgonje, P., Y. Grishina, et M. Stede (2017). Toward a bilingual lexical database on connectives : Exploiting a German/Italian parallel corpus. In *Italian Conference on Computational Linguistics–CLIC-IT*, pp. 53–58.

Bracewell, D. B., F. Ren, et S. Kuriowa (2005). Multilingual single document keyword extraction for Information Retrieval. In *2005 International Conference on Natural Language Processing and Knowledge Engineering*, Wuhan, China, pp. 517–522. IEEE.

Bradley, M. M. et P. J. Lang (1999). Affective norms for English words (ANEW) : Instruction manual and affective ratings. Technical report, Center for research in psychophysiology, University of Florida.

Cabrera-Diego, L. A. et J.-M. Torres-Moreno (2018). SummTriver : A new trivergent model to evaluate summaries automatically without human references. *Data & Knowledge Engineering 113*, 184–197.

Chen, Y. et S. Skiena (2014). Building sentiment lexicons for all major languages. In *52nd Annual Meeting of the ACL*, Volume 2, pp. 383–389.

Clark, E., Y. Ji, et N. A. Smith (2018). Neural Text Generation in Stories Using Entity Representations as Context. In *NACACL-HLT '18*, Volume 1, New Orleans, Louisiana, pp. 2250–2260.

Claude, M. (2004). *La terminologie : principes et techniques*, Volume 4. Presses de l'Université de Montréal.

Clutton-Brock, T., S. West, F. Ratnieks, et R. Foley (2009). The evolution of society. *Philosophical transactions of the Royal Society of London 364*(1533), 3127–3133.

Colton, S., G. A. Wiggins, et al. (2012). Computational creativity : The final frontier ? In *20th European Conference on Artificial Intelligence*, pp. 21–26. ACL.

da Cunha, I., M. T. Cabré, E. SanJuan, G. Sierra, J.-M. Torres-Moreno, et J. Vivaldi (2011). Automatic Specialized vs. Non-specialized Sentence Differentiation. In *12th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2011)*, Tokyo, Japan, pp. 266–276.

Darwin, C. (1909). *The origin of species*. PF Collier & son. New York, USA.

Denkowski, M. et A. Lavie (2014). Meteor universal : Language specific translation evaluation for any target language. In *Ninth workshop on statistical machine translation*, pp. 376–380.

Devlin, J., M. Chang, K. Lee, et K. Toutanova (2018). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR abs/1810.04805*, arXiv :1810.04805.

Dragoni, M., A. G. B. Tettamanzi, et C. da Costa Pereira (2015). Propagating and Aggregating Fuzzy Polarities for Concept-Level Sentiment Analysis. *Cognitive Computation 7*, 186–197.

Drozd, A., A. Gladkova, et S. Matsuoka (2016). Word embeddings, analogies, and machine learning : Beyond king-man+ woman= queen. In *COLING 2016 : Technical papers*, pp. 3519–3530.

Eensoo, E. et M. Valette (2015). Associer heuristiques textométriques et méthodes d'évaluation issues du traitement automatique des langues. *Ela. Etudes de linguistique appliquee 1*(4), 429–436.

El Colegio de México, A. (2022). Diccionario del Español de México (DEM). http ://dem.colmex.mx.

Fu, R., J. Guo, B. Qin, W. Che, H. Wang, et T. Liu (2014). Learning semantic hierarchies via word embeddings. In *52nd Annual Meeting of the ACL*, Volume 1, Baltimore, Maryland, USA, pp. 1199–1209. ACL.

Gervás, P., R. Hervás, et C. León (2015). Generating Plots for a Given Query Using a Case-Base of Narrative Schemas. In *ICCBR*, pp. 103–112.

Goldberg, L. R. (1990). An alternative" description of personality" : the big-five factor structure. *Journal of personality and social psychology 59*(6), 1216.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, et Y. Bengio (2014). Generative adversarial nets. In *NIPS*, pp. 2672–2680.

Hazem, A. et B. Daille (2018). Word Embedding Approach for Synonym Extraction of Multi-Word Terms. In *Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. ELRA.

Hochreiter, S. et J. Schmidhuber (1997). Long Short-Term Memory. *Neural Computation 9*(8), 1735–1780.

Howells, K. et A. Ertugan (2017). Applying fuzzy logic for sentiment analysis of social media network data in marketing. In *9th International Conference on Theory and Application of Soft Computing, Computing with Words and Perception*, Volume 120, pp. 664–670. Elsevier.

Hu, Z., X. Ma, Z. Liu, E. Hovy, et E. Xing (2016). Harnessing deep neural networks with logic rules.

Huang, E. H., R. Socher, C. D. Manning, et A. Y. Ng (2012). Improving word representations via global context and multiple word prototypes. In *50th Annual Meeting of the ACL*, Volume 1, pp. 873–882. ACL.

Indhuja, K. et R. P. C. Reghu (2014). Fuzzy Logic Based Sentiment Analysis of Product Review Documents. In *1st ICCSC*, Trivandrum, India, pp. 18–22. IEEE.

Kayacan, E. et M. A. Khanesar (2016). Chapter 2 - Fundamentals of Type-1 Fuzzy Logic Theory. In E. Kayacan et M. A. Khanesar (Eds.), *Fuzzy Neural Networks for Real Time Control Applications*, pp. 13–24. Butterworth-Heinemann.

Ke, W. et W. Xiaojun (2018). SentiGAN : Generating Sentimental Texts via Mixture Adversarial Networks. In *Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, Stockholm, Sweden, pp. 4446–4452. AAAI Press.

Kharazmi, M. A. et M. Z. Kharazmi (2017). Text coherence new method using word2vec sentence vectors and most likely n-grams. In *3rd Iranian Conference on Intelligent Systems and Signal Processing*, Shahrood, Iran, pp. 105–109. IEEE.

Kiddon, C., L. Zettlemoyer, et Y. Choi (2016). Globally coherent text generation with neural checklist models. In *EMNLP '16*, Austin, Texas, pp. 329–339. Association for Computational Linguistics.

Kullback, S. et R. Leibler (1951). On Information and Sufficiency. *Annals of Mathematical Statistics 22*(1), 79–86.

Kumar, L. A., D. K. Renuka, S. L. Rose, I. M. Wartana, et al. (2022). Deep learning based assistive technology on audio visual speech recognition for hearing impaired. *International Journal of Cognitive Computing in Engineering 3*, 24–30.

Lebret, R., D. Grangier, et M. Auli (2016). Neural text generation from structured data with application to the biography domain. *CoRR abs/1603.07771*, arXiv :1603.07771.

Lecoq, T. (2021). *Les grandes oubliées : Pourquoi l'Histoire a effacé les femmes.* L'iconoclaste, Paris (France).

Lin, C.-Y. (2004). Rouge : A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81.

Majumder, N., S. Poria, A. Gelbukh, et E. Cambria (2017). Deep learning-based document modeling for personality detection from text. *IEEE Intelligent Systems 32*(2), 74–79.

Manning, C. D. et H. Schütze (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, Massachusetts : The MIT Press.

Martin, L., B. Muller, P. J. O. Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, et B. Sagot (2020). CamemBERT : a Tasty French Language Model. In *58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Martínez, G. S. (2018). Introducción a los corpus lingüísticos. *Series del Instituto de Ingeniería, UNAM 1*, 0.

Matiko, J. W., S. P. Beeby, et J. Tudor (2014). Fuzzy logic based emotion classification. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4389–4393. IEEE.

McRoy, S. W., S. Channarukul, et S. S. Ali (2003). An augmented template-based approach to text realization. *Natural Language Engineering 9*(4), 381.

Medina Urrea, A. (2018). *Diccionario de rimas asonantes y consonantes del español de México*. El Colegio de México, Mexico.

Mikolov, T., K. Chen, G. Corrado, et J. Dean (2013a). Efficient Estimation of Word Representations in Vector Space. In Y. Bengio et Y. LeCun (Eds.), *ICLR '13*, Scottsdale, Arizona, USA. ICLR.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, et J. Dean (2013b). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc.

Mikolov, T., W.-t. Yih, et G. Zweig (2013c). Linguistic regularities in continuous space word representations. In *NACACL-HLT '13*, Atlanta, USA, pp. 746–751.

Mikolov, T. et G. Zweig (2012). Context dependent recurrent neural network language model. In *2012 IEEE Spoken Language Technology Workshop (SLT)*, Miami, FL, USA, pp. 234–239. IEEE.

Minu, R., G. Nagarajan, S. Borah, et D. Mishra (2022). LSTM-RNN-Based Automatic Music Generation Algorithm. In *Intelligent and Cloud Computing*, pp. 327–339. Springer.

Mohammad, S. M. et P. D. Turney (2013). NRC emotion lexicon V2. Technical report, National Research Council, Canada.

Molins, P. et G. Lapalme (2015). JSrealB : A Bilingual Text Realizer for Web Programming. In *ENLG '15*, Brighton, UK, pp. 109–111. ACL.

Montfort, N. (2008a). Computational Poems & Related Digital Projects. The Two. https://nickm.com/2/the_two.html.

Montfort, N. (2008b). Computational Poems & Related Digital Projects. Through the Park. https://nickm.com/poems/through_the_park.html.

Montfort, N. (2009). Computational Poems & Related Digital Projects. Taroko Gorge. https://nickm.com/poems/taroko_gorge.html.

Nadali, S., M. Murad, et R. Kadir (2010). Sentiment classification of customer reviews based on fuzzy logic. In *2010 International Symposium on Information Technology*, Volume 2, pp. 1037–1044. IEEE.

Navas-Loro, M., V. Rodríguez-Doncel, I. Santana-Pérez, et A. Sánchez (2017). Spanish Corpus for Sentiment Analysis Towards Brands. In *International Conference on Speech and Computer*, pp. 680–689.

Oliveira, H. G. (2012). PoeTryMe : a versatile platform for poetry generation. In *Computational Creativity, Concept Invention and General Intelligence*, Volume 1, Osnabrück, Germany. Institute of Cognitive Science.

Oliveira, H. G. (2017). A survey on intelligent poetry generation : Languages, features, techniques, reutilisation and evaluation. In *ICNLG '17*, pp. 11–20.

Oliveira, H. G. et A. Cardoso (2015). Poetry Generation with PoeTryMe. In *CCR-TCM '15*, Volume 7, Paris. Atlantis Thinking Machines.

Oruh, J., S. Viriri, et A. Adegun (2022). Long Short-Term Memory Recurrent Neural Network for Automatic Speech Recognition. *IEEE Access 10*, 30069–30079.

Padró, L. et E. Stanilovsky (2012). FreeLing 3.0 : Towards Wider Multilinguality. In *8th on LREC '12*, Istanbul, Turkey, pp. 2473–2479.

Papay, S. et S. Padó (2020). RiQuA : A Corpus of Rich Quotation Annotation for English Literary Text. In *12th Language Resources and Evaluation Conference*, Marseille, France, pp. 835–841. European Language Resources Association.

Papineni, K., S. Roukos, T. Ward, et W.-J. Zhu (2002). Bleu : a method for automatic evaluation of machine translation. In *40th annual meeting of the Association for Computational Linguistics*, pp. 311–318.

Pennebaker, J. W. et L. A. King (1999). Linguistic styles : language use as an individual difference. *Journal of personality and social psychology 77*(6), 1296.

Pérez y Pérez, R. (2015). *Creatividad Computacional*. México : Larousse - Grupo Editorial Patria.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog 1*(8), 9.

Ramesh, A., P. Dhariwal, A. Nichol, C. Chu, et M. Chen (2022). Hierarchical Text-Conditional Image Generation with CLIP Latents.

Řehůřek, R. et P. Sojka (2010). Software Framework for Topic Modelling with Large Corpora. In *LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, pp. 45–50. ELRA.

Riedl, M. O. et R. M. Young (2006). Story planning as exploratory creativity : Techniques for expanding the narrative search space. *New Generation Computing 24*(3), 303–323.

Rosso, O. A., H. Craig, et P. Moscato (2009). Shakespeare and other English Renaissance authors as characterized by Information Theory complexity quantifiers. *Physica A Statistical Mechanics and its Applications 388*(6), 916–926.

Sharples, M. (1996). *How We Write : Writing as creative design*. London, UK : Routledge.

Siddiqui, M., R. S. Wedemann, et H. J. Jensen (2018). Avalanches and generalized memory associativity in a network model for conscious and unconscious mental functioning. *Physica A : Statistical Mechanics and its Applications 490*, 127–138.

Singhal, A. et al. (2001). Modern information retrieval : A brief overview. *IEEE Data Eng. Bull. 24*(4), 35–43.

Spärck-Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28*, 11–21.

Sridhara, G., E. Hill, D. Muppaneni, L. Pollock, et K. Vijay-Shanker (2010a). Towards Automatically Generating Summary Comments for Java Methods. In *IEEE/ACM International Conference on Automated Software Engineering*, Antwerp, Belgium, pp. 43–52. ACM.

Sridhara, G., E. Hill, D. Muppaneni, L. Pollock, et K. Vijay-Shanker (2010b). Towards Automatically Generating Summary Comments for Java Methods. In *IEEE/ACM International Conference on Automated Software Engineering*, Antwerp, Belgium, pp. 43–52. ACM.

Stymne, S. et C. Östman (2020). SLäNDa : An Annotated Corpus of Narrative and Dialogue in Swedish Literary Fiction. In *LREC Conference*, Marseille, pp. 826–834. European Language Resources Association.

Szymanski, G. et Z. Ciota (2002). Hidden Markov Models Suitable for Text Generation. In N. Mastorakis, V. Kluev, et D. Koruga (Eds.), *WSEAS '02*, Athens, pp. 3081–3084. WSEAS - Press.

Tashtoush, Y. M. et D. A. Al Aziz Orabi (2019). Tweets Emotion Prediction by Using Fuzzy Logic System. In *6th International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pp. 83–90. IEEE.

Todorov, T., L. Moss, et B. Braunrot (1973). *The notion of literature*, Volume 5. JSTOR.

Torres-Moreno, J.-M. (2012). Beyond Stemming and Lemmatization : Ultra-stemming to Improve Automatic Text Summarization. *ArXiv abs/1209.3126*, 1209–3126.

Torres-Moreno, J.-M. (2014). *Automatic Text Summarization*. London, UK, Hoboken, USA : ISTE, Wiley.

Torres-Moreno, J.-M., A. Molina, et G. Sierra (2010). La energía textual como medida de distancia en agrupamiento de definiciones. In *International Conference on Statistical Analysis of Textual Data*, Rome, Italy.

Torres-Moreno, J. M., L. Moreno, et I. Morgado (2022). MEGALITE. https ://hdl.handle.net/11403/megalite/v1. ORTOLANG (Open Resources and TOols for LANGuage).

Turing, A. M. (2009). Computing machinery and intelligence. In *Parsing the Turing test*, pp. 23–65. Springer.

Urrea, A. M. et J.-M. Torres-Moreno (2019). RIMAX : Ranking Semantic Rhymes by calculating Definition Similarity.

van Deemter, K., M. Theune, et E. Krahmer (2005). Real versus Template-Based Natural Language Generation : A False Opposition ? *Computational Linguistics 31*(1), 15–24.

Vashishtha, S. et S. Susan (2020). Inferring Sentiments from Supervised Classification of Text and Speech cues using Fuzzy Rules. *Procedia Computer Science 167*, 1370 – 1379.

Villena-Román, J., S. Lana-Serrano, E. Martínez-Cámara, et J. C. González-Cristóbal (2013). TASS - Workshop on Sentiment Analysis at SEPLN. *Procesamiento del Lenguaje Natural 50*(0), 37–44.

Viterbi, A. J. (2006). A personal history of the Viterbi algorithm. *IEEE Signal Processing Magazine 23*(4), 120–142.

von Goethe, J. W. (2006, First edition 1774). *The Sorrows of Young Werther*. London, England : Penguin.

Wedemann, R. S. et A. R. Plastino (2016). Física Estadística, Redes Neuronales y Freud. *Revista Núcleos 3*, 4–10.

Yang, L. et A. Coxhead (2020). A Corpus-based Study of Vocabulary in the New Concept English Textbook Series. *RELC Journal 0*(0), 0033688220964162.

Zhang, X. et M. Lapata (2014). Chinese Poetry Generation with Recurrent Neural Networks. In *EMNLP '14*, Doha, Qatar, pp. 670–680. ACL.

Bibliographie personnelle

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, H. Boucheneb, et R. S. Wedemann (2020b). FLE : A Fuzzy Logic Algorithm for Classification of Emotions in Literary Corpora. In *IC3K 2020) – Volume 1 : KDIR*, pp. 202–209.

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, N. A. Castro-Sánchez, A. Nava-Zea, et G. Sierra (2017). Criminal Events Detection in News Stories Using Intuitive Classification. In *Mexican International Conference on Artificial Intelligence*, pp. 120–132. Springer.

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, et R. S. Wedemann (2020a). Literary Natural Language Generation with Psychological Traits. In *NLPIS '20, LNCS*, Volume 12089, Cham, pp. 193–204. Springer.

Moreno-Jiménez, L. G., J. M. Torres-Moreno, R. S. Wedemann, et E. SanJuan (2020). Generación automática de frases literarias. *Linguamática 12*(1), 15–30.

Moreno-Jiménez, L.-G. et J.-M. Torres-Moreno (2020a). Génération de phrases littéraires : une approche combinant Traitement Automatique des Langues et Apprentissage automatique. *Sphères* (1), 106–118.

Moreno-Jiménez, L.-G. et J.-M. Torres-Moreno (2020b). LiSSS : A New Multi-annotated Multi-emotion Corpus of Literary Spanish Sentences. *Computación and Sistemas 24*, 1139–1147.

Moreno-Jiménez, L.-G. et J.-M. Torres-Moreno (2021). Megalite : A New Spanish Literature Corpus for NLP Tasks. In D. N. E. David C. Wyld (Ed.), *8th International Conference on Artificial Intelligence and Applications (AIAP 2021)*, Zurich, Switzerland.

Moreno-Jiménez, L.-G. et J.-M. Torres-Moreno (2022). MegaLite-2 : An Extended Bilingual Comparative Literary Corpus. *Intelligent Computing. Lecture Notes in Networks and Systems 283*, 1014–1029.

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, C.-E. Gonzalez-Gallardo, et R. Wedemann (2021b). Estudio de hiperparámetros de modelos neuronales en la

generación de frases literarias. In *Research in Computing Science*. Congreso Mexicano de Inteligencia Artificial (COMIA).

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, et R. Wedemann (2020). Generación de Frases Literarias : un experimento preliminar. In *36th Annual SEPLN Congres*, Number 65, pp. 29–36. SEPLN.

Moreno-Jiménez, L.-G., J.-M. Torres-Moreno, et R. Wedemann (2021a). A Preliminary Study for Literary Rhyme Generation based on Neuronal Representation, Semantics and Shallow Parsing. In *Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, Porto Alegre, RS, Brasil, pp. 190–198. SBC.

Morgado, I., L. Moreno-Jiménez, J.-M. Torres-Moreno, et R. Wedemann (2022). MegaLite-PT : A Corpus of Literature in Portuguese for NLP. À paraître. In *BRACIS 2022*.

Torres-Moreno, J.-M. et L.-G. Moreno-Jiménez (2020). LISSS : A toy corpus of literary Spanish sentences sentiment for emotions detection. arXiv :2005.08223 [cs.CL].