DEVOIR FINAL DE SÉMANTIQUE COMPUTATIONNELLE

par

Natacha Miniconi et Léna Gaubert

Table des matières

1. Introduction

Dans le cadre de notre projet, nous entreprenons une étude approfondie sur la vectorisation des fichiers textuels, une étape cruciale dans le traitement de texte. Notre objectif principal est de constituer un corpus thématique autour de Noël, en utilisant des textes provenant du Projet Gutenberg. Nous allons comparer qualitativement trois types de vecteurs largement utilisés dans le domaine : Word2Vec, CountVectorizer, et le vecteur CountVectorizer avec une réduction de dimensionalité grâce à l'analyse en composantes principales (PCA).

Mots clefs. — Plongements lexicaux, Santa Claus.

L'importance de choisir un corpus thématique spécifique réside dans la pertinence des résultats obtenus lors de la recherche de mots voisins. En nous concentrant sur la thématique de Noël, nous anticipons une meilleure adéquation entre les vecteurs et le contenu des textes. Cette approche permettra d'évaluer la capacité des vecteurs à capturer les nuances et les associations sémantiques spécifiques à la période festive, offrant ainsi des résultats plus pertinents dans l'analyse qualitative des voisins de mots.

En outre, afin d'évaluer la performance de chaque méthode de vectorisation, nous réalisons une étude comparative à l'aide de deux listes de 25 mots clés : la première liée à la thématique de Noël, la seconde constituée des 25 termes les plus fréquents dans notre corpus. Cette approche nous permettra de comparer la capacité des vecteurs à saisir la richesse sémantique des termes spécifiques à cette période particulière de l'année, et à comprendre les termes beaucoup plus fréquents et non-spécifiques. En synthèse, notre étude vise à fournir des informations approfondies sur la pertinence des différents vecteurs dans la représentation sémantique des textes liés à Noël, en mettant en avant leur capacité à proposer des voisins de mots significatifs.

L'intégralité de notre travail est disponible sur notre dépôt Github [1].

2. Corpus

Le corpus utilisé ici a été téléchargé intégralement sur Projet Guthenberg. Il comporte 17 oeuvres : romans, et recueil de nouvelles, en langue anglaise, appartenant tous au domaine public. Le détail des oeuvres utilisées est joint en annexe.

2.1. Prétraitement. — Afin de pouvoir extraire des vecteurs avec le moins de bruit possible, un prétraitement s'est imposé. Pour effectuer ce prétraitement nous avons utilisé le module Python spacy avec sa liste de stopwords et sa tokenization. Le choix de spacy s'est fait au vu de ses performances. Grâce à ce module nous retirons les espaces, les ponctuations, et les mots vides. Nous avons aussi utilisé une regex qui effectue un deuxième filtrage sur les espaces pour s'assurer de leurs disparitions. Nous obtenons ainsi un total de 229 245 tokens.

Lors du pré-traitement, nous avons pris le parti de retirer les chiffres numériques : ces derniers ne nous semblent pas pertinents dans le cadre de ce projet, notre corpus n'étant pas un corpus historique.

1. Github `https://github.com/kittog/xmas-vect-spaces`.

3. Création des vecteurs

3.1. Choix des dimensions. — Le premier dilemme lors de la création de nos vecteurs a été celui du nombre de dimension : trop de dimensions impliquerait trop de bruit et pas assez de dimensions impliquerait peu de résultats concluants. Afin de déterminer le nombre de dimension nécessaire pour notre ACP, nous calculons la variance expliquée (*explained variance*). La variance "expliquée" est une mesure statistique indiquant quelle portion de la variation de nos données peut être attribuée à chaque composante principale générée par l'analyse en composantes principales. Ci-dessous, voici un extrait de notre code nous permettant de trouver le nombre minimal de composantes représentant 95% de la variance expliquée. De façons à ce que nos modèles soient comparables, nous allons également utiliser le même nombre de dimension pour Word2Vec pour que les modèles soient comparables.

```
    if n_components is None:
        cumulative_variance_ratio = np.cumsum(pca.explained_variance_ratio_)
        n_components = np.argmax(cumulative_variance_ratio
        >= cumulative_variance_threshold) + 1

        pca = PCA(n_components=n_components)
        vectors_pca = pca.fit_transform(df_ppmi.values)

    entity_names = df_ppmi.index

    # Plot cumulative explained variance
    plt.plot(np.cumsum(pca.explained_variance_ratio_), marker='o')
    plt.xlabel('Number of Components')
    plt.ylabel('Cumulative Explained Variance')
    plt.title('Cumulative Explained Variance vs. Number of Components')
    plt.show()
```

Le graphique ci-dessus, démontre que les 15 premières composantes de l'ACP représente 95% de la variance cumulative. Nous choisissons donc de réduire les vecteurs obtenus par PPMI à 15 dimensions.

3.2. W2vec. — Pour la réalisation des vecteurs W2vec nous avons utilisé les paramètres par défaut hormis pour le nombre de dimension choisi à 15. Cela a été fait avec le module gensim.

3.3. Count Vectorize avec PPMI. — Pour la vectorisation à l'aide du compte de mots, nous avons employé CountVectorizer avec des paramètres

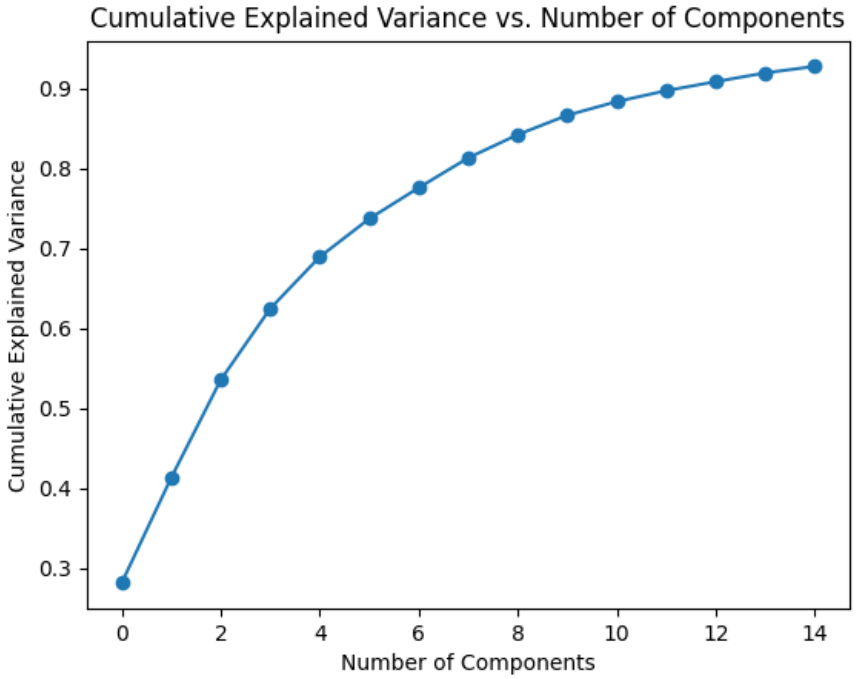

FIGURE 1. Variance cumulative expliquée et nombre de composantes

spécifiques, fixant le seuil de fréquence minimale min df à 0.30 c'est à dire que seuls les termes qui sont présents dans au moins 30% des documents seront inclus dans la représentation vectorielle. Cela nous a permis de créer une matrice de comptage des occurrences de mots, reflétant la distribution des termes dans nos documents.

Afin d'améliorer la représentation et de capturer des relations sémantiques plus fines, nous avons appliqué la transformation PPMI à la matrice résultante. La transformation PPMI (Pointwise Mutual Information) vise à atténuer l'impact des termes fréquents et à mettre en évidence les associations significatives entre les termes.

L'idée principale derrière PPMI est de mesurer la co-occurrence de mots tout en ajustant les attentes basées sur les probabilités marginales d'occurrence de chaque mot. Cette approche contribue à réduire la dimensionnalité de la représentation tout en préservant des informations sémantiques importantes.

L'étape de PPMI a été réalisée en calculant la transformation logaritmique des fréquences observées par rapport aux fréquences attendues. Cela permet de donner davantage de poids aux co-occurrences rares mais informatives, tout

en réduisant l'importance des co-occurrences fréquentes mais moins discriminantes.

En résumé, l'utilisation conjointe de CountVectorizer et de PPMI dans notre pipeline de prétraitement des données textuelles vise à obtenir une représentation plus riche et significative de nos documents, favorisant ainsi des performances potentiellement améliorées dans les tâches subséquentes d'analyse ou de modélisation.

3.4. CountVectorize avec PPMI et réduction de dimensionalité sous PCA. — Dans une étape supplémentaire visant à gérer la dimensionnalité, nous avons intégré l'analyse en composantes principales (PCA). PCA est une technique de réduction de dimensionnalité qui permet de condenser l'information contenue dans la matrice de manière à préserver les caractéristiques les plus importantes. Cette approche contribue à atténuer les effets de la dimensionnalité élevée tout en conservant les informations essentielles pour l'analyse.

4. Choix des mots

Nous établissons deux listes de mots. Notre premier choix s'est orienté vers des mots associés à la période de Noël ou ayant un lien étroit avec l'hiver. Cette approche vise à créer une base sémantique ancrée dans la saison festive, mais avec la possibilité d'observer des résultats surprenants, que ce soit des associations peu joyeuses en contraste avec la thématique de Noël, ou des termes étroitement liés aux célébrations hivernales. Nous espérons que cette sélection soigneusement pensée apportera une richesse sémantique à nos analyses, ouvrant la voie à des découvertes inattendues et à une meilleure compréhension des relations entre les mots dans notre corpus.

Notre deuxième liste comprend les 25 mots les plus fréquents à travers l'entièreté de notre corpus. Bien que nous ayons choisis des récits autour du thème de Noël, nous sommes conscientes que les mots choisis dans notre première liste ne seront probablement pas les plus fréquents : ainsi, il est fort probable que les espaces sémantiques obtenus par les modèles implémentés ne parviennent pas à bien représenter sémantiquement les mots qui nous intéressent car trop peu fréquents. Nous sélectionnons les 25 termes les plus fréquents grâce à une matrice terme-document, que nous réalisons après le pré-traitement du corpus (tokenisation, suppression des stopwords...).

5. Voisins trouvés

Nous avons générés, pour chaque mot de nos deux listes, des listes de 10, puis 20 voisins, à partir des embeddings obtenus précédemment. Nous faisons appel pour cela à la fonction NearestNeighbors du module scikit-learn : nous choisissons alors de mesurer la distance entre les vecteurs avec la similarité cosinus. À défaut de pouvoir présenter dans ce rapport les voisins obtenus pour les 50 termes considérés, nous présentons explicitement dans les tableaux ci-dessous, les voisins obtenus pour quatre termes de nos listes, à $k = 10$ et $k = 20$.

5.1. Pour k=10. — Selon nous, les résultats obtenus à partir de nos modèles diffèrent en fonction de la liste de mot considérée mais aussi en fonction du modèle. En effet, pour les termes les plus fréquents dans le corpus, Word2Vec donnent des voisins peu proches sémantiquement des mots. Pour le verbe say par exemple (voir la table 1, Word2Vec trouve les voisins *"oh"*, *"yes"*, *"dear"* ou encore *"father"* qui ne sont pas des verbes. Sur les 10 voisins, on compte 4 verbes, dont seulement 2 ayant un lien sémantique avec la parole (*"ask"*, *"tell"*). Les voisins obtenus avec la PPMI avec ou sans PCA ne sont pas particulièrement meilleurs sur le plan sémantique. Toujour pour le terme *"say"*, nous notons seulement 2 verbes de parole voisins selon la représentation PPMI : *"murmur"* et *"exclaim"* (et aucun pour PPMI avec PCA).

Les résultats pour le terme *"say"* ne sont cependant pas généralisables à tous les mots étudiés ici ! En effet, pour les termes *"look"*, *"christmas"*, *"think"* par exemple, Word2Vec parvient à trouver davantage de voisins proches sémantiquement. Ce n'est pas le cas des représentations PPMI, et PPMI avec PCA : selon nous, ce sont les listes de voisins issus du modèle Word2Vec qui sont les plus proches sur le plan sémantique.

mot	PPMI	PPMI&PCA	Word2Vec
say	dark	home	know
	flushed	voice	oh
	exclaim	dark	yes
	murmur	heap	think
	hope	care	dear
	door	knock	ask
	sight	night	tell
	patch	breath	father
	sure	drag	good
	brave	finger	nastenka

TABLE 1. Les 10 voisins les plus proches du mot "say", d'après chaque modèle

Qu'en est-il des 25 termes associés à Noël ? Bien que les oeuvres choisies soient toutes en lien avec cette période hivernale et festive, les mots que nous avons sélectionné, à l'exception de *"christmas"* sont nettement moins fréquents dans le corpus. Ainsi, les espaces sémantiques donnés par PPMI (avec et sans PCA), ne parviennent pas à établir pour un mot des voisins équivalent à des synonymes ou appartenant au même champ lexical. Pour *"reindeer"*, la représentation PPMI indique *"paint"*, ou encore *cupboard* parmi les 10 voisins les plus proches. Encore une fois, c'est Word2Vec qui propose les voisins les plus intéressants. Pour *"christmas"* notamment, nous avons comme voisins : *"merry"*, *"carol"*, *"happy"*, *"eve"*, ou encore *"day"*. Word2Vec parvient ici à obtenir des termes associés à la temporalité de Noël (réveillon, jour de Noël), et à la joie liée à ce jour. Comme le montre bien la table 2, PPMI (avec ou sans PCA) parviennent à établir des liens avec d'autres termes relatifs à Noël mais pas aussi bien que Word2Vec. Nous retrouvons cette fois-ci des mots en rapport avec la cuisine (repas de Noël), mais à l'instar de Word2Vec, nous avons également des termes associés au bonheur (*fun*, *enjoy*, *cheerfully*).

mot	PPMI	PPMI&PCA	Word2Vec
christmas	outside	potato	day
	enjoy	fun	merry
	potato	papa	eve
	north	tis	happy
	coal	cheerfully	year
	window	dish	morning
	present	north	carol
	tis	corn	child
	dish	frock	christ
	frock	storm	good

TABLE 2. Les 10 voisins les plus proches du mot "christmas", d'après chaque modèle

mot	PPMI	PPMI&PCA	Word2Vec
tree	branch	branch	present
	hut	hut	morning
	shoe	shoe	time
	gold	package	grow
	plan	gold	carol
	purple	inside	new
	little	dollar	beautiful
	package	bird	gift
	inside	purple	merry
	warm	moon	evening
	moon	city	pretty
	city	brunch	give
	slowly	altar	different
	wonder	winter	wish
	wooden	twinkle	content
	bunch	wooden	bear
	twinkle	lap	sing
	sadly	sadly	get
	winter	urge	song
	dollar	midnight	baby

TABLE 3. Les 20 voisins les plus proches du mot "tree", d'après chaque modèle

mot	PPMI	PPMI&PCA	Word2Vec
look	press	kiss	face
	prefer	sit	turn
	settle	time	see
	word	speak	head
	know	tell	tear
	expect	get	moment
	squeeze	step	take
	tear	way	stand
	like	try	stop
	disease	suppose	suddenly
	feel	money	foot
	trick	right	enter
	ought	like	stone
	person	turn	low
	sort	run	follow
	letter	ready	table
	calm	read	black
	hardly	pull	glance
	conclusion	eye	wall
	utmost	take	round

TABLE 4. Les 20 voisins les plus proches du mot "look", d'après chaque modèle

5.2. Pour k=20. — Lorsque nous étendons le nombre de voisins, leur rapport avec le mot dont ils sont voisins est d'une manière subjective moins pertinent pour la liste contenant les 25 mots les plus fréquents. Exemple : *"look"* a comme voisin pour le vecteur PPMI à la position 16 *"letter"* Nous voyons dans cet exemple que la detection des voisins est leurs pertinences est de plus en plus influencées par la thématique du corpus. La lettre fait référence à la lettre au père Noël. Nous avons donc un lien étroit dans les romans, mais d'un point de vue purement sémantique et même grammatical, *"look"* et *"letter"* sont opposés. Si nous prenons Word2Vec, nous avons *"table"* qui est un mot fortement présent et utilitaire. Cette brève comparaison nous montre tout de même que nous gardons en voisin pour *"look"* des objets que nous regardons, donc des objets dans sa thématique mais, nous n'avons pas de mots qui auraient pu être associés au champ lexical de la vision ou du regard tel que *"eye"*.

Pour la liste de Noël nous avons également ce cas de figure. Prenons *"christmas"*, dans l'espace sémantique donné par PPMI, son voisin numéro

20 est *"papa"* , nous avons aussi *"eat"*. Ces termes démontrent que les voisins de ce terme sont largement influencés par le thème propre à Noël. Si nous prenons un mot plus large tel que *"tree"* nous avons un fait intéressant : son voisin numéro 19 est *"winter"*, nous nous attendions à avoir par exemple des variétés d'arbre en voisin, ou d'autres façons de dire arbre, des synonymes. Cependant, il est à noter que si nous disons en français *"sapin de Noël"*, en anglais, il est plus commun de dire *"Christmas tree"*, ce qui explique peut-être pourquoi aucun terme désignant une espèce d'arbre n'est présent dans les voisins de *"tree"*. Toujours dans l'espace sémantique donné par PPMI (avec ou sans PCA), nous avons comme voisins *"branch"* et *wooden*, mais aussi *"bird"*. Nous parvenons donc à avoir des termes en lien avec le bois, l'arbre (dans sa composition : ici, *"branch"* est le méronyme de *"tree"*), et la nature. La présence du voisin *"winter"* indique que les espaces obtenus établissent des liens entre l'arbre et la saison hivernale (donc un élément temporel) qui constitue un élément clé des récits présents dans notre corpus.

6. Étude qualitative

Une fois les voisins obtenus pour chaque méthode de plongement lexical, nous procédons à une étude qualitative, afin de comparer les différents espaces sémantiques calculés. Pour cela, nous nous reprenons le score de variation proposé par Pierrejean et Tanguy (2018) : ce dernier est une mesure visant à comparer les listes de voisins pour un mot donné par deux modèles différents. Le score de variation est définit selon la formule suivante :

$$var^{n}_{M_1,M_2}(w) = \frac{|\text{neighb}^{n}_{M_1}(w) \cap \text{neighb}^{n}_{M_2}(w)|}{n}$$

Avec n, le nombre de voisins, M_1, M_2 les deux modèles comparés, et w le mot considéré. Plus le score est élevé (et donc proche de 1), plus les deux représentations sémantiques du mot considéré divergent, et inversement.

Tandis que Pierrejean et Tanguy (2018) comparaient 19 modèles de plongements lexicaux à Word2vec (initialisé avec ses paramètres par défaut), nous avons décidé de comparer nos trois modèles entre eux : ainsi, en plus de pouvoir évaluer qualitativement nos modèles, nous pourrons également évaluer l'impact de l'analyse en composantes principales sur la PPMI, ce qui ne nous semble pas négligeable. Nous calculons donc les scores de variation pour $k = 10$, et $k = 20$, de façon à bien comparer les différents espaces sémantiques que nous avons.
Ce qui est frappant, c'est que pour $k = 10$ comme pour $k = 20$, ces espaces

sémantiques se superposent peu ou pas du tout. Ainsi, lorsqu'on compare Word2Vec à PPMI (avec ou sans PCA), nous avons des scores de variation égaux à 1.0 ou s'en approchant énormément. Les scores de variation obtenus pour PPMI avec et sans PCA présentent cependant des résultats différents dépendant des listes de mots considérés : nous cherchons à comprendre pourquoi dans les sections suivantes.

voisin	W2V/PPMI	W2V/PPMI&PCA	PPMI/PPMI&PCA
christmas	1.0	1.0	0.5
present	1.0	1.0	1.0
light	1.0	1.0	0.6
tree	1.0	1.0	0.3
snow	1.0	1.0	0.09
bell	1.0	1.0	0.19
santa	0.9	0.9	0.3
sleigh	0.9	1.0	0.4
reindeer	1.0	1.0	0.09
angel	1.0	1.0	0.19

TABLE 5. Scores de variation pour 9 des mots de notre liste "Noël", pour $k = 10$

voisin	W2V/PPMI	W2V/PPMI&PCA	PPMI/PPMI&PCA
say	1.0	1.0	0.9
come	1.0	1.0	1.0
little	1.0	1.0	0.6
know	1.0	1.0	0.9
christmas	1.0	1.0	0.5
look	0.9	1.0	1.0
go	1.0	0.9	0.9
child	1.0	1.0	0.3
man	1.0	1.0	1.0
old	1.0	1.0	0.4

TABLE 6. Scores de variation pour 9 des mots de notre liste des termes les plus fréquents, pour $k = 10$

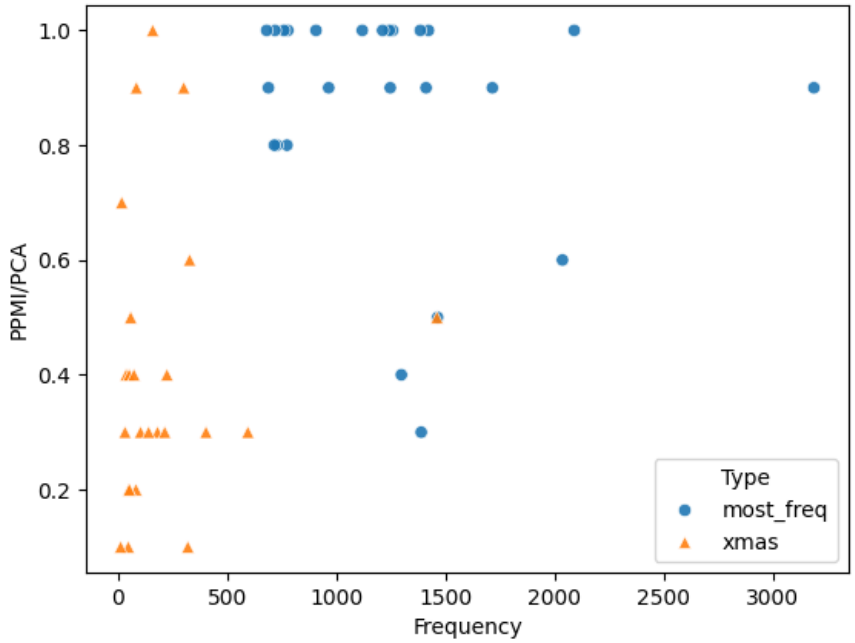

FIGURE 2. Score de variation en fonction de la fréquence d'apparition des mots dans le corpus

6.1. L'influence de la fréquence sur le score de variation. — Nous nous intéressons au lien entre la fréquence du mot dans le corpus et le score de variation $var^n(w)$. Le table 6 montre que pour les mots de notre liste associée à Noël, les espaces sémantiques donnés par PPMI (avec ou sans PCA) se superposent davantage. Pour $k = 10$, on a un score de variation moyen de 0.396 (contre 0.976 pour W2V/PPMI&PCA, et 0.980 pour W2V/PPMI). Pour $k = 20$, ce score de variation moyen diminue de 3 centièmes, mais nous observons la même tendance. Dans le contexte de cette liste de mot thématique, les espaces sémantiques données par PPMI avec ou sans PCA sont beaucoup plus proches, et donnent des représentations similaires des mots étudiés.

Ces résultats sont notamment intéressants car nous n'obtenons pas les mêmes scores de variation moyen pour PPMI/PPMI&PCA lorsque nous considérons les 25 mots les plus fréquents. En effet, pour $k = 10$ nous obtenons un score de variation moyen de 0.863. Ce score, beaucoup plus élevé, tend à dire que la réduction de dimension opérée par PCA a peut être une plus grande influence face aux mots très fréquents dans un texte. En étendant le nombre

de voisins à $k = 20$, nous avons un score moyen légèrement plus bas, de 0.838, mais la tendance reste la même.
Nous pouvons voir sur la figure 2 que les mots moins fréquents (dans notre cas, les mots de la liste de Noël) présentent des scores de variation beaucoup plus bas que les mots les plus fréquents du corpus. La figure 2, réalisée pour $k = 10$, montre bien une séparation distincte entre les deux listes de mots. L'un des seuls mots avec une grande fréquence d'apparition dans notre corpus, obtenant un score de variation plus bas pour PPMI/PPMI&PCA est *"christmas"*. Pour une grande majorité des mots de notre liste de Noël, les espaces sémantiques données par PPMI/PPMI&PCA se superposent davantage. Cette figure montre donc bien l'influence de la fréquence du mot sur le score de variation, lorsque nous comparons ces deux modèles. En effet, il est important de noter que nous ne retrouvons pas ce résultat lorsque nous comparons Word2Vec à PPMI ou Word2Vec à PPMI&PCA, que cela soit pour les mots les plus fréquents du corpus, où les mots associés à Noël, à $k = 10$ comme à $k = 20$.

7. Conclusions

Les résultats indiquent que la variation n'affecte pas tous les mots de manière égale, et certaines caractéristiques telles que la fréquence et la sémantique des mots peuvent jouer un rôle dans la variation. Nous observons également qu'avoir effectué ces tests sur un corpus ayant une thématique particulière a influencé grandement les voisins. La réalisation des deux listes : ayant une thématique ou non, nous a également permis de nous rendre compte que des mots ayant un sens particulier dans un corpus précis présenteront peu de variations pour les modèles PPMI avec et sans PCA, à contrario d'un mot plus commun dénoué de thématique. Nous avons donc observé qu'en plus de la fréquence, la sémantique a été également déterminante au niveau du degrés de variation entre les espaces donnés par les modèles. Au cours de ce projet, nous nous sommes également intérrogées sur la définition d'un mot voisin à un autre. Suite à ces observations, nous pouvons nous dire qu'un mot voisin peut être considéré autrement que par des synonymes. La thématique d'un corpus crée d'autres types de voisins qui eux se rapprochent du mot dont ils sont voisins d'une manière plus subjective et qui sont sujets à interprétation. Les scores de variation entre nos modèles ont montré que l'interprétation d'un voisin n'est pas si évidente et n'est pas propre à la notion de synonyme. La machine est fortement influencée par la fréquence des mots dans le corpus mais aussi par la thématique du corpus.

Références

[1] Pierrejean, Tanguy (2018) *Towards qualitative word embedding evaluation: measuring neighbors variation*

Appendice A
Corpus

— *A Christmas Carol*, Charles Dickens
— *A budget of Christmas Tales*, Charles Dickens et autres
— *A Kidnapped Santa Claus*, L. Frank Baum
— *A little book for Christmas*, Cyrus Brady
— *Christmas at Thompson Hall*, Anthony Trollope
— *Christmas Carols*, sélectionnés et édités par L. Edna Walter
— *Christams Stories and Legends*, rassemblés par Phebe A. Curtiss
— *Old Christmas from the Sketch Book of Washington Irving*, Washington Irving
— *Some Christmas Stories*, Charles Dickens
— *The Birds' Christmas Carol*, Kate Douglas Smith Wiggin
— *White Nights and Other Stories*, Fyodor Dostoyevsky
— *The Squire's young folk A Christmas story*, Eleanora H. Stooke
— *The Children's Book of Christmas Stories*, édité par Asa Don Dickison et Ada M. Skinner
— *The cricket on the hearth*, Charles Dickens
— *The Life and Adventures of Santa Claus*, L. Frank Baum
— *The Night before Christmas*, auteurs divers
— *Twilight Stories*, Catharine Shaw

Appendice B
Graphiques

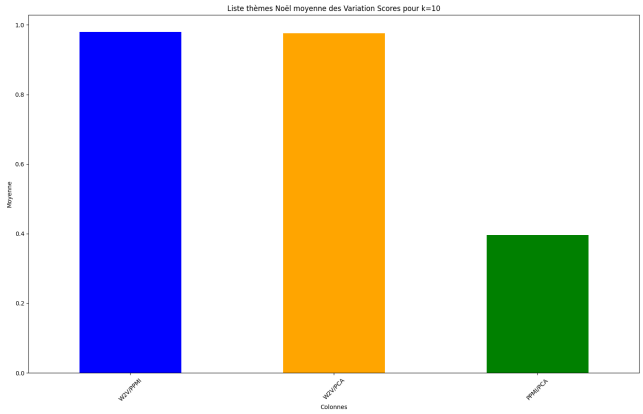

FIGURE 3. Scores de variation moyen pour la liste "Noël", pour $k = 10$ $\overline{var}_{W2V,PPMI} = 0.980$, $\overline{var}_{W2V/PCA} = 0.976$, $\overline{var}_{PPMI,PCA} = 0.396$

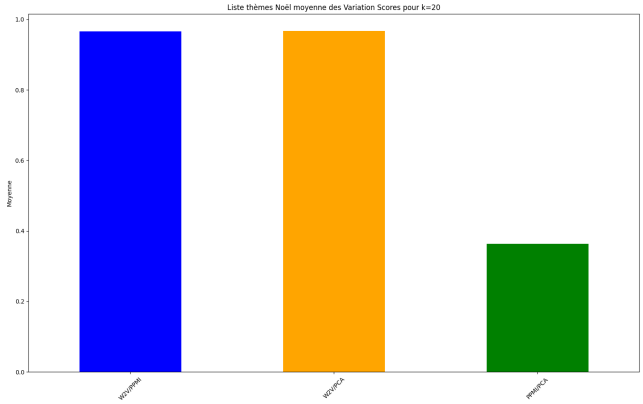

FIGURE 4. Scores de variation moyen pour la liste "Noël", pour $k = 20$ $\overline{var}_{W2V,PPMI} = 0.965$, $\overline{var}_{W2V/PCA} = 0.967$, $\overline{var}_{PPMI,PCA} = 0.364$

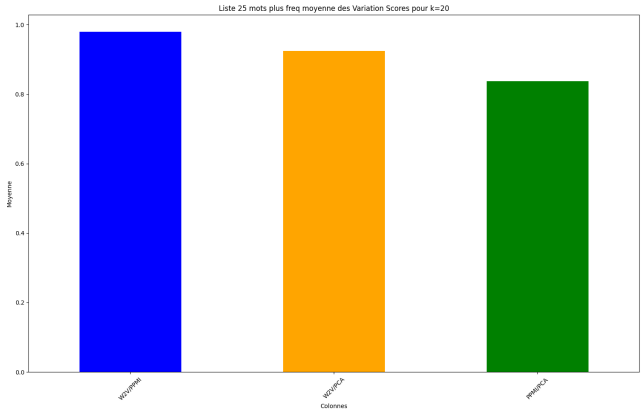

FIGURE 5. Scores de variation moyen pour la liste des 25 termes les plus fréquents,
pour $k = 20$ $\overline{var}_{W2V,PPMI} = 0.980$, $\overline{var}_{W2V/PCA} = 0.924$, $\overline{var}_{PPMI,PCA} = 0.838$

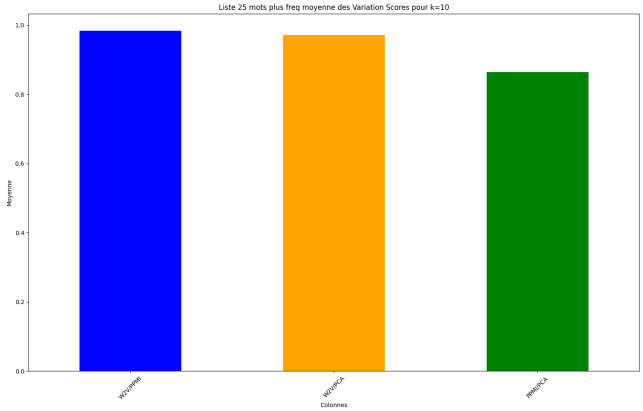

FIGURE 6. Scores de variation moyen pour la liste des 25 termes les plus fréquents,
pour $k = 10$ $\overline{var}_{W2V,PPMI} = 0.984$, $\overline{var}_{W2V/PCA} = 0.972$, $\overline{var}_{PPMI,PCA} = 0.864$

NATACHA MINICONI ET LÉNA GAUBERT
Url : `https://github.com/kittog/xmas-vect-spaces`