RÉPUBLIQUE FRANÇAISE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE

Université Gustave Eiffel

Rapport de fin de formation

Text Mining : Analyse des sentiments des tweets sur le Covid-19

Présenté par Bernice AGOSSOUVO

Supervisé par Roland TROSIC

Le 3 septembre 2021

Licence Professionnelle Métiers du décisionnel et de la Statistique

Année universitaire 2020-2021

REMERCIEMENTS

- Mes remerciements vont à l'endroit de tout le corps administratif notamment M. Hervé CLEMENT, Responsable de la formation pour son dévouement et son sens de l'organisation

- Je ne saurais également oublier M. Roland TROSIC, Responsable adjoint de la formation. Son soutient est d'une importance capitale dès le début de l'année jusqu'au choix du sujet relatif au présent rapport.

Table des matières

INTRODUCTION

L’avènement de l’intelligence artificielle et du big data ont fait émerger de nombreuses pratiques de traitement de données existantes déjà mais peu connues. C’est le cas notamment du text mining. La majorité des informations créées de nos jours sont des données non structurées, c'est-à-dire qu'elles ne rentrent pas dans une structure ou un cadre bien défini. La plupart de ces données se présentent sous forme de texte : messages sur les médias sociaux, courriels, critiques en ligne, rapports d'activité. Elles recèlent d'énormes quantités d'informations, jusqu'aux opinions et aux émotions de leurs auteurs. Pourvoir les traiter et en tirer des informations précieuses pour le développement des activités d’une structure est le champ d’action du Text Mining.

Le Text mining est le processus d'examen de grandes collections de documents pour découvrir de nouvelles informations ou aider à répondre à des questions de recherche spécifiques. Il permet d'identifier des faits, des relations et des affirmations qui, autrement, resteraient enfouis dans la masse des big data textuelles. Une fois extraites, ces informations sont converties en une forme structurée qui peut être analysée plus en profondeur.

Durant 3 mois, je me suis donnée pour mission d’explorer le champ d’action du Text Mining, de comprendre la théorie sous-jacente de cette dernière et son utilisation pour répondre aux diverses problématiques d’analyse de données textuelles.

Qu’est-ce que le Text Mining du point de vue technique et pratique ? Comment faire son usage et en tirer profit au sein d’une entreprise ? Exemple d’analyse de Text Mining : algorithme de sentiment sur des tweets publiés sur Twitter. Voilà en somme les différents éléments que nous aborderons dans les lignes à suivre. Une trame bien définie et détaillée dans notre table des matières nous en éclaire davantage.

I. PRESENTATION THEORIQUE DU TEXT MINING

I.1 A LA DECOUVERTE DU TEXT MINING

I.1.1 Définition du Text Mining

Le Text Mining fait partie intégrante des sciences regroupées dans la data science, et donc dans l'IA. C'est un ensemble de méthodes, techniques d'analyse linguistique et outils utilisés pour manipuler et traiter de la donnée textuelle. Il s'agit surtout de données non structurées, non référencées dans une base et qui ne sont donc pas interprétables par des machines.

On appelle également cette technologie l'analyse textuelle. Toutefois certaines personnes établissent une distinction entre les deux termes. L'analyse textuelle fait référence à l'application utilisant des techniques de Text Mining pour trier les ensembles de données. Cette technologie qui remonte à plusieurs années.

En effet l'utilisation de l'informatique pour appliquer des techniques d'analyse textuelle n'est pas récente. En 1957 existait déjà l'automatisation de résumé de texte pour un article ! Avant même la naissance du terme Business Intelligence ! (« The Automatic Creation of Literature Abstracts » par Hans Peter Luhn). (Anthony DEMOGUE, 2021) [1]

I.1.2 Les applications du Text Mining

❖ *Gestion des risques*

L'une des principales causes d'échec dans le secteur des affaires est l'absence ou l'insuffisance d'analyse des risques. L'adoption et l'intégration d'un logiciel de gestion des risques reposant sur des technologies d'exploration de texte telles que SAS Text Miner peuvent aider les entreprises à se tenir au courant de toutes les tendances actuelles du marché des affaires et à renforcer leurs capacités à atténuer les risques. Comme les outils et les technologies de Text Mining peuvent rassembler des informations pertinentes à partir de milliers de sources de données textuelles et créer des liens entre les informations extraites, ils permettent aux entreprises d'accéder à la bonne information au bon moment, améliorant ainsi l'ensemble du processus de gestion des risques.

[1] https://blogdigital.beijaflore.com/text-mining/

❖ *Améliorer l'expérience client*

Les techniques du Text Mining, en particulier le traitement automatique des langues, prennent de plus en plus d'importance dans le domaine de la relation client. Les entreprises investissent dans les logiciels d'analyse de texte pour améliorer l'expérience globale de leurs clients en accédant aux données textuelles provenant de sources variées telles que les enquêtes, les commentaires des clients, les appels des clients, etc.

Ainsi le Text Mining permet d'analyser les retours et avis des clients sur une marque et ses produits. Dans le but de comprendre leurs opinions, mais aussi leurs attentes et la qualité de leur expérience auprès de l'entreprise. Les avis sur les produits, les commentaires sur les réseaux sociaux, les réponses aux sondages peuvent être passés au crible.

❖ *Social Media Analysis*

Il existe de nombreux outils de Text Mining conçus exclusivement pour analyser les performances des plateformes de réseaux sociaux. Ils permettent de suivre et d'interpréter les textes générés en ligne à partir des actualités, des blogs, des e-mails, etc. En outre, les outils du Text Mining peuvent analyser efficacement le nombre de messages, de likes et de followers de votre marque sur les réseaux sociaux, vous permettant ainsi de comprendre la réaction des personnes qui interagissent avec votre marque et votre contenu en ligne. De cette manière, il est possible de s'appuyer sur les données pour prendre les bonnes décisions et améliorer les points faibles.

❖ *Business Intelligence*

Les organisations et les entreprises ont commencé à exploiter les techniques d'exploration de texte dans le cadre de leur veille économique. En plus de fournir des informations approfondies sur le comportement et les tendances des clients, les techniques d'extraction de texte aident également les entreprises à analyser les forces et les faiblesses de leurs concurrents, leur donnant ainsi un avantage concurrentiel sur le marché. Les outils de Text Mining tels que Cogito Intelligence Platform et IBM text analytics fournissent des informations sur la performance des stratégies de marketing, les dernières tendances des clients et du marché (Rai, 2019)[2].

IL existe d'autres utilisations courantes comme :

- La sélection des candidats à l'emploi en fonction du libellé de leur curriculum vitae
- Le blocage des courriers indésirables

[2] https://www.upgrad.com/blog/what-is-text-mining-techniques-and-applications/

- La classification du contenu du site Web
- Le signalement des réclamations d'assurance pouvant être frauduleuses,

I.1.3 La logique (et la technologie) derrière la fouille de textes

De manière générale, l'objectif principal du Text mining est de transformer le texte en données de manière à pouvoir l'analyser. Pour y parvenir, il est nécessaire d'appliquer différents algorithmes d'intelligence artificielle et des techniques statistiques aux documents textuels. Le Text mining fait appel ainsi à un large éventail de tâches dans lequel il est possible de distinguer quatre différentes étapes :

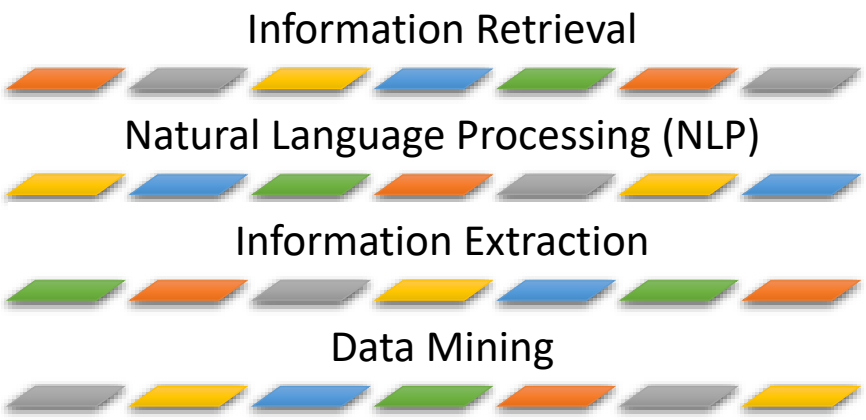


Figure 1: Techniques du Text mining -- Sources : Auteur

I.1.3.1 Information Retrieval

La recherche d'information est utilisée pour identifier des documents pertinents à partir d'une large collection de documents textuels. Elle vise à identifier le sous-ensemble de documents qui correspondent à la requête de l'utilisateur. Les outils utilisés dans les bibliothèques pour rechercher des livres et le moteur de recherche google sont deux exemples de systèmes de recherche d'information.

I.1.3.2 Natural Language Processing (NLP)

Les outils de traitement du langage naturel constituent la base du Text Mining. Leur rôle est d'apporter au texte un premier niveau de compréhension à partir d'une analyse syntaxique et grammaticale. Ces outils s'enchaînent un à un pour augmenter au fur et à mesure la compréhension des phrases. Ainsi, une détection des mots est suivie par une détection des phrases. Un outil va chercher les lemmes des mots, c'est-à-dire leur forme de base sans pluriel ni conjugaison, tandis qu'un autre va chercher leur nature grammaticale. Enfin, le dernier outil, le parseur, aura pour rôle de comprendre la structure globale de la phrase en analysant sujet, verbe, complément etc., en s'adaptant à la tournure des phrases. Le résultat de ces outils de traitement du

langage naturel servira ensuite d'entrée pour des fonctionnalités plus poussées comme l'analyse sémantique et la découverte d'informations.

I.1.3.3 Information Extraction

Afin d'être exploité comme tout autre type de données, le document non structuré doit être transformé en données sous une forme structurée. Cette étape s'appelle l'extraction d'informations et ce sont les données générées par les systèmes NLP. La tâche la plus courante effectuée au cours de cette étape est l'identification de termes spécifiques, qui peuvent consister en un ou plusieurs mots, comme dans le cas des documents de recherche scientifique contenant de nombreux termes complexes comportant plusieurs mots.

L'extraction d'informations nous permet également de lier des noms et des entités (par exemple : les personnes et les organisations auxquelles elles sont affiliées) et des faits plus complexes tels que des relations entre des événements ou des noms.

I.1.3.4 Data Mining

Lorsque la base de données structurée est constituée, les données sont prêtes à être analyser. Pour ce faire, étant donné que les données se présentent désormais sous une forme exploitable, il est possible de recourir à des procédures et techniques statistiques standard appliquées aux données textuelles désormais structurées.

I.1.4 Les Avantages du Text Mining

La recherche d'opinion via Text Mining peut aider les entreprises à détecter les problèmes liés aux produits et aux affaires. Cela permet de les résoudre avant qu'ils ne deviennent de gros problèmes et affectent les ventes. Faire du Text Mining dans les avis clients et les communications peut également identifier les nouvelles fonctionnalités souhaitées pour renforcer les offres de produits. L'expérience client globale en est améliorée ce qui, espérons-le, entraînera une augmentation des revenus et des bénéfices.

Cette science peut également aider à prédire le taux de désabonnement des clients. Les entreprises peuvent alors prendre des mesures pour éviter les résiliations potentielles de contrats vers des concurrents commerciaux. La détection de fraude, la gestion des risques, la publicité en ligne et la gestion de contenu Web sont d'autres fonctions qui peuvent bénéficier de l'utilisation d'outils de Text Mining.

Dans le domaine de la santé, cela peut aider à diagnostiquer les pathologies des patients en fonction des symptômes signalé.

I.2 LA THEORIE DU TEXT MINING

I.2.1 Le Text Mining, comment ça marche ?

Le Text Mining est de nature similaire au Data Mining. La différence est qu'il met l'accent sur le texte plutôt que sur des formes de données plus structurées. Cependant, l'une des premières étapes du processus de Text Mining consiste à organiser et structurer les données afin de pouvoir les soumettre à une analyse à la fois qualitative et quantitative. Cela implique généralement l'utilisation des algorithmes NLP (Natural Language Processing), qui appliquent les principes de la linguistique informatique pour analyser et interpréter les ensembles de données.

Petit rappel sur les principales notions

Document = individu statistique

« Base » d'apprentissage = collection de documents = Corpus

Enjeu : Traduire la collection de document en un tableau de donnée propice à l'analyse en minimisant la perte de l'information

I.2.2 Les Etapes du pré-traitement des documents.

Les différentes étapes du pré-traitement sont au nombre de trois de façon globale. Il s'agit de commencer par diviser le document ou le texte en mots, cela s'appelle la Tokenisation, viens ensuite les étapes de lemmatisation et ou la racinisation (Stemming en anglais) et la suppression des Stop Word.

I.2.2.1 Tokenisation

La tokenisation des documents consiste à identifier les unités de textes élémentaires qui peuvent être des mots, mais aussi des lettres, des syllabes, des phrases, ou des séquences de ces éléments. Chaque document devient alors une liste ordonnée (ou non) de termes élémentaires : les tokens. Nous passons d'un plan de données qui liste des documents (et les associe éventuellement à des auteurs), à un plan qui associe un document à une série d'attributs qui sont ses éléments unitaires (lettres, mots, syllabes, phrases). Si les mots sont des unités de sens évidentes, les paires, les triplets

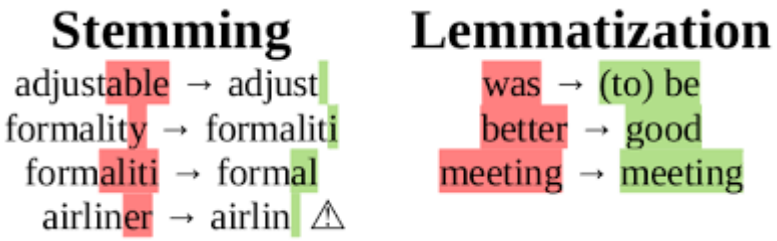
de mots le sont aussi. Par exemple, dans l'analyse du corpus du Grand Débat National, l'expression « mille-feuille administratif » apparaît fréquemment, il est à lui seul une unité signifiante. Les n-grammes sont ainsi des suites de 1,2, … n lettres, syllabes ou mots dont on va mesurer la fréquence d'apparition dans le corpus.

I.2.2.2 Stemming (racinisation) et Lemmatisation

Pour des raisons grammaticales, les documents vont utiliser différentes formes d'un même mot comme « écrire, écrire et écrit ». En outre, il existe des familles de mots apparentés par dérivation ayant des significations similaires. L'objectif de la racinisation et de la lemmatisation est de réduire les formes flexionnelles et parfois les formes dérivées d'un mot à une forme de base commune.

- La racinisation fait généralement référence à un processus qui coupe les extrémités des mots dans l'espoir d'atteindre l'objectif correctement la plupart du temps et inclut souvent la suppression des affixes de dérivation.

- La lemmatisation fait référence quant à elle à l'utilisation d'un vocabulaire et d'une analyse morphologique des mots, visant normalement à supprimer uniquement les terminaisons flexionnelles et à ramener la forme de base et la forme du dictionnaire d'un mot.



I.2.2.3 Les Stop Word

Un Stop Word est un mot vide qui est communément utilisé dans une langue, non porteur de sens dans un document (ex. préposition, pronoms, etc.). Formellement, sa fréquence d'apparition est la même dans tous les documents. De fait, les mots vides ne permettent pas de discriminer les documents (de distinguer les documents les uns des autres), ils sont inutilisables en Text mining.

Après ce processus de pré- traitement, on passe à la construction du tableau de données appelés Matrice Documents-termes

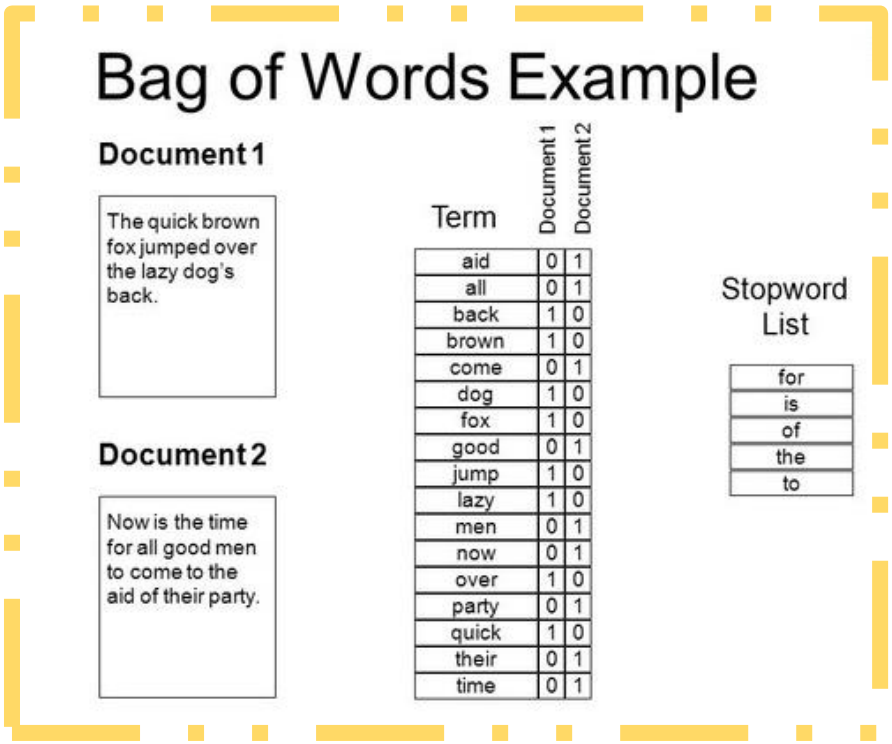
I.2.3 Matrices Documents – Termes

I.2.3.1 Bag of words / Bag of N-grams

Bag of words est une représentation simplifiée utilisée dans le traitement du langage naturel qui transforme un texte (phrase ou document) arbitraire en vecteurs de longueur fixe en comptant le nombre d'occurrences de chaque mot. Ce processus est souvent appelé vectorisation.

Nous créons alors une matrice document – terme. Cette matrice présente en colonne l'ensemble des mots présents dans notre vocabulaire. Pour chaque mot, nous renseignons dans notre matrice la présence ou non

L'illustration de ce procédé se résume sur ce graphique



Cette pondération par 0 ou 1 qui indique si le mot est présent ou pas dans le texte va vite atteindre ses limites. Notamment pour les textes plus longs, elle ne permettra pas de faire la différence entre un texte qui ne mentionne qu'une seule fois le mot contre un texte qui le mentionne plusieurs fois.

Ainsi il existe plusieurs métriques pour compléter cette matrice document terme. Une des plus utilisées est le TF-IDF qui permet de connaitre l'importance relative de chaque mot dans les textes.

I.2.3.2 La pondération tf-idf

La valeur tf-idf, abréviation de term frequency-inverse document frequency, est une statistique numérique destinée à refléter l'importance d'un mot pour un document dans une collection ou un corpus. Elle est souvent utilisée comme facteur de pondération dans les recherches d'informations. La valeur de tf-idf augmente proportionnellement au nombre de fois qu'un mot apparaît dans le document et est compensée par le nombre de documents du corpus qui contiennent le mot, ce qui permet d'ajuster le fait que certains mots apparaissent plus fréquemment en général.

Le tf-idf d'un terme dans un document est élevé quand ce dernier apparait beaucoup dans le document mais se fait rare par ailleurs

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

N = total number of documents

Bien que ces représentations des mots soient simples et faciles à mettre en œuvre, elle pose certains problèmes. En effet, il n'est pas possible de déduire de relation entre deux mots à partir de leur représentation. Alors que cette approche de contextualisation des mots est tout aussi importante pour tirer le maximum d'information de l'analyse textuelle.

I.2.3.3 Words embedding

Le word embedding repose sur la théorie linguistique fondée par Zelling Harris et connue sous le nom de Distributional Semantics. Cette théorie considère qu'un mot est caractérisé par son contexte, c'est à dire par les mots qui l'entourent. Ainsi, des mots qui partagent des contextes similaires partagent également des significations similaires. Les algorithmes de word embedding sont le plus souvent employés pour décrire des mots à travers des vecteurs numériques (Rai, 2019)[3].

L'objectif est de faire en sorte que les mots ayant un contexte similaire occupent des positions spatiales proches. Mathématiquement, le cosinus de l'angle entre de tels vecteurs devrait être proche de 1, c'est-à-dire un angle proche de 0.

[3] https://dataanalyticspost.com/Lexique/word-embedding/

Il existe plusieurs approches de Word embedding (voir annexe). La plus répandue est le Wor2vec.

Word2Vec peut être obtenue à l'aide de deux variantes (impliquant toutes deux des réseaux de neurones) : Skip Gram et Common Bag Of Words (**CBOW**).

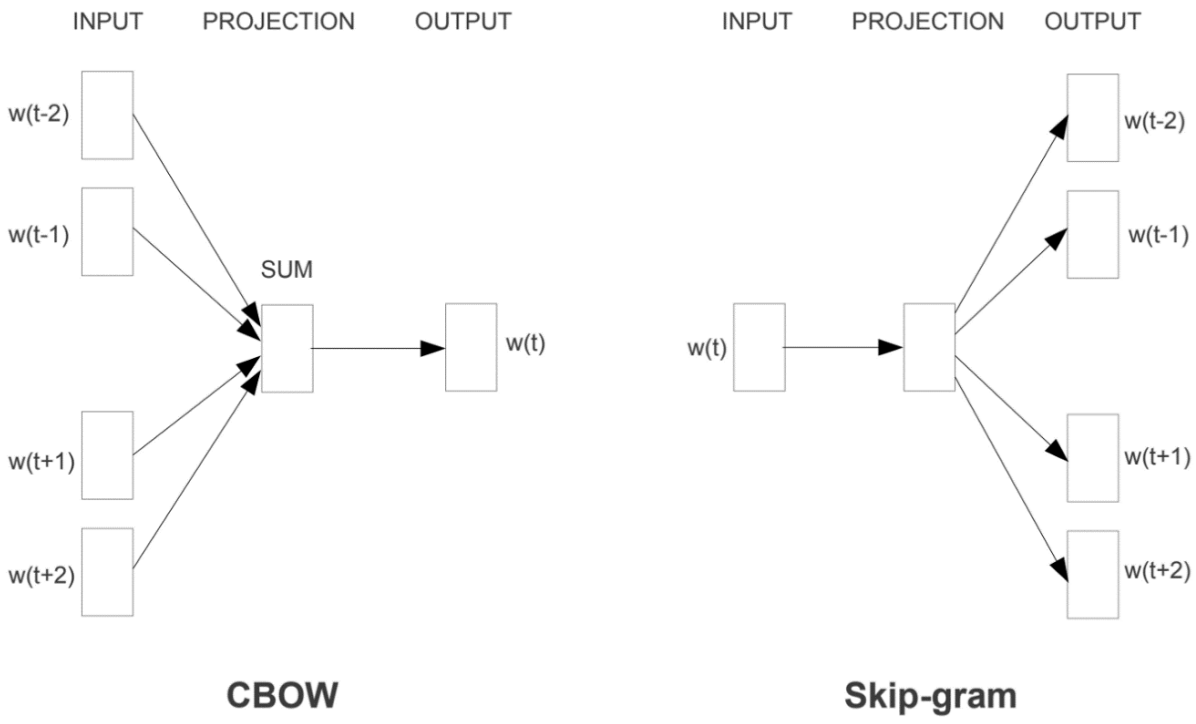
CBOW : Cette méthode prend le contexte de chaque mot comme entrée et essaie de prédire le mot correspondant au contexte.

Skip Gram est en revanche tout le contraire car il utilise le mot cible (dont nous voulons générer la représentation) pour prédire le contexte. Le graphe ci-dessus schématise le process

Prédiction de la probabilité qu'un mot apparaisse étant donnée les mots qui l'entourent

Prédiction des contextes à partir d'un mot cible

(htt1)



Pour récapituler, on peut transformer un texte en ses features soit :

- En utilisant une représentation de comptage creuse - fréquence d'apparition du mot dans un document, ou vecteur tf-idf d'un document, etc.

- En utilisant une représentation type word2vec dense - dans laquelle le mot possède une représentation dans un espace qui le positionne en fonction des mots adjacents

Plusieurs autres approches de recherche d'information pertinente sont à citer.

- NER (Named Entity Recognition) : reconnaître des personnes, endroits, entreprises, etc.
- POS Tagging (Part-of-Speech Tagging) : représente les méthodes qui récupèrent la nature grammaticale des mots d'une phrase - nom, verbe, adjectif, etc. Ce sont des propriétés qui peuvent servir de caractéristiques utiles lors de la création de certains modèles

I.2.4 Classification

Après le traitement, la normalisation et la transformation des données, il est commun d'appliquer les algorithmes de Machines Learning pour catégoriser les documents dans le Text mining. Ici également les notions de classification supervisée et non supervisée sont utilisées.

I.2.4.1 Classification supervisée

A l'instar de la data mining, on peut utiliser tous les différents algorithmes de machine learning supervisés usuels pour répondre aux questions de prédiction des documents en se référant aux labels déjà prédéfinis.

I.2.4.2 Classification non supervisée

En revanche, pour la classification supervisée, il n'est pas commun d'utiliser les algorithmes de K-means, de classification hiérarchique ascendante. D'autre modèles sont d'usage et traitent particulièrement ces cas de figure sous le nom de topic Modeling. Il existe plusieurs méthodes de topic modeling comme le Latent Semantic Analysis (LAS) et le Latent Dirichlet Allocation (LDA)

II. ANALYSE DES SENTIMENTS DES TWEETS DE COVID19

Twitter est utilisé de nos jours par des centaines de millions de personnes dans le monde entier. Il est l'un des réseaux sociaux générant un flux dense d'informations en temps réel. Les données utilisées dans ce cas d'usage proviennent de ce dernier. La base de données est composée de tweets sur le covid (l'heure des tweets, le texte tweeté, le sentiment que relate chaque tweet etc…). En somme elle contient 41157 lignes et 06 variables pour la base d'apprentissage et 3798 lignes et 06 Variables pour la base de test. Le tableau ci-dessous nous donne un aperçu de la base d'apprentissage.

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
0	1	44953	NYC	02-03-2020	TRENDING: New Yorkers encounter empty supermar...	Extremely Negative
1	2	44954	Seattle, WA	02-03-2020	When I couldn't find hand sanitizer at Fred Me...	Positive
2	3	44955	NaN	02-03-2020	Find out how you can protect yourself and love...	Extremely Positive
3	4	44956	Chicagoland	02-03-2020	#Panic buying hits #NewYork City as anxious sh...	Negative
4	5	44957	Melbourne, Victoria	03-03-2020	#toiletpaper #dunnypaper #coronavirus #coronav...	Neutral

Figure 2 : Aperçu des données de la base d'apprentissage

II.1 EXPLORATION DES DONNEES

II.1.1 Analyse descriptive

Il convient avant l'analyse proprement dite des données de faire une petite exploration afin de comprendre le jeu de donnée.

Sentiments relatés par les tweets

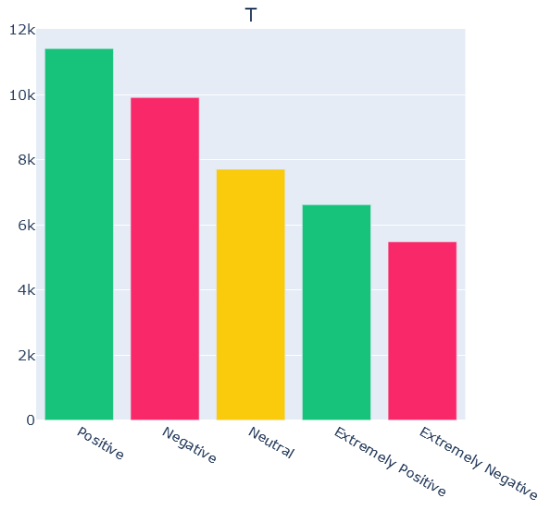


Figure 3 : Bar plot de la variable sentiments

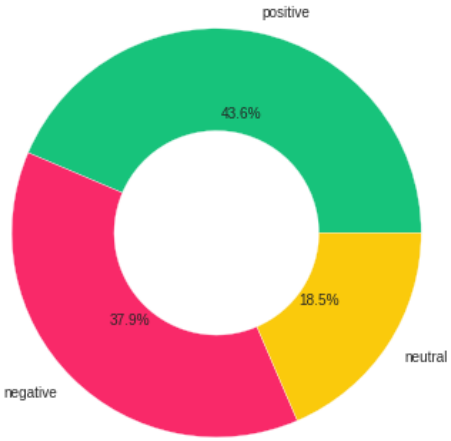


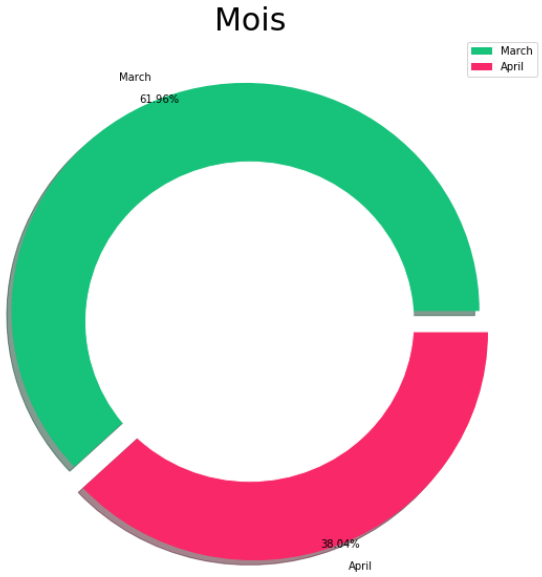
Figure 4 : Camembert de sentiments après regroupement

Le graphe ci-dessus montre la distribution des sentiments selon les tweets. Les labels sont découpés en 5 catégories de sentiments (sentiment positif, extrêmement positif,

négatif, extrêmement négatif et sentiment neutre). On remarque ainsi que les sentiments positifs reviennent le plus souvent que ceux négatifs. Plus concrètement 43,6% des tweets de la base d'apprentissage décrivent un sentiment positif alors que 37 ,9% décrivent un sentiment négatif. Ces valeurs ne s'éloignent pas tellement l'une de l'autre. Quant aux sentiments neutres ils regroupent un pourcentage de 18.5%.

Ce sont les tweets étiquetés de la base d'apprentissage qui nous permettront de prédire le sentiment que décrit un tweet lambda concernant le covid19 sur la base des données test.

Mois des tweets



Les tweets s'étalent sur deux mois de l'année 2020, les mois de mars et avril.

Il y a eu plus de tweets sur le covid19 au cours du mois de mars qu'au cours du mois d'avril

Figure 5 : Analyse univariée de TweetAt (Mois des tweets)

Aussi le mois ne change rien quant aux sentiments relatés par les tweets. Dans le mois de mars comme le mois d'avril, le sentiment positif est le plus enregistré et le sentiment neutre le moins enregistré.

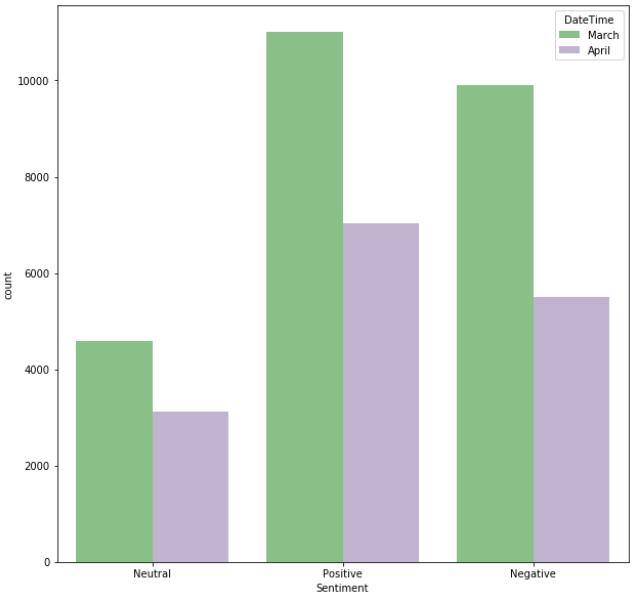


Figure 6 : Analyse Bivariée entre Mois et sentiments

II.1.2 Analyse lexicale

Pour la suite, les variables qui nous intéressent sont : Originaltweet et Sentiment.

Originaltweet rassemble les écrits des internautes et Sentiment décrit le sentiment relaté par le tweet (sentiment positif, négatif ou neutre).

Pour chaque catégorie de sentiment, il est fait une analyse lexicale pour connaitre le nombre de caractère, la taille moyenne des mots contenus dans les tweets

- *Le nombre de charactères dans chaque catégorie de tweets*

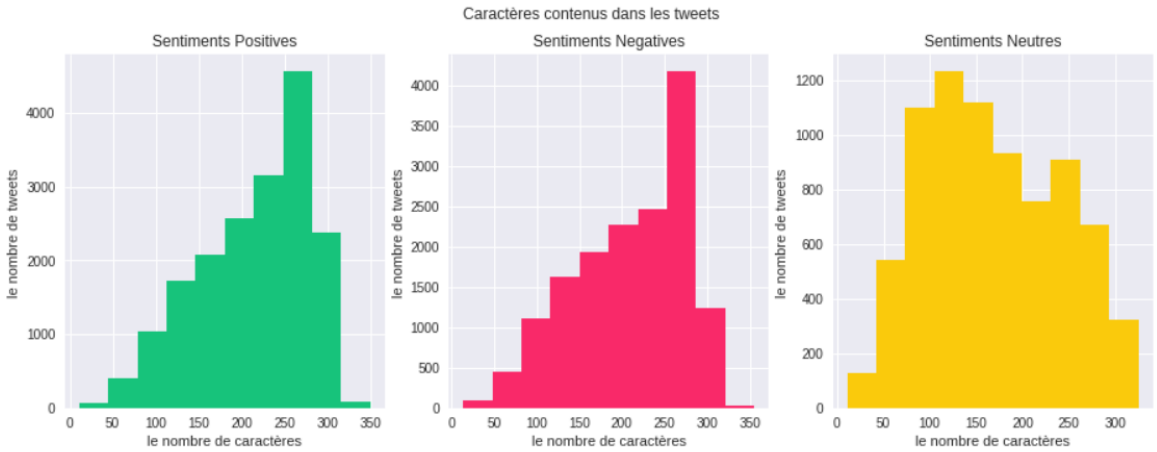


Figure 7 : Caractères contenus dans les tweets

On voit sur ce graphe que la plupart des tweets ont entre 50 et 300 caractères. Que le sentiment soit positif ou négatif, il n'y a pas tellement de différence dans la répartition des caractères par tweets. Plus de 4000 tweets dans les deux cas ont 250 à 300 caractères. En revanche pour les tweets relatant un sentiment neutre, on enregistre plus de tweets contenant 100 à 150 caractères. La distribution est aussi très différente par rapport aux autres sentiments.

- *Moyenne de la taille des mots tweetés*

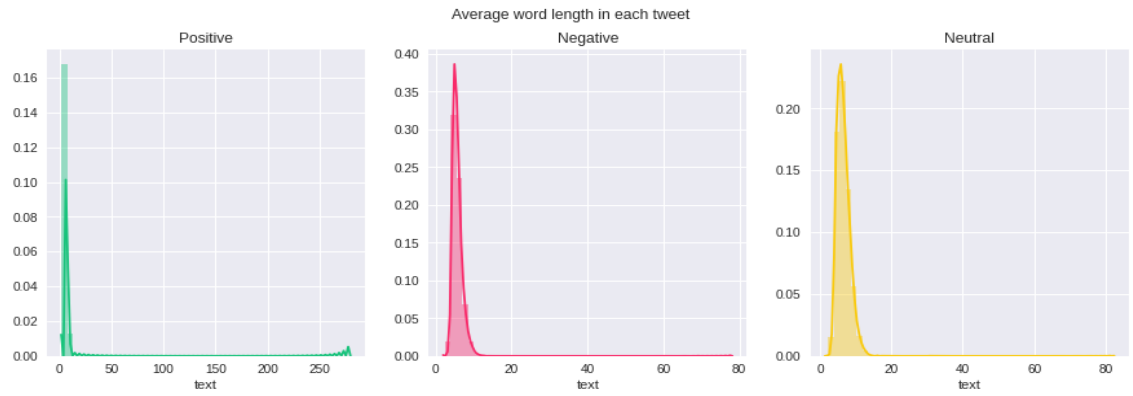


Figure 8 : Moyenne de la taille des mots contenus dans les tweets

Ici la taille moyenne des mots dans les tweets est comprise entre 0 et 15 en général. Néanmoins les tweets des sentiments positifs vont jusqu'à enregistrer comme moyenne une taille de 250. Ce qui pourrait laisser croire que plus la taille d'un mot dépasse 200 ou plus précisément plus un tweet comporte des hyper lien plus ce tweet est susceptible de refléter un sentiment positif.

- *Hashtags*

Les mots les plus référencés tournent autour de Coronavirus et sont libellés sous les noms Covid19, COVID19, CoronaCrisis et Social distancing

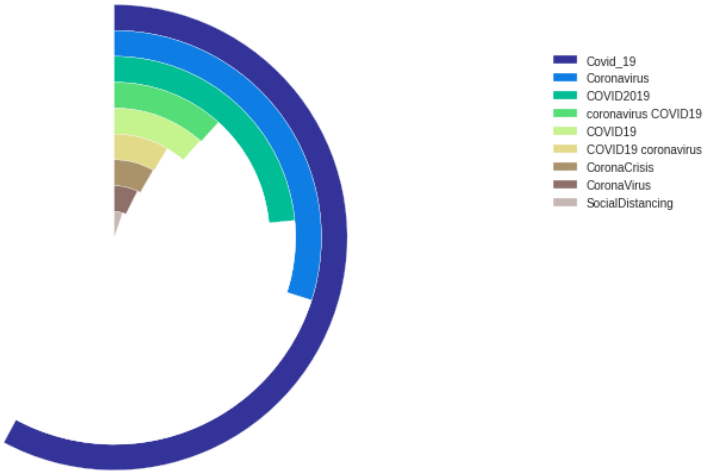


Figure 9 : Hashtags les plus référencés

- *Stop words*

*L'*ensemble de la base contient des stops words. On peut ainsi remarquer que need, now, haven reviennent le plus souvent.

Les tweets rapportent pourraient on dire des manques, des besoins de tel ou telle autre chose. Ce qui nous amène à rechercher en dépit de l'analyse des sentiments, les principaux sujets dont feraient mention les textes écrits par les internautes

Figure 10 : Word Cloud des Stop Word

II.2 NETTOYAGE ET NORMALISATION DES DONNEES

Cette étape nous permet de transformer les données brutes en données exploitables. Elle commence par des opérations élémentaires de normalisation du texte qui sont les suivantes :

● Repérer (et éliminer) les liens URL, les codes html, les émojis et les mentions de personnes,

● supprimer les ponctuations, les chiffres et les symboles,

● Éliminer aussi les mots sans signification que l'anglais dénomme par « stopwords » avec des dictionnaires,

● Repérer et traiter les émoticônes. Le langage numérique a la particularité de réintroduire des éléments iconiques dans une écriture culturellement alphabétique

Ainsi le graphique ci-dessous donne un aperçu du nettoyage effectué sur les données.

Données brutes

```
@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/iFz9FAn2Pa and https://t.co/xX6ghGFzCC and https://t.co/I2NlzdxNo8
Me, ready to go at supermarket during the #COVID19 outbreak.

Not because I'm paranoid, but because my food stock is litteraly empty. The #coronavirus is a serious thing, but plea
se, don't panic. It causes shortage...

#CoronavirusFrance #restezchezvous #StayAtHome #confinement https://t.co/usmuaLq72n
```

Après nettoyage

```
    and  and
Me, ready to go at supermarket during the  outbreak.

Not because I'm paranoid, but because my food stock is litteraly empty. The  is a serious thing, but please, don't pa
nic. It causes shortage...
```

On voit sur la première image, des tweets comportant les pseudos des internautes avec le symbole arobase (@), des urls vers d'autres sites web et des hashtags.

Après nettoyage on a des textes ne comportant que des mots, des expressions prêts à être comptés pour rendre compte de leur importance. Mais avant cela il est indispensable de supprimer les stop words.

Suppression des Stop Words

Avec l'analyse syntaxique précédemment réalisée on a pu visualiser le wordcloud des stop words. Ici nous les supprimons ou nous les mettons de côté pour rendre la construction de nos matrices document termes pertinentes

De ce fait, dans l'exemple ci-dessous, les mots tels que « to », « at », « the », « because », « my », « is », « but » ont été supprimés.

Avec Stop Words	Sans Stop Words
Me, ready to go at supermarket during the outbreak.	Me, ready go supermarket outbreak.
Not because I'm paranoid, but because my food stock is litteraly empty.	Not I'm paranoid, food stock litteraly empty.
The is a serious thing, but please, don't panic. It causes shortage	The serious thing, please, panic. It causes shortage

Tokenisation et Lemmatisation.

Apres suppression des stop words, nous procédons à la tokenisation et à la lemmatisation de nos textes. Et nous pouvons ainsi voir le Wordcloud de nos textes nettoyés suivant leurs labels.

Figure 11 : WordCloud des mots après nettoyage par catégorie de tweets

Dans les tweets de sentiment négative, le mot Covid revient le plus souvent que dans les autres tweets. Il est suivi du mot Super – Market.

Au niveau des sentiments positifs on voit hand sanitizer (hygiène des mains) dont certains font mention. On pourrait ainsi dire qu'un tweet qui contient cette expression est un tweet rapportant un sentiment positif.

II.3 STRUCTURATION DES DONNEES

Le traitement préalablement réalisé nous a permis de construire des données structurées à base de nos données non structurées. Nous avons pu dégager de la base brute trois différents tableaux.

Le 1er tableau comporte le nombre de mots, de mentions, de hashtags, de urls contenu dans chaque tweet.

	count_words	count_mentions	count_hashtags	count_capital_words	count_excl_quest_marks	count_urls	sentiment
0	17	3	0	0	0	3	1
1	38	0	0	1	0	0	2
2	18	0	0	1	0	1	2
3	46	0	7	8	0	1	2
4	45	0	6	0	0	1	0
5	41	1	0	1	0	1	2
6	33	0	1	0	0	1	2
7	17	0	3	0	0	1	1
8	50	0	0	1	2	1	2
9	44	0	2	1	1	0	0

Le 2ème et le 3ème tableau sont des matrices documents- termes réalisées sur la base des données nettoyées. le premier contient le bag of Word simple avec la présence ou non d'un terme dans un tweet et le second est le bag of word plus complexe réalisé avec td-idf.

II.4 CONSTRUCTION DES MODELES

Comme décrit plus haut, Les algorithmes de machine learning sont tout autant utilisés pour le Text mining et plus spécialement pour l'analyse des sentiments. La procédure est la même, et les indicateurs de performance de modèle sont également les mêmes. Nous tenons à notifier néanmoins que dans ce cas de figure, ce sont les algorithmes de classification supervisée qui seront de mise étant donné que nous cherchons à classer des textes suivant des labels prédéfinis.

De ce fait, la liste des algorithmes est bien évidemment exhaustive, néanmoins nous faisons le choix de deux dans ce projet : Naïves Bayes et Stochastic Gradient Descent (SGD Classifier) et interprétons les résultats suivant la matrice de confusion et la courbe roc.

Aussi pour rendre la classification possible nous encodons la variable Sentiment

De ce fait :

- Sentiment Positif (extrêmement Positif et Positif) -> 2
- Sentiment Neutre -> 1
- Sentiment Négatif (Extrêmement Négatif et Négatif) -> 0

Petit rappel

- Précision = VP /(VP +FN) ou VN/(VN +FP)
- Recall = VN/(VN + FN) ou VP/(VP + FP)
- F1 score est la moyenne harmonique de la précision et du recall

II.4.1 Résultats du 1er Tableau

Le premier résultat que nous présentons est celui de la classification réalisée sur la base du tableau résumant les mots, urls contenus dans chaque texte.

Le but ici est de voir si le nombre de mots, d'urls ou de hashtags présents dans un tweet pourrait influer sur le sentiment rapporté par ce dernier.

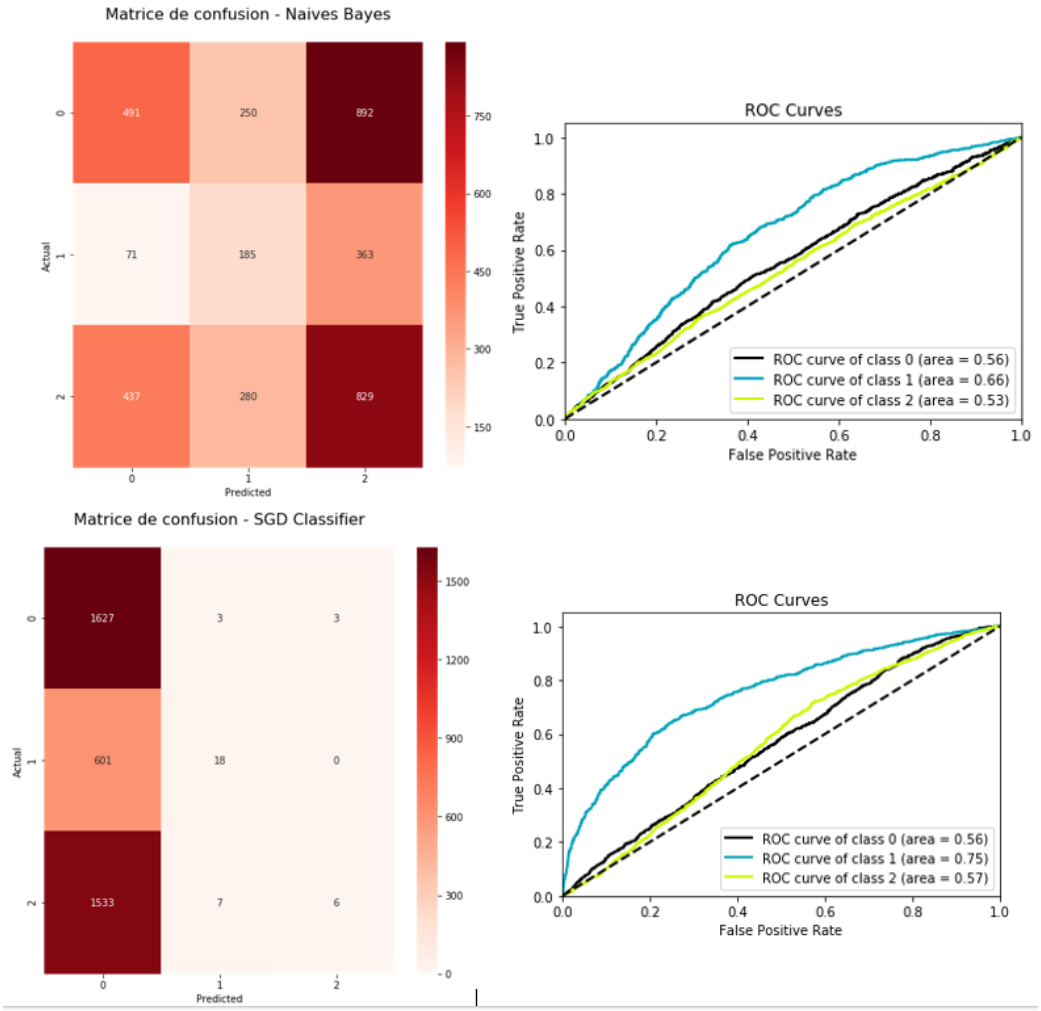


Figure 12 : Matrice de confusion et Courbe Roc du 1er tableau

❖ Interprétation

Nos résultats nous indiquent que ce modèle n'est pas performant car la prédiction dans ce cas de figure n'est pas différente de celle qu'on aurait en choisissant au hasard les labels. Cela s'explique par la courbe roc qui est pratiquement confondue à la bissectrice

II.4.2 Résultats du 2ème Tableau

Ici nous construisons le modèle sur la base du bag of words (nombre de fréquence d'un mot dans un texte) avec la fonction CountVectorizer().

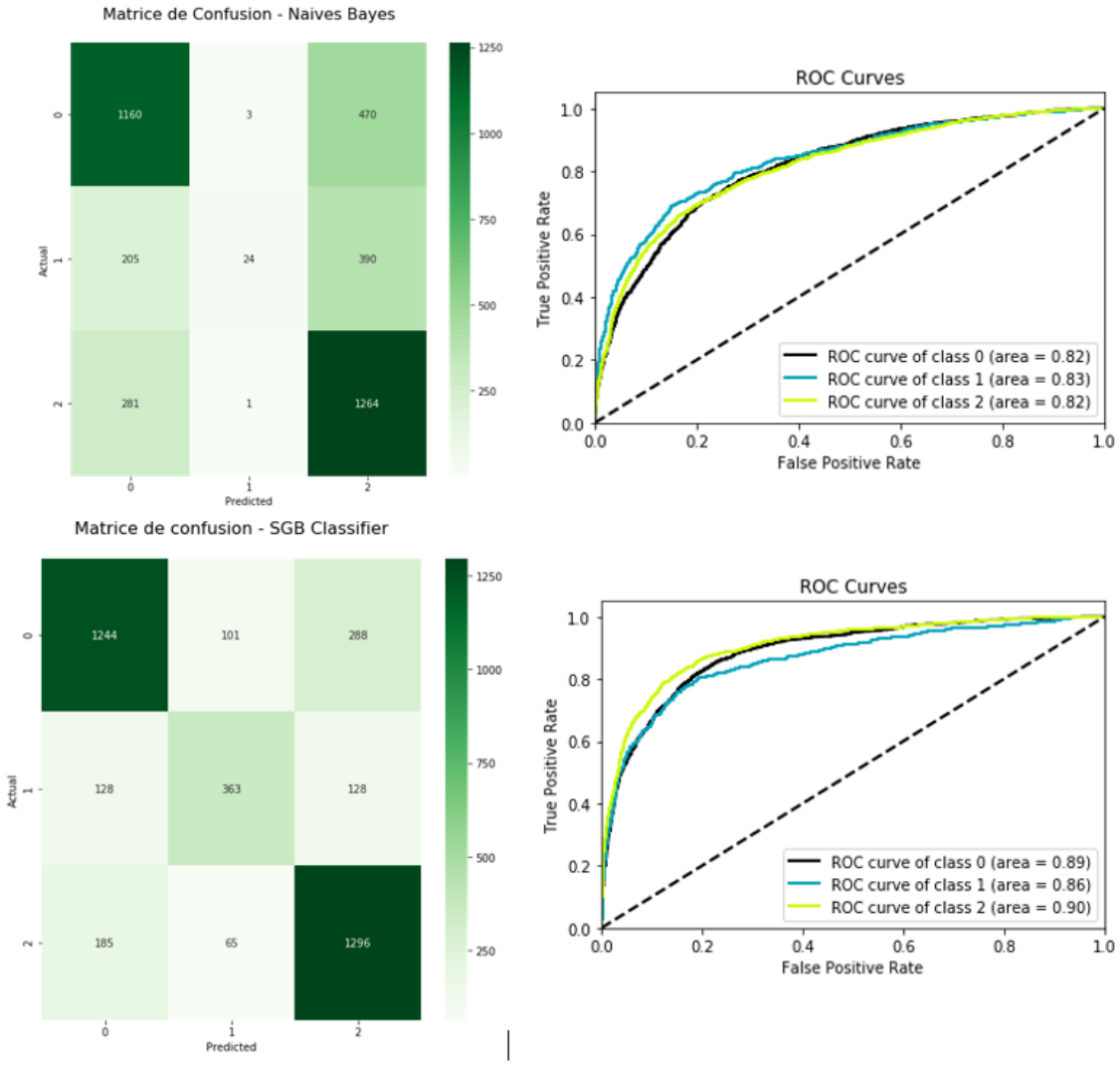


Figure 13: Matrice de confusion et courbe roc du 2ème Tableau

❖ Interprétation

Des deux modèles présentés ci-dessus, nous voyons que le second modèle prédit mieux les sentiments des tweets. Notre attention se portera sur ce dernier quant à l'interprétation.

La matrice de confusion de ce modèle nous montre que 2903 /3798 tweets sont bien classés au détriment de 895. L'accuracy est ainsi de 0,764. Ce qui signifie que l'indice de gini (2*0,764-1) est supérieur à 0,5. On note aussi toujours sur la matrice de confusion que le modèle a du mal à prédire les sentiments neutres. En effet pour cette classe la précision est de 0,59 alors qu'elle est respectivement de 0,76 et 0,84 pour la classe 0(sentiment négatif) et 2 (sentiment positif)

Aussi la courbe roc s'éloigne véritablement de la bissectrice et l'aire sous la courbe est de 0,89. Ce qui tend vers 1.

II.4.3 Résultats du 3ème tableau

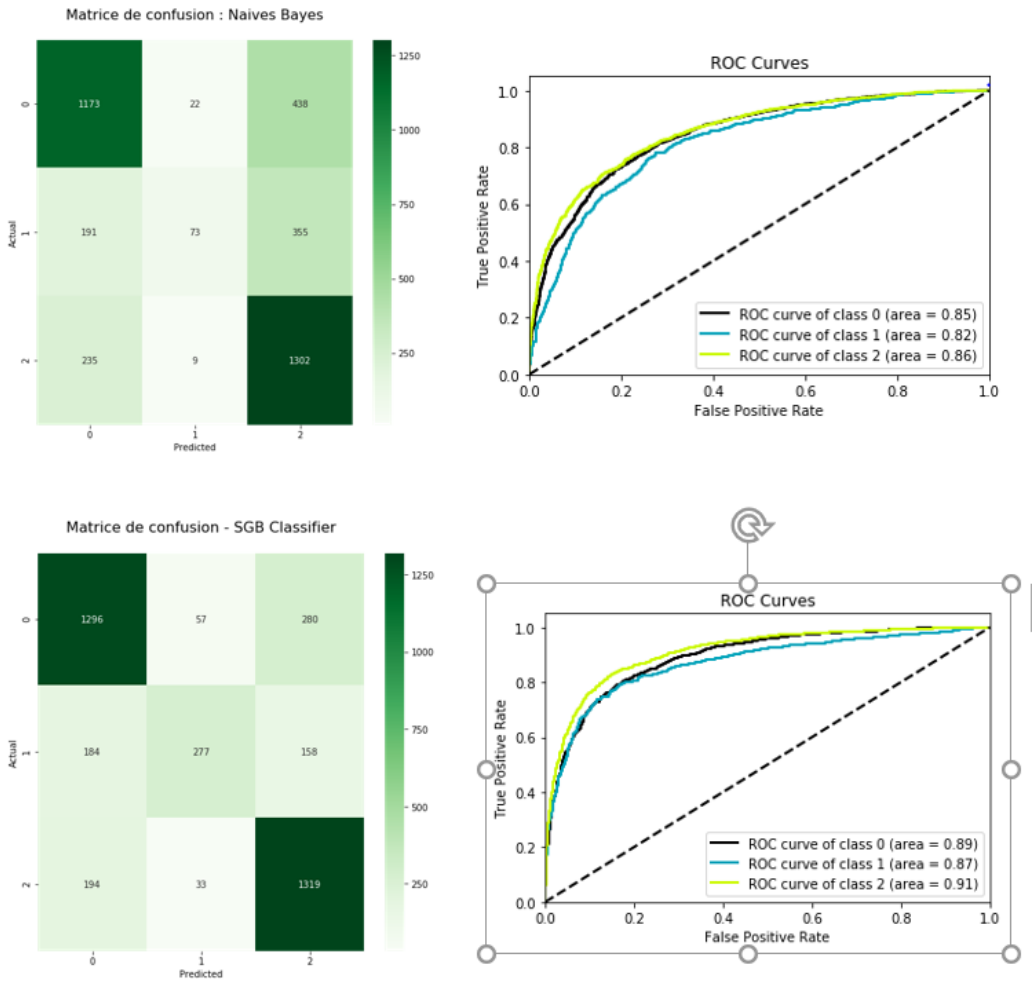


Figure 14 : Matrice de confusion et courbe roc du 3ème tableau

❖ Interprétation

De toute évidence le modèle SGD Classifier est plus robuste que celui de Naïves Bayes. Car on fait la même remarque que sur le tableau précédent. Le modèle de Naïves Bayes a du mal à prédire la classe neutre. En plus de cela ces résultats sont moins performants que ceux du second modèle. L'aire sous la courbe ROC du second modèle est considérable. La précision globale est 0,761. Ce qui signifie que l'indice de Gini est > à 0,5.

II.5 COMPARAISON DES MODELES

Le tableau suivant permet de comparer la performance de chaque modèle afin de faciliter la prise de décision sur le choix de classifieur le plus apte ou approprié pour prédire les sentiments d'autres tweets.

		PRECISION	F1 SCORE	INDICE DE GINI
TABLEAU N°1	Naives bayes	0,4	0,39	-0,2
	SGD Classifier	0,43	0,27	-0,14
TABLEAU N°2	Naives bayes	0,64	0,6	0,28
	SGD Classifier	0,76	0,75	0,52
TABLEAU N°3	Naives bayes	0,67	0,64	0,34
	SGD Classifier	0,76	0,73	0,52

Tout d'abord, on peut voir que les indicateurs obtenus avec le tableau 1 sont assez révélateur de la non-pertinence de ces données. En effet, un tweet ne peut pas être classer en tenant uniquement compte du nombre de caractères, du nombre de mots, d'hyper lien ou de hashtags qu'il contient. Construit un modèle avec ce tableau revient à faire une mauvaise prédiction.

Cependant les modèles obtenus avec les deux derniers tableaux présentent d'assez bon résultats comparés au 1er tableau. Il permet de bien classer plus de la moitié des tweets mais les meilleurs résultats sont ceux obtenus avec le SGD Classifier

Le F1-score qui est une moyenne harmonique pondérée de la précision et de la sensibilité est respectivement de 0,75 et 0,73 pour le 2ème et le 3ème tableau.

Il ressort que l'application du même modèle sur les deux différents tableaux montre que le tableau obtenu avec le bag of word (fréquence) notamment le tableau 2 est plus optimal que celui obtenu avec la méthode tf-idf.

III. RETOUR D'EXPERIENCE

III.1 RETOUR CRITIQUE SUR L'ANNEE

Ma licence à l'université Gustave Eiffel a été d'une importance capitale. Grâce à ce choix et cette opportunité de formation, j'ai embrassé le monde de la data et j'ai concrètement entamé ma démarche d'acquisition de connaissances fondamentales dans l'univers du big data.

Les différentes matières m'ont permis de développer mes connaissances et compétences dans la data. Ces dernières sont essentiellement caractérisées par : la collecte, le traitement, l'analyse de données structurées et non structurées. Concrètement, j'ai pu approfondir mes connaissances de la gestion de bases de données, de l'utilisation des langages de programmation (Python, SQL), des outils statistiques (R) et les outils de data visualisation. Les termes comme Analyse en Composante Principale (ACP), Classification supervisée, non supervisée, Data mining, Text mining, Réseaux de neurones, Informatique Décisionnelle et Web Analytics n'ont plus de secrets pour moi.

Aussi, j'ai aimé l'organisation, la structuration et le contenu des cours qui m'ont été dispensés. Les cours sont très adaptés et m'ont permis d'élargir ma vision de la data.

Le zoom prévu les jours ou l'on n'a pas de cours est une très bonne initiative en mon sens. Cela m'a permis de rester focus sur l'objectif. L'organisation des projets tutorés, l'accompagnement des responsables a permis de réaliser des projets concrets, qui enrichit le cv.

Cependant, j'ai relevé quelques points que je voudrais remonter en termes de suggestions : il y a eu quelques cours où nous avions eu juste à travailler sur nos ordinateurs chacun. Je cite en guise d'exemple le cours sur les Réseaux de neurones. Il aurait été plus approprié de poser des bases théoriques permettant aux étudiants de mieux appréhender cette notion. Puisque la pratique perd de son sens sans une bonne dose de théorie. Aussi la manière dont ce cours précisément a été dispensé n'a pas permis de maximiser les heures.

III.2 RETOUR GLOBAL SUR L'ANNEE

L'une des choses que j'ai aimées et qui m'ont fait grandir c'est l'ensemble des entretiens réalisés dans le cadre de ma recherche de stage ou d'alternance au cours de

l’année. Quand bien même je n’ai pas eu l’opportunité d’effectuer un stage ou l’alternance, j’ai pu m’ouvrir davantage et développer les attitudes et aptitudes convenables à la recherche d’opportunité. A titre d’exemple, j’ai appris à gérer le stress Cela constitue une réelle valeur ajoutée pour moi.

La réalisation des différents projets entre collègues de classe m’a fait découvrir mes facultés de communication et permis d’expérimenter le travail en équipe, de même que le management de l’humain quelques soient les situations sachant qu’il n’est pas à perdre de vue l’objectif fixé qui est à atteindre.

Par ailleurs, j’aurais souhaité qu’il y ait un Master professionnel qui soit convenable le à la licence professionnelle de ma formation actuelle. Cela permettra aux étudiants qui veulent continuer leur Master de le faire. Je proposerais également que l’université Gustave Eiffel analyse la possibilité d’établir des partenariats avec les entreprises afin que les apprenants qui ont fini leur formation aient la possibilité d’effectuer un stage de longue durée au terme de leurs études. Ceci va surtout à l’endroit de ceux qui n’ont pas eu la chance d’avoir ni le stage ni l’alternance durant leur cursus.

CONCLUSION

En définitive, le Text mining, à l'instar du data mining permet de repérer les patterns contenus dans des données textuelles. C'est une bonne approche qui permet de tirer le meilleur des données et qui aident une majorité d'entreprises à se perfectionner tout en développant leur activité.

Ce rapport qui couronne ma formation cette année m'a permis d'explorer un autre domaine de l'analyse des données. Désormais, qu'il me soit permis d'affirmer que le Text mining ne m'est plus étranger. J'ai fait miennes les techniques du NLP (la tokenisation, la lemmatisation, le parsing) et j'ai également appris à structurer des données textuelles pour l'analyse avec les méthodes de bag of word, de word embedding. Je ne manquerai pas de mentionner les méthodes de classification supervisée dont j'ai pris connaissance notamment le topic modeling.

J'ai la ferme conviction que j'ai pu développer une forte appétence pour le traitement des documents et je compte poursuivre dans cette même lancée en apprenant à traiter d'autres données non structurées notamment, les images, les vidéos. Quelles seraient les spécificités liées au traitement de ces dernières ?

TABLE DES ILLUSTRATIONS

Bibliographie

(s.d.). Récupéré sur https://datascientest.com/nlp-word-embedding-word2vec

Anthony DEMOGUE, L. D. (2021). Text Mining : Le machine learning sur les données textuelles.

https://dataanalyticspost.com/Lexique/word-embedding/. (s.d.).

post, D. A. (s.d.). *WORD EMBEDDING.*

Rai, A. (2019). What is Text Mining: Techniques and Applications. *UpGrade blog*.

ANNEXES

Annexe 01 : Base de données

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is emp...	Positive
4	3803	48755	NaN	16-03-2020	Me, ready to go at supermarket during the #COV...	Extremely Negative
5	3804	48756	ÃT: 36.319708,-82.363649	16-03-2020	As news of the regionÂs first confirmed COVID...	Positive
6	3805	48757	35.926541,-78.753267	16-03-2020	Cashier at grocery store was sharing his insig...	Positive
7	3806	48758	Austria	16-03-2020	Was at the supermarket today. Didn't buy toile...	Neutral
8	3807	48759	Atlanta, GA USA	16-03-2020	Due to COVID-19 our retail store and classroom...	Positive
9	3808	48760	BHAVNAGAR,GUJRAT	16-03-2020	For corona prevention,we should stop to buy th...	Negative
10	3809	48761	Makati, Manila	16-03-2020	All month there hasn't been crowding in the su...	Neutral
11	3810	48762	Pitt Meadows, BC, Canada	16-03-2020	Due to the Covid-19 situation, we have increas...	Extremely Positive
12	3811	48763	Horningsea	16-03-2020	#horningsea is a caring community. LetÂs ALL ...	Extremely Positive
13	3812	48764	Chicago, IL	16-03-2020	Me: I don't need to stock up on food, I'll jus...	Positive
14	3813	48765	NaN	16-03-2020	ADARA Releases COVID-19 Resource Center for Tr...	Positive
15	3814	48766	Houston, Texas	16-03-2020	Lines at the grocery store have been unpredict...	Positive
16	3815	48767	Saudi Arabia	16-03-2020	????? ????? ????? ????? ??\r\r\n?????? ????? ?...	Neutral
17	3816	48768	Ontario, Canada	16-03-2020	@eyeonthearctic 16MAR20 Russia consumer survei...	Neutral
18	3817	48769	North America	16-03-2020	Amazon Glitch Stymies Whole Foods, Fresh Groce...	Extremely Positive
19	3818	48770	Denver, CO	16-03-2020	For those who aren't struggling, please consid...	Positive
20	3819	48771	southampton soxx xxx	16-03-2020	with 100 nations inficted with covid 19 th...	Extremely Negative
21	3820	48772	Global	16-03-2020	https://t.co/AVKrR9syff\r\r\n\r\r\nThe COVID-1...	Neutral
22	3821	48773	NaN	16-03-2020	We have AMAZING CHEAP DEALS! FOR THE #COVID201...	Extremely Positive
23	3822	48774	NaN	16-03-2020	We have AMAZING CHEAP DEALS! FOR THE #COVID201...	Extremely Positive
24	3823	48775	Downstage centre	16-03-2020	@10DowningStreet @grantshapps what is being do...	Negative
25	3824	48776	London	16-03-2020	UK #consumer poll indicates the majority expec...	Extremely Positive
26	3825	48777	Ketchum, Idaho	16-03-2020	In preparation for higher demand and a potenti...	Negative
27	3826	48778	Everywhere You Are!	16-03-2020	This morning I tested positive for Covid 19. I...	Extremely Negative
28	3827	48779	New York, NY	16-03-2020	Do you see malicious price increases in NYC? T...	Negative
29	3828	48780	Someplace, USA	16-03-2020	@7SealsOfTheEnd Soon with dwindling supplies u...	Extremely Negative
30	3829	48781	NaN	16-03-2020	There Is of in the Country The more empty she...	Negative
31	3830	48782	NaN	16-03-2020	'Hole' Foods...\r\r\n\r\r\n...images from the ...	Extremely Positive
32	3831	48783	Markham, Ontario	16-03-2020	Retail store closures could explode because of...	Neutral
33	3832	48784	Virginia, USA	16-03-2020	Coronavirus fun fact: if you cough at the groc...	Extremely Positive
34	3833	48785	London, England	16-03-2020	We're sorry to say that our @FinFabUK event is...	Negative
35	3834	48786	Sverige	16-03-2020	Went to the supermarket yesterday and the toil...	Neutral
36	3835	48787	Where The Wild Things Are	16-03-2020	Yes, buy only what you need.\r\r\n\r\r\nBut wh...	Positive
37	3836	48788	Canada	16-03-2020	Worried about the impact of the current COVID-...	Positive
38	3837	48789	NaN	16-03-2020	my wife works retail&a customer came in ye...	Negative
39	3838	48790	United States	16-03-2020	Now I can go to the supermarket like this with...	Positive
40	3839	48791	Fort Worth, Texas	16-03-2020	We're here to provide a safe shopping experien...	Extremely Positive
41	3840	48792	NaN	16-03-2020	Curious, do we think retail shoppers will do ...	Positive
42	3841	48793	Houston	16-03-2020	CHECK VIDEO ?? https://t.co/1ksn9Brl02 ??No fo...	Extremely Negative
43	3842	48794	Vancouver, British Columbia	16-03-2020	Breaking Story: Online clothes shopping rises ...	Neutral
44	3843	48795	NaN	16-03-2020	This is the line outside @Target in as custo...	Neutral

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
41137	44936	89888	LES, NYC	14-04-2020	Distilleries have switched portions of their p...	Extremely Positive
41138	44937	89889	Los Angeles, CA	14-04-2020	HMU FOR PRICES!! Got great deals going right n...	Extremely Positive
41139	44938	89890	NaN	14-04-2020	Hello everyone \r\r\nPlease share this in your...	Positive
41140	44939	89891	Pakistan	14-04-2020	Good News! \r\r\nWe'll Soon Announce Our High ...	Positive
41141	44940	89892	India	14-04-2020	#Coronavirus ?? ????? ??? ????? ?? ??? ???????...	Neutral
41142	44941	89893	Juba south sudan	14-04-2020	@MajangChien @MTNSSD @MTNSSD is worst than COV...	Extremely Positive
41143	44942	89894	In burning hell.	14-04-2020	https://t.co/8s4vKvcO1r #5gtowers?? #EcuadorUn...	Neutral
41144	44943	89895	NaN	14-04-2020	@_Sunrise_SV @Gamzap @NPR What does not having...	Neutral
41145	44944	89896	Manhattan, NY	14-04-2020	How exactly are we going to re-open New York C...	Positive
41146	44945	89897	Gurgaon, India	14-04-2020	#Gold prices rose to a more than 7-year high t...	Positive
41147	44946	89898	Brooklyn, NY	14-04-2020	YÂall really shitting that much more at home?...	Negative
41148	44947	89899	NaN	14-04-2020	UV light Sterilizer Sanitizer for your mask an...	Extremely Positive
41149	44948	89900	Toronto, Ontario	14-04-2020	Still shocked by the number of #Toronto superm...	Negative
41150	44949	89901	OHIO	14-04-2020	I never that weÂd be in a situation & wor...	Positive
41151	44950	89902	NaN	14-04-2020	@MrSilverScott you are definitely my man. I fe...	Extremely Positive
41152	44951	89903	Wellington City, New Zealand	14-04-2020	Airline pilots offering to stock supermarket s...	Neutral
41153	44952	89904	NaN	14-04-2020	Response to complaint not provided citing COVI...	Extremely Negative
41154	44953	89905	NaN	14-04-2020	You know itÂs getting tough when @KameronWild...	Positive
41155	44954	89906	NaN	14-04-2020	Is it wrong that the smell of hand sanitizer i...	Neutral
41156	44955	89907	i love you so much || he/him	14-04-2020	@TartiiCat Well new/used Rift S are going for ...	Negative

Récapitulatif du traitement et de l'analyse des donnés sur Python

```
import pandas as pd
import numpy as np
import re
import nltk
from sklearn.pipeline import Pipeline
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split, cross_val_score, KFold


from io import StringIO
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_selection import chi2
from IPython.display import display
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import LinearSVC
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix
from sklearn import metrics

from sklearn.decomposition import PCA, TruncatedSVD
from sklearn.metrics import classification_report,confusion_matrix
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import matplotlib.patches as mpatches
```

```
Train = pd.read_csv('Corona_NLP_train.csv',encoding ='latin1',error_bad_lines=False)
Test = pd.read_csv('Corona_NLP_test.csv',encoding ='latin1',error_bad_lines=False)
```

```
Train['text'] = Train.OriginalTweet
Train["text"] = Train["text"].astype(str)

Test['text'] = Test.OriginalTweet
Test["text"] = Test["text"].astype(str)

# Data has 5 classes, let's convert them to 3

def classes_def(x):
    if x ==  "Extremely Positive":
        return "2"
    elif x == "Extremely Negative":
        return "0"
    elif x == "Negative":
        return "0"
    elif x ==  "Positive":
        return "2"
    else:
        return "1"


Train['label']=Train['Sentiment'].apply(lambda x:classes_def(x))
Test['label']=Test['Sentiment'].apply(lambda x:classes_def(x))


Train.label.value_counts(normalize= True)
```

Text cleaning

```
#Remove Urls and HTML links
def remove_urls(text):
    url_remove = re.compile(r'https?://\S+|www\.\S+')
    return url_remove.sub(r'', text)
Train['text_new']=Train['text'].apply(lambda x:remove_urls(x))
Test['text_new']=Test['text'].apply(lambda x:remove_urls(x))

def remove_html(text):
    html=re.compile(r'<.*?>')
    return html.sub(r'',text)
Train['text']=Train['text_new'].apply(lambda x:remove_html(x))
Test['text']=Test['text_new'].apply(lambda x:remove_html(x))

# Lower casing
def lower(text):
    low_text= text.lower()
    return low_text
Train['text_new']=Train['text'].apply(lambda x:lower(x))
Test['text_new']=Test['text'].apply(lambda x:lower(x))


# Number removal
def remove_num(text):
    remove= re.sub(r'\d+', '', text)
    return remove
Train['text']=Train['text_new'].apply(lambda x:remove_num(x))
Test['text']=Test['text_new'].apply(lambda x:remove_num(x))

#Remove stopwords & Punctuations
from nltk.corpus import stopwords
", ".join(stopwords.words('english'))
STOPWORDS = set(stopwords.words('english'))
```

```
def punct_remove(text):
    punct = re.sub(r"[^\w\s\d]","", text)
    return punct
Train['text_new']=Train['text'].apply(lambda x:punct_remove(x))
Test['text_new']=Test['text'].apply(lambda x:punct_remove(x))


def remove_stopwords(text):
    """custom function to remove the stopwords"""
    return " ".join([word for word in str(text).split() if word not in STOPWORDS])
Train['text']=Train['text_new'].apply(lambda x:remove_stopwords(x))
Test['text']=Test['text_new'].apply(lambda x:remove_stopwords(x))

#Remove mentions and hashtags
def remove_mention(x):
    text=re.sub(r'@\w+','',x)
    return text
Train['text_new']=Train['text'].apply(lambda x:remove_mention(x))
Test['text_new']=Test['text'].apply(lambda x:remove_mention(x))

def remove_hash(x):
    text=re.sub(r'#\w+','',x)
    return text
Train['text']=Train['text_new'].apply(lambda x:remove_hash(x))
Test['text']=Test['text_new'].apply(lambda x:remove_hash(x))

#Remove extra white space left while removing stuff
def remove_space(text):
    space_remove = re.sub(r"\s+"," ",text).strip()
    return space_remove
Train['text_new']=Train['text'].apply(lambda x:remove_space(x))
Test['text_new']=Test['text'].apply(lambda x:remove_space(x))
Test = Test.drop(columns=['text_new'])
Train = Train.drop(columns=['text_new'])
```

```
stemmer = SnowballStemmer('english')
doc_sample = documents[documents['index'] == 4310].values[0][0]

print('original document: ')
words = []
for word in doc_sample.split(' '):
    words.append(word)
print(words)
print('\n\n tokenized and lemmatized document: ')
print(preprocess(doc_sample))
```

```
original document:
['Not', 'a', 'lot', 'of', 'food', 'left', 'in', 'the', 'supermarket', 'tonight.', '', '\r\r\n\r\r\nNo', 'eggs,', 'mea
t,', 'bread,', 'milk,', 'pasta', 'or', 'rice.', 'Plus,', 'of', 'course,', 'no', 'toilet', 'paper', 'or', 'disinfectan
t.', '\r\r\nSome', 'frozen', 'food,', 'fruit', 'and', 'plenty', 'of', 'snacks.', '(Chips,', 'chocolates', 'etc)', '\
r\r\n\r\r\n#sydney', '#coronavirus', '#COVID19au', 'https://t.co/EcZg6KTV5q']


 tokenized and lemmatized document:
['food', 'leav', 'supermarket', 'tonight', 'egg', 'meat', 'bread', 'milk', 'pasta', 'rice', 'plus', 'cours', 'toilet
', 'paper', 'disinfect', 'freez', 'food', 'fruit', 'plenti', 'snack', 'chip', 'chocol', 'sydney', 'coronavirus', 'cov
id', 'https', 'eczg']
```

```
processed_docs = documents['OriginalTweet'].map(preprocess)
processed_docs[:10]
```

```
0    [menyrbi, phil_gahan, chrisitv, https, https, ...
1    [advic, talk, neighbour, famili, exchang, phon...
2    [coronavirus, australia, woolworth, elder, dis...
3    [food, stock, panic, food, need, stay, calm, s...
4    [readi, supermarket, covid, outbreak, paranoid...
5    [news, regionâ, confirm, covid, case, come, su...
6    [cashier, groceri, store, share, insight, covi...
7    [supermarket, today, toilet, paper, rebel, cov...
8    [covid, retail, store, classroom, atlanta, ope...
9    [corona, prevent, stop, thing, cash, onlin, pa...
Name: OriginalTweet, dtype: object
```

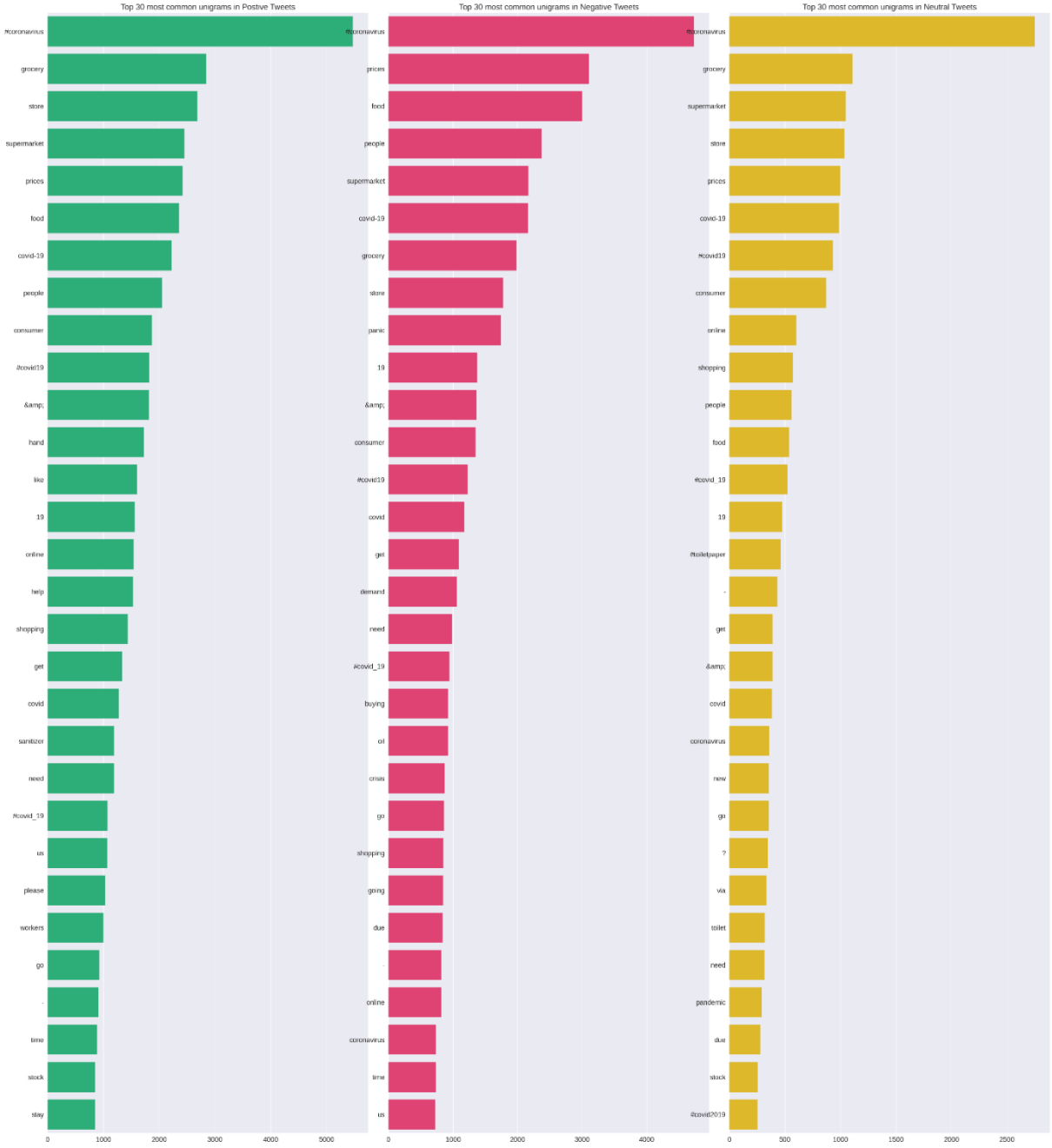
```
dictionary = gensim.corpora.Dictionary(processed_docs)

count = 0
for k, v in dictionary.iteritems():
    print(k, v)
    count += 1
    if count > 10:
        break
```

```
0 chrisitv
1 ghgfzcc
2 https
3 menyrbi
4 nlzdxno
5 phil_gahan
6 account
7 adequ
8 advic
9 chemist
10 contact
```

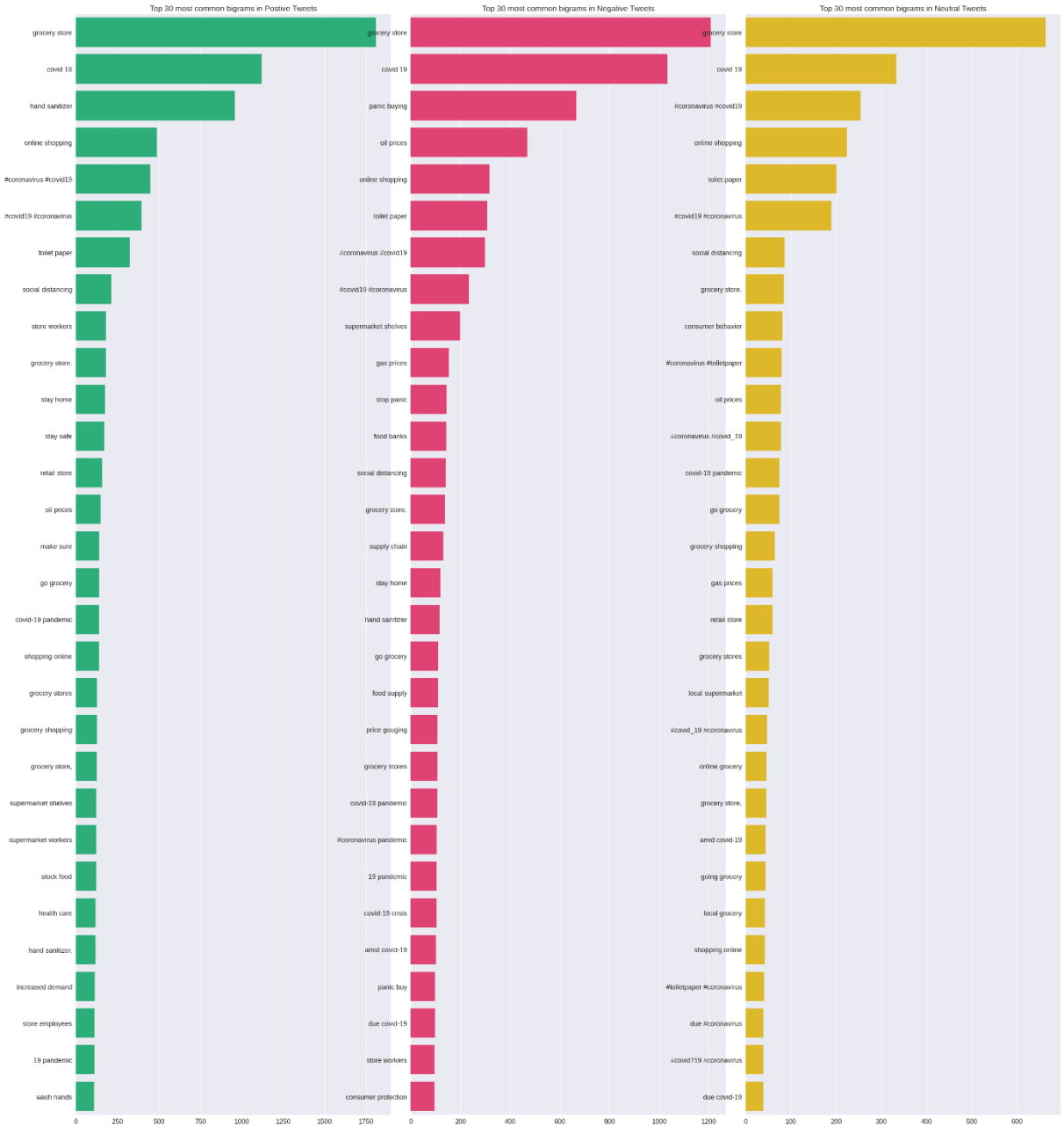
```
bow_doc_4310 = bow_corpus[4310]

for i in range(len(bow_doc_4310)):
    print("Word {} (\"{}\") appears {}time.".format(bow_doc_4310[i][0],

dictionary[bow_doc_4310[i][0]],
bow_doc_4310[i][1]))
```

```
Word 0 ("https") appears 1time.
Word 24 ("coronavirus") appears 1time.
Word 25 ("covid") appears 1time.
Word 35 ("food") appears 2time.
Word 47 ("supermarket") appears 1time.
Word 58 ("paper") appears 1time.
Word 64 ("toilet") appears 1time.
Word 213 ("leav") appears 1time.
Word 305 ("disinfect") appears 1time.
Word 360 ("plus") appears 1time.
Word 561 ("bread") appears 1time.
Word 567 ("freez") appears 1time.
Word 584 ("fruit") appears 1time.
Word 632 ("meat") appears 1time.
Word 703 ("rice") appears 1time.
Word 824 ("pasta") appears 1time.
Word 845 ("chocol") appears 1time.
Word 890 ("milk") appears 1time.
Word 1095 ("tonight") appears 1time.
Word 1170 ("egg") appears 1time.
Word 1173 ("cours") appears 1time.
Word 1306 ("plenti") appears 1time.
Word 1535 ("sydney") appears 1time.
Word 1807 ("chip") appears 1time.
Word 2242 ("snack") appears 1time.
```

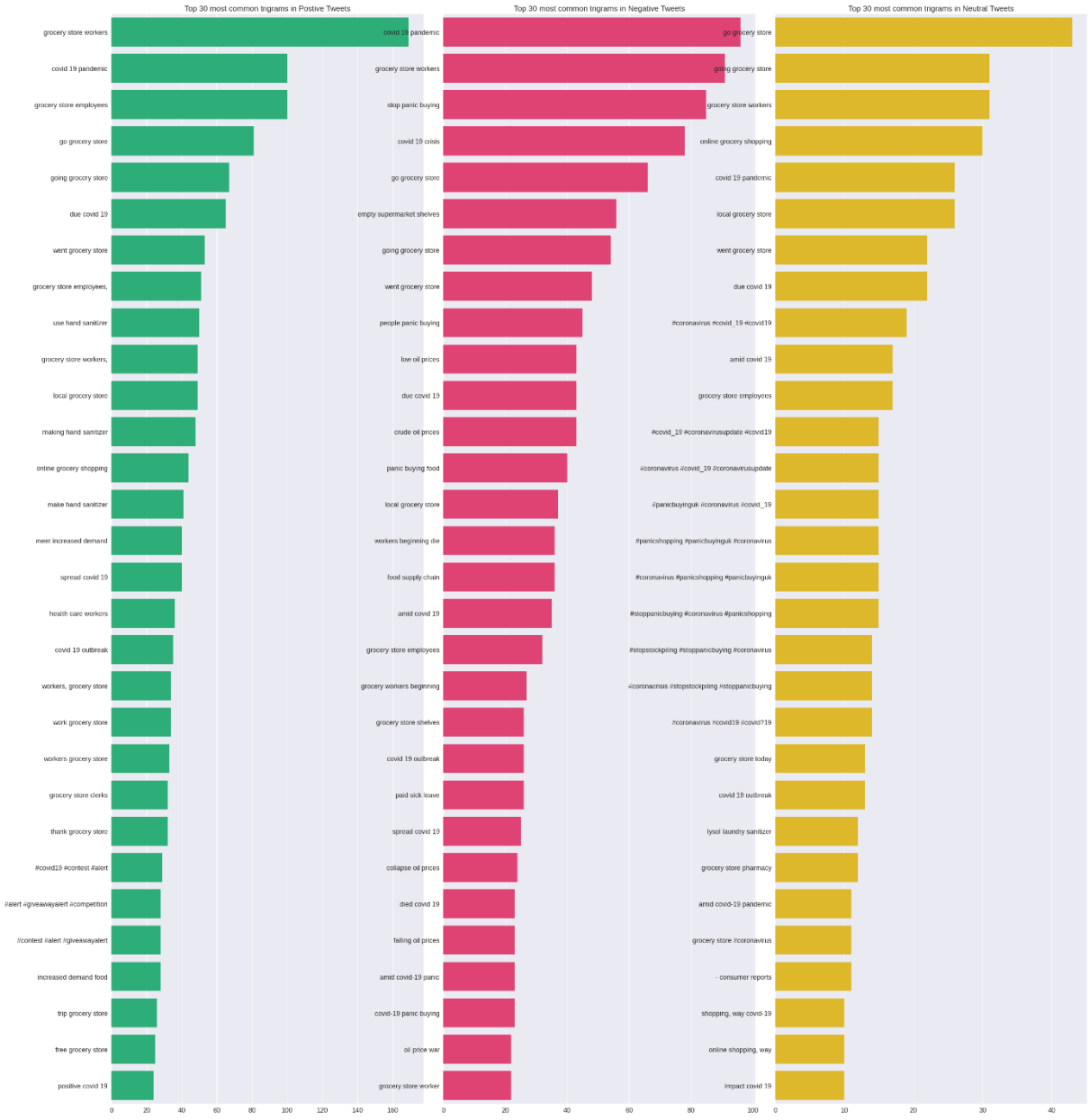
Unigram


Top 30 most common unigrams in Postive Tweets
#coronavirus
grocery
store
supermarket
prices
food
covid-19
people
consumer
#covid19
&
hand
like
19
online
help
shopping
get
covid
sanitizer
need
#covid_19
us
please
workers
go
.
time
stock
stay
0
1000
2000
3000
4000
5000
Top 30 most common unigrams in Negative Tweets
#coronavirus
prices
food
people
supermarket
covid-19
grocery
store
panic
19
&
consumer
#covid19
covid
get
demand
need
#covid_19
buying
oil
crisis
go
shopping
going
due
.
online
coronavirus
time
us
0
1000
2000
3000
4000
Top 30 most common unigrams in Neutral Tweets
#coronavirus
grocery
supermarket
store
prices
covid-19
#covid19
consumer
online
shopping
people
food
#covid_19
19
#toiletpaper
.
get
&
covid
coronavirus
new
go
?
via
toilet
need
pandemic
due
stock
#covid2019
0
500
1000
1500
2000
2500

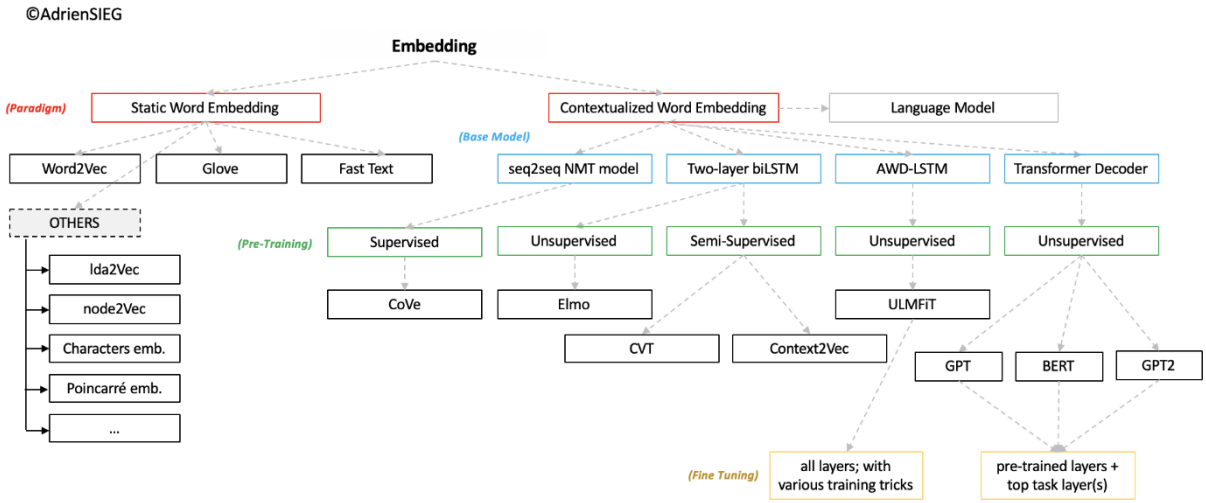
Bigrams


Top 30 most common bigrams in Postive Tweets
grocery store
covid 19
hand sanitizer
online shopping
#coronavirus #covid19
#covid19 #coronavirus
toilet paper
social distancing
store workers
grocery store,
stay home
stay safe
retail store
oil prices
make sure
go grocery
covid-19 pandemic
shopping online
grocery stores
grocery shopping
grocery store,
supermarket shelves
supermarket workers
stock food
health care
hand sanitizer,
increased demand
store employees
19 pandemic
wash hands
0
250
500
750
1000
1250
1500
1750
Top 30 most common bigrams in Negative Tweets
grocery store
covid 19
panic buying
oil prices
online shopping
toilet paper
#coronavirus #covid19
#covid19 #coronavirus
supermarket shelves
gas prices
stop panic
food banks
social distancing
grocery store,
supply chain
stay home
hand sanitizer
go grocery
food supply
price gouging
grocery stores
covid-19 pandemic
#coronavirus pandemic
19 pandemic
covid-19 crisis
amid covid-19
panic buy
due covid-19
store workers
consumer protection
0
200
400
600
800
1000
1200
Top 30 most common bigrams in Neutral Tweets
grocery store
covid 19
#coronavirus #covid19
online shopping
toilet paper
#covid19 #coronavirus
social distancing
grocery store,
consumer behavior
#coronavirus #toiletpaper
oil prices
#coronavirus #covid_19
covid-19 pandemic
go grocery
grocery shopping
gas prices
retail store
grocery stores
local supermarket
#covid_19 #coronavirus
online grocery
grocery store,
amid covid-19
going grocery
local grocery
shopping online
#toiletpaper #coronavirus
due #coronavirus
#covid?19 #coronavirus
due covid-19
0
100
200
300
400
500
600

Trigrams


Top 30 most common trigrams in Postive Tweets
grocery store workers
covid 19 pandemic
grocery store employees
go grocery store
going grocery store
due covid 19
went grocery store
grocery store employees,
use hand sanitizer
grocery store workers,
local grocery store
making hand sanitizer
online grocery shopping
make hand sanitizer
meet increased demand
spread covid 19
health care workers
covid 19 outbreak
workers, grocery store
work grocery store
workers grocery store
grocery store clerks
thank grocery store
#covid19 #contest #alert
#alert #giveawayalert #competition
#contest #alert #giveawayalert
increased demand food
trip grocery store
free grocery store
positive covid 19
0 20 40 60 80 100 120 140 160
Top 30 most common trigrams in Negative Tweets
covid 19 pandemic
grocery store workers
stop panic buying
covid 19 crisis
go grocery store
empty supermarket shelves
going grocery store
went grocery store
people panic buying
low oil prices
due covid 19
crude oil prices
panic buying food
local grocery store
workers beginning die
food supply chain
amid covid 19
grocery store employees
grocery workers beginning
grocery store shelves
covid 19 outbreak
paid sick leave
spread covid 19
collapse oil prices
died covid 19
falling oil prices
amid covid-19 panic
covid-19 panic buying
oil price war
grocery store worker
0 20 40 60 80 100
Top 30 most common trigrams in Neutral Tweets
go grocery store
going grocery store
grocery store workers
online grocery shopping
covid 19 pandemic
local grocery store
went grocery store
due covid 19
#coronavirus #covid_19 #covid19
amid covid 19
grocery store employees
#covid_19 #coronavirusupdate #covid19
#coronavirus #covid_19 #coronavirusupdate
#panicbuyinguk #coronavirus #covid_19
#panicshopping #panicbuyinguk #coronavirus
#coronavirus #panicshopping #panicbuyinguk
#stoppanicbuying #coronavirus #panicshopping
#stopstockpiling #stoppanicbuying #coronavirus
#coronacrisis #stopstockpiling #stoppanicbuying
#coronavirus #covid19 #covid?19
grocery store today
covid 19 outbreak
lysol laundry sanitizer
grocery store pharmacy
amid covid-19 pandemic
grocery store #coronavirus
- consumer reports
shopping, way covid-19
online shopping, way
impact covid 19
0 10 20 30 40

Annexe Les différentes approches de Word embedding



Topic modelling : LDA

```
]: lda_model = gensim.models.LdaMulticore(bow_corpus, num_topics=5, id2word=dictionary, passes=2, workers=2)
```

```
]: for idx, topic in lda_model.print_topics(-1):
       print('Topic: {} \nWords: {}'.format(idx, topic))
```

```
Topic: 0
Words: 0.055*"price" + 0.038*"covid" + 0.036*"coronavirus" + 0.034*"https" + 0.019*"supermarket" + 0.016*"sanit" + 0.
013*"hand" + 0.008*"toilet" + 0.008*"paper" + 0.007*"peopl"
Topic: 1
Words: 0.030*"store" + 0.028*"groceri" + 0.024*"covid" + 0.021*"shop" + 0.020*"coronavirus" + 0.018*"peopl" + 0.016*"
onlin" + 0.014*"worker" + 0.013*"food" + 0.012*"https"
Topic: 2
Words: 0.045*"https" + 0.034*"covid" + 0.026*"supermarket" + 0.016*"coronavirus" + 0.015*"store" + 0.012*"shop" + 0.0
11*"social" + 0.011*"distanc" + 0.011*"peopl" + 0.010*"onlin"
Topic: 3
Words: 0.072*"https" + 0.058*"coronavirus" + 0.047*"covid" + 0.025*"store" + 0.024*"groceri" + 0.016*"supermarket" +
0.014*"toiletpap" + 0.010*"covid_" + 0.010*"home" + 0.010*"shop"
Topic: 4
Words: 0.060*"https" + 0.044*"covid" + 0.032*"consum" + 0.023*"food" + 0.021*"coronavirus" + 0.016*"price" + 0.013*"d
emand" + 0.011*"pandem" + 0.008*"suppli" + 0.008*"sanit"
```