



M1 INFORMATIQUE AIGLE

**HMIN232M**

MÉTHODES DE LA SCIENCE DES DONNÉES

---

# Classification de documents d'opinions

---

**Bachar RIMA**  
**Tasnim SHAQURA**  
**Emile YOUSSEF**

23 avril 2019

# Table des matières

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>   | <b>1</b>  |
| <b>2</b> | <b>Pré-traitements des données</b>                          | <b>2</b>  |
| 2.1      | Préparation à la tokenisation . . . . .                     | 2         |
| 2.2      | Tokenisation et normalisation . . . . .                     | 4         |
| <b>3</b> | <b>Apprentissage du modèle</b>                              | <b>7</b>  |
| 3.1      | Vectorisation . . . . .                                     | 7         |
| 3.2      | Feature-Engineering . . . . .                               | 7         |
| 3.3      | Cross validation et calibrage des hyperparamètres . . . . . | 7         |
| 3.3.1    | Résultats de la cross-validation . . . . .                  | 8         |
| 3.3.2    | Calibrage des hyperparamètres des modèles choisis . . . . . | 8         |
| 3.4      | Création d'un pipeline et sérialisation du modèle . . . . . | 9         |
| 3.4.1    | Pipeline pour Logistic Regression . . . . .                 | 9         |
| 3.4.2    | Pipeline pour Gaussian Naive Bayes . . . . .                | 10        |
| <b>4</b> | <b>Optimisation</b>   | <b>12</b> |
| 4.1      | WordCloud . . . . .   | 12        |
| 4.2      | Traitement de l'ironie . . . . .                            | 12        |
| <b>5</b> | <b>Conclusion</b>   | <b>13</b> |
|          | <b>Annexes</b>  | <b>14</b> |
|          | <b>Annexe A Snapshots des pré-traitements</b>               | <b>15</b> |
|          | <b>Annexe B Snapshots des évaluations des classifieurs</b>  | <b>21</b> |

# Chapitre 1

## Introduction

Aujourd'hui, on trouve sur Internet un grand nombre de données quantitatives, mais aussi et surtout, des données textuelles qualitatives. Ces données se divisent en deux domaines principaux : faits et opinions. D'un côté, les faits reposent sur l'objectivité et la fiabilité des données et de l'autre, les opinions représentent les sentiments de leurs auteurs.

De plus en plus de sites web offrent la possibilité aux clients de laisser leurs avis sur leurs produits. Parfois l'avis d'un client est accompagné par une note facilement interprétable en tant que mesure de satisfaction du client. Cependant, ce n'est pas toujours le cas. Une fouille des avis subjectifs s'avère ainsi essentielle pour apporter des informations supplémentaires sur les sentiments des clients, et par conséquent en tirer les défauts et les avantages des produits concernés. Ceci permet aux vendeurs d'avoir une idée beaucoup plus précise des attentes de leur clientèle afin de mieux y répondre.

L'analyse du sentiment de l'avis d'un internaute se fait de manière naturelle pour un être humain. Cependant, l'explosion actuelle des données ne lui permet plus de les analyser de la même manière. Pour résoudre ce problème, le traitement automatique du langage naturel et la classification des textes sont des tâches inévitables à effectuer par n'importe quelle machine souhaitant traiter des données volumineuses de sentiments. Ces approches permettent de récolter des statistiques sur un texte et de reconnaître et filtrer des motifs y inclus afin de révéler son orientation.

L'objectif de ce projet est de créer une intelligence artificielle qui permettrait, à partir d'une phrase, de reconnaître la polarité de celle-ci : est-ce un avis positif ou bien négatif ? Comment lui apprendre à reconnaître la négation ou bien encore plus complexe, le sarcasme ? Nous essaierons d'y répondre par la suite.

# Chapitre 2

## Pré-traitements des données

Dans le cadre de l'analyse des sentiments, le pré-traitement des documents à classifier est fondamental afin d'obtenir un classifieur généralisable, performant et assez robuste. On distingue ainsi deux grains de pré-traitement :

**document** : traitement de l'ensemble des tokens d'un document collectivement.

**token** : traitement de chaque token d'un document individuellement.

À cet effet, nous utilisons des techniques de pré-traitement :

**syntaxiques** : techniques générales telles que la suppression de contenu superflu (*e.g.* les liens **URL**, les balises **HTML** vides, ...).

**sémantiques** : techniques s'appuyant sur des concepts de **TAL**<sup>1</sup> telles que la lemmatisation basée sur les **POS**<sup>2</sup> tags.

### 2.1 Préparation à la tokenisation

Tout d'abord, nous commençons les pré-traitements au niveau d'un document et nous remarquons, en visualisant un échantillon des documents, qu'il s'agit du langage naturel anglais, sans balisage (cf. Figure A.1).

#### Remplacement des expressions contractées

Nous détectons en premier la présence d'expressions contractées. Or la tokenization qu'on utilisera via la fonction `nlk.tokenize.word_tokenize()` est sensible au guillemet simple<sup>3</sup> « ' » désignant les contractions, il faudra

- 
1. **Traitement Automatique du Langage Naturel**
  2. **Part-Of-Speech**
  3. et par conséquent aux expressions contractées

ainsi les remplacer par leurs expressions complètes équivalentes. Pour ceci, nous utilisons la fonction `contractions.fix(document)`, encapsulée par la fonction wrapper `replace_contractions(document)` (cf. Figure A.2).

**Remarque.** Un autre avantage de traiter les expressions contractées est de traiter le cas de la négation, ayant un effet assez conséquent dans le cadre de l’analyse des sentiments : *e.g.* *I don’t understand*  $\rightarrow$  *I do not understand*.

## Filtrage des balises HTML auto-fermantes et liens URL

Nous remarquons ensuite la présence de balises **HTML** auto-fermantes<sup>4</sup> et de liens **URL**, superflus dans le cadre de l’analyse des sentiments (cf. Figure). Nous les filtrons ainsi en utilisant, respectivement, les fonctions `remove_empty_html_tags(document)` et `remove_urls(document)` basées sur des expressions régulières (cf. Figures A.3 et A.4).

## Nettoyage syntaxique des phrases

Enfin, nous observons la présence de phrases syntaxiquement mal terminées et/ou mal commencées (cf. Figure), et dont les effets s’avèrent non-favorables lors de la tokenisation. En effet, la tokenisation d’un document les contenant résulte en un token problématique ayant la forme *X.Y* désignant l’un des motifs suivants :

**sentence\_ends\_alphabetic(.+)sentence\_starts\_alphabetic** : une phrase qui se termine par une lettre de l’alphabet, suivie d’un ou plusieurs points, suivi(s) d’une lettre de l’alphabet d’une phrase qui commence, sans aucun espace blanc entre la fin de la 1<sup>re</sup> et le début de la 2<sup>e</sup>.

**sentence\_ends\_digit(.+)sentence\_starts\_alphabetic** : *idem* que le 1<sup>er</sup> mais avec une phrase qui se termine par un chiffre et une phrase qui commence par une lettre de l’alphabet.

**sentence\_ends\_alphabetic(.+)sentence\_starts\_digit** : *idem* mais avec une phrase qui se termine par une lettre de l’alphabet et une phrase qui commence par un chiffre.

Pour pallier ce problème, nous utilisons une fonction `clean_sentence_anchors(document)` basée sur des expressions régulières et permettant d’obtenir des phrases bien terminées/commencées (cf. Figure A.5).

**Remarque.** Le motif `digit(.+)digit` n’est pas pris en compte, vu que les expressions régulières ne pourrions ainsi le distinguer d’un nombre à virgule.

---

4. *e.g.* `<br />`

Ce cas reste ainsi non traité, vu qu'il est plus délicat à contourner, surtout que la présence de nombres en tant qu'amplificateurs d'émotions pourrait être utile pour l'analyse des sentiments, notamment en utilisant les **n-grams** lors de la vectorisation et la sélection des features.

## 2.2 Tokenisation et normalisation

La deuxième étape de pré-traitement s'effectue au niveau des tokens d'un document.

### Normalisation Unicode, encodage ASCII et conversion en minuscule

Afin de n'avoir que des caractères **ASCII** normalisés à traiter, nous effectuons la démarche suivante au sein de la fonction wrapper `remove_non_ascii(tokens)` sur chaque token (cf. Figure A.6) :

1. une normalisation Unicode **NKFD**<sup>5</sup> via la méthode `unicodedata.normalize('NFKD', token)`.
2. un encodage **ASCII** des caractères normalisés via la méthode `token.encode('ASCII', 'ignore')` en ignorant les caractères non **ASCII**.
3. un décodage **UTF-8** des octets encodés en **ASCII** via la méthode `token.decode('utf-8', 'ignore')` prenant en compte tous les caractères **ASCII**.

Une fois les caractères encodés en **ASCII**, nous les convertissons en minuscule en utilisant la fonction `token.lower()`, encapsulée par la fonction wrapper `to_lowercase(tokens)` (cf. Figure A.7).

**Remarque.** Au cours du projet, nous nous sommes rendu compte que le traitement de motifs contenant des caractères majuscules aurait pu être intéressant dans le cadre de l'analyse des sentiments. Effectivement, étant donné que notre approche primaire était celle du machine learning et non pas celle d'une analyse par lexicons<sup>6</sup>, une approche hybride aurait été encore plus intéressante, mais malheureusement plus chronophage.

### Découpage des tokens composés, remplacement des chiffres et filtrage des caractères de ponctuation

Suite à la conversion des tokens en minuscule, nous supprimons les caractères de ponctuation via une fonction `remove_punctuation(tokens)` basée

---

5. **Normalization Form Compatibility Decomposition**

6. celle-ci nécessitant plus de connaissances en **TAL**

sur des expressions régulières. Cependant, nous nous trouvons ainsi avec des tokens mal formés obtenus par la concaténation de plusieurs tokens. En analysant les résultats, nous trouvons que ces tokens étaient concaténés au préalable par des caractères de ponctuation tels que `_`, `-`, `~`, `...` qui ont été supprimés (cf. Figure A.8). Ainsi nous reportons la suppression des caractères de ponctuation.

Nous commençons alors par découper ces tokens composés, en utilisant la fonction `split_on_character_set(tokens, characters)` basée sur des expressions régulières (cf. Figure A.9).

Après, nous remplaçons les tokens désignant des nombres en chiffres par leurs équivalents en lettres en utilisant la fonction wrapper `replace_numbers(tokens)`, encapsulant la fonction `inflect.engine().number_to_words(token)` (cf. Figure).

Enfin, nous supprimons les caractères de ponctuation en utilisant la fonction susmentionnée `remove_punctuation(tokens)` (cf. Figure A.11).

**Remarque.** Le coût d'utiliser la fonction de découpage des tokens composés est l'éventuelle perte de sémantique lors de la séparation des mots composés<sup>7</sup>. La fonction de suppression des caractères de ponctuation, quant à elle, supprime la possibilité d'analyser les sentiments à travers des motifs matchant des expressions contenant des caractères tels que «!», «?», voire même les `...`. Toutefois, bien que nous avons sous-estimé les conséquences d'utiliser ces fonctions, les dégâts étaient négligeables par rapport à l'ensemble du dataset. Mais il fallait quand même bien le noter.

## Suppression des stopwords et lemmatisation

Après avoir effectué les pré-traitements syntaxiques, nous appliquons des pré-traitements sémantiques s'appuyant sur des notions du **TAL**, notamment le traitement des **stopwords** et la lemmatisation basée sur les **POS** tags.

Pour le traitement des **stopwords**, nous avons considéré le dictionnaire des **stopwords** offert par NLTK pour le langage anglais. Ensuite, nous en avons supprimé le mot «*not*», celui-ci étant utilisé pour le traitement de la négation lors de la vectorisation. Puis, nous y avons ajouté des **stopwords** désignant les mots neutres les plus fréquents<sup>8</sup> relativement au domaine de la cinéma (*e.g.* *actor* et ses inclinaisons, *movie* et ses synonymes et inclinaisons, `...`). Enfin, la suppression des **stopwords** était effectuée par la fonction wrapper `remove_stopwords(tokens, stopwords)`.

---

7. *e.g.* *Well-being*, *first-hand*, `...`

8. cf. le fichier `utility_ML.py` pour plus d'informations

La dernière étape de prétraitement consiste en la lemmatisation des tokens en s'appuyant sur leurs catégories grammaticales grâce au **POS-Tagging**. Pour ce faire, nous jugeons, dans le cadre de l'analyse des sentiments, l'importance de considérer les catégories `{noms9, verbes, adjectifs, adverbes}`

Ainsi, pour chaque collection de tokens, les tokens sont taggés par le **POS-Taggeur** collectivement en utilisant la fonction `nltk.pos_tag(tokens)` et ensuite lemmatisés individuellement en considérant leurs tags en utilisant le dictionnaire des catégories à considérer et la fonction `nltk.stem.WordNetLemmatizer().lemmatize(pos_tag)`. L'avantage de cette approche est de convertir un token au lemma le plus sémantiquement cohérent, prenant en compte le contexte du token lemmatisé (cf. Figure A.13).

Une fois la lemmatisation effectuée, nous joignons les tokens normalisés de chaque document afin de pouvoir le vectoriser après.

---

9. catégorie par défaut



# Chapitre 3

## Apprentissage du modèle

### 3.1 Vectorisation

### 3.2 Feature-Engineering

### 3.3 Cross validation et calibrage des hyperparamètres

Une fois la vectorisation effectuée, nous préparons un ensemble de modèles à tester sur l'ensemble des features choisies. Pour assurer la bonne performance des classifieurs, nous utilisons une cross-validation sur 10 partitions différentes du dataset et la métrique **Accuracy** pour évaluer leurs performances.

Pour chaque modèle, nous calculons le score pour chaque partition, puis le score moyen et la déviation standard de l'ensemble des scores de toutes les partitions. Pour ce faire nous utilisons la fonction `cross_val_score()` de la bibliothèque `scikit-learn` et l'objet `KFold` pour choisir les partitions et leur nombre et leur préparer pour la cross-validation.

Vu qu'il s'agit d'un problème de classification binaire, les modèles qui paraissent adaptés sont soit **Logistic Regression**, soit un **SVM**, soit un modèle probabiliste tel que **Naive Bayes**, soit un modèle aléatoire adapté au données volumineuses tel que **Stochastic Gradient Descent**. Mais vu que la science de la données est un domaine empirique, la notion de meilleur modèle pour un dataset n'existe pas vraiment. Nous essayons ainsi ces modèles et d'autres modèles aussi, qu'ils soient adaptés ou pas, afin d'expérimenter.

| Modèle                 | Score moyen | Déviati  n standard |
|------------------------|-------------|---------------------|
| LinearSVC              | 92%         | 1%                  |
| SGDClassifier          | 92%         | 1%                  |
| LogisticRegression     | 91%         | 0.8%                |
| GaussianNB             | 84%         | 1%                  |
| RandomForestClassifier | 81%         | 1%                  |
| KNeighborsClassifier   | 79%         | 1%                  |
| DecisionTreeClassifier | 75%         | 0.8%                |

TABLE 3.1 – R  sultats de la cross-validation des mod  les choisis

### 3.3.1 R  sultats de la cross-validation

En appliquant la cross-validation sur l’ensemble des mod  les choisis, nous nous retrouvons avec les r  sultats de la table 3.1 ordonn  s par ordre d  croissant sur les scores moyens.

En s’appuyant sur ces r  sultats, nous choisissons ainsi les mod  les **Logistic Regression** (91%, 0.8%) et **Linear SVM** (92%, 1%) pour continuer l’optimisation de leurs apprentissages. Nous aurons   galement pu choisir le mod  le **Stochastic Gradient Descent** au lieu du mod  le **Linear SVM**, mais vu que **Stochastic Gradient Descent** est d  pendant du hasard pour avoir une bonne performance, nous ne privil  gions pas son utilisation.

### 3.3.2 Calibrage des hyperparam  tres des mod  les choisis

Suite    l’  tape de cross-validation, il faut trouver les meilleurs hyperparam  tres permettant de raffiner les r  gions de d  cision de chaque mod  le choisi, afin d’avoir les meilleurs pr  dictions possibles. Ceci est effectu   par une recherche de grille, permettant de tester diff  rentes combinaisons des valeurs des hyperparam  tres fournis pour chaque mod  le. Pour ce faire nous utilisons l’objet `GridSearchCV` de la biblioth  que `scikit-learn`.

Un autre point en faveur de `GridSearchCV` c’est qu’il permet d’effectuer une cross-validation pour chaque combinaison des hyperparam  tres d’un mod  le. Nous choisissons ainsi de r  p  ter le processus sur 5 partitions diff  rentes du dataset pour chaque mod  le, en utilisant la m  trique **Accuracy** pour   valuer leurs diff  rents calibrages.

Pour le mod  le **Logistic Regression**, les hyperparam  tres    calibrer sont :

- C** : la valeur de l’inverse de la r  gularization pour la r  gression (sauts de  $10^k$  avec  $k \in [-4; 4]$ )

| Modèle             | Score moyen | Meilleurs calibrages           |
|--------------------|-------------|--------------------------------|
| LogisticRegression | 90%         | $C = 11.288$ ; $penalty = L_2$ |
| LinearSVC          | 90%         | $C = 1$                        |

TABLE 3.2 – Résultats de la cross-validation de la recherche de grille des modèles choisis

**P** : la norme à choisir pour les pénalités ( $L_1$  et  $L_2$ )

Pour le modèle **Linear SVM**, nous choisissons de calibrer l’hyperparamètre **C**, ayant comme valeurs possibles les sauts de  $10^k$  avec  $k \in [-4; 2]$

### Résultats du GridSearchCV

En appliquant la cross-validation lors de la recherche de grille sur l’ensemble des modèles choisis, nous nous retrouvons avec les résultats de la table 3.2 ordonnés par ordre décroissant sur les scores moyens.

En s’appuyant sur ces résultats, nous avons choisi le modèle **Logistic Regression** avec les meilleurs calibrages de ses hyperparamètres (90%,  $C = 11.288$ ,  $penalty = L_2$ ) pour apprendre le modèle et effectuer la prédiction.

## 3.4 Création d’un pipeline et sérialisation du modèle

Après avoir choisi le modèle et ses hyperparamètres calibrés, nous créons un pipeline permettant d’enchaîner les étapes de pré-traitement, la vectorisation et l’apprentissage d’un modèle.

### 3.4.1 Pipeline pour Logistic Regression

Le modèle **Logistic Regression** étant le meilleur modèle choisi et dont les hyperparamètres sont bien calibrés, nous lui créons un pipeline dédié en utilisant la fonction `sklearn.pipeline.Pipeline(vectoriser, eventuel_transformateur, classifieur)` en suivant le schéma suivant :

1. pour la vectorisation, nous utilisons un `sklearn.feature_selection.TfidfVectorizer` en lui passant la fonction de pré-traitement wrapper `preprocess(document)` de l’ensemble des fonctions vu précédemment. Le résultat est une matrice de fréquences des termes, utilisée pour l’apprentissage.
2. pour obtenir les jeux d’entraînement et de test, nous utilisons la fonction `sklearn.model_selection.train_test_split(X, y, training_size, test_size, random_state)`.

3. pour l'apprentissage nous utilisons la fonction `pipeline.fit(training_features, training_labels)`.
4. pour essayer le modèle appris sur le jeu de données de test afin de prédire les labels associés, nous utilisons la fonction `pipeline.predict(test_features)`.
5. pour évaluer la performance du modèle appris, nous s'appuyons sur les mesures d'évaluation suivantes avec leurs fonctions correspondantes :

**accuracy** : `sklearn.metrics.accuracy_score(results, test_labels)` ;

**confusion matrix** : `sklearn.metrics.confusion_matrix(test_labels, results)` ;

**precision, recall, f1-score** : `sklearn.metrics.classification_report(test_labels, results)`.

6. pour sérialiser le modèle appris avec le module `pickle`, nous utilisons la fonction `pickle.dump(pipeline, open(pipeline_file, 'wb'))`.

Afin d'évaluer davantage la performance du pipeline, nous essayons de prédire les labels de deux datasets :

- un dataset de test fournis dans le cadre du projet, comportant 4000 lignes (2000 *avis positifs* et 2000 *avis négatifs*) ;
- un dataset de reviews **IMDB**, téléchargé par nous-même, comportant 10000 lignes (5000 *avis positifs* et 5000 *avis négatifs*).

Les résultats des prédictions et les évaluations sont consultables dans les figures [B.1](#), [B.2](#) et [B.3](#).

### 3.4.2 Pipeline pour Gaussian Naive Bayes

Afin de visualiser l'effet d'utiliser un modèle probabiliste dans le cadre de l'analyse des sentiments, nous décidons de créer un pipeline pour le modèle **Gaussian Naive Bayes**. Ce modèle n'admet pas d'hyperparamètres à calibrer sauf **priors**, désignant les probabilités estimées au préalable pour chacune des classes. Toutefois, nous ne touchons pas à cet hyperparamètre afin qu'il s'adapte dynamiquement aux données<sup>1</sup>.

Le schéma de création d'un pipeline pour le modèle **Gaussian Naive Bayes** est identique à celui du modèle **Logistic Regression**, à la différence qu'il faut lui ajouter un transformateur intermédiaire des données, précédant la phase d'apprentissage. En effet, ce modèle travaille avec des matrices denses (*i.e. dense matrices*) pour apprendre le modèle, au lieu de matrices creuses (*i.e. sparse matrices*), telles que celles fournies par la vectorisation. Il faut alors utiliser un transformateur permettant d'effectuer cette

---

1. cf. [https://scikit-learn.org/stable/modules/generated/sklearn.naive\\_bayes.GaussianNB.html](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html)

étape avant l'étape de "*fitting*" du pipeline. Nous définissons donc une classe `DenseTransformer`, héritant de la classe de base des transformateurs du module `scikit-learn TransformerMixin`, accomplissant cette tâche et fournissant les données transformées à la phase d'apprentissage.

Enfin nous évaluons la performance du pipeline en l'essayant sur les mêmes datasets utilisés pour le modèle **Logistic Regression**. Les résultats des prédictions et les évaluations sont consultables dans les figures [B.4](#), [B.5](#) et [B.6](#).

# Chapitre 4

## Optimisation

### 4.1 WordCloud

### 4.2 Traitement de l'ironie

Chapitre 5

Conclusion

# Annexes



# Annexe A

## Snapshots des pré-traitements

```
In [7]: 1 opinions = df['Avis'][:10]
        2 for opinion in opinions:
        3     print(opinion + "\n")
```

When the young Kevin gets the boat of his dead uncle as a gift, he invites five friends of him to a trip to Catalina Isl and for the weekend. While in the journey, they drink booze, have sex and play games, with each one of them telling his or her greatest fear. Later Kevin drowns in the open sea, the engine stops, and they are haunted and murdered by their greatest innermost fear. Yesterday, my wife, son, daughter and three other friends joined to watch "Haunted Boat" on DVD. With less than 30 minutes running time, the group gave up watching this messy and boring amateurish piece of crap, and we decided to see another film. Later, I decided to watch the rest of this flick to see how bad it could be and it would have been better off going to bed to sleep. The confused story has an awful cinematography and camera work, with a cast that is probably studying to be actors and actresses and in the end this film seems to be a bad project of cinema school. The terrible and pretentious screenplay shows a ridiculous twist in the end, actually a complete mess that made me not understand what the story is all about. Was the girl insane and traveled alone in the boat, imagining the whole situation with imaginary friends? If that is true, are their friend again in the very end fruit of her madness? My vote is one. Title (Brazil): "Viagem Para a Morte" ("Trip to the Death")

Since I first saw Anchors Aweigh in 1945, viewing it on videotape holds a lot of nostalgia for me. At age 15, it was easy for me to be drawn into the first of the great MGM Technicolor musicals. Now I am perhaps most interested in thinking about the future careers of the leading players. Though Sinatra had done a couple of negligible films soon after his emergence after his Dorsey days, as a solo singer, this was his first major film appearance. As another viewer noted, this seems almost to be a warm-up for On the Town. Sinatra may have had to work hard at it, but his dance with Kelly is credible, and he would do better in their next pairings. However, observing his physique, it's easy to see why he was caricatured as a string bean. Who would have imagined that within a decade he would win an academy award for acting, and go on to play many roles as a tough detective or leader in combat. Though Gene Kelly's personality and dancing dominated this film, his winsome performance did not suggest that he would become a major creative force, almost the iconic figure, for MGM musicals, where he developed a style of dance complementary to that of Fred Astaire. Finally, it was strange to see

FIGURE A.1 – échantillon de documents

Before replacing contractions:

=====

When I saw the commercial for this, I was all about seeing it. Now, forgive me, but it's been so long since I've seen it that I don't recall how it went. Suffice it to say, the movie I saw bore no resemblance to the "movie" they sold me on. I was bored, annoyed, and incredibly disappointed by this movie. And if it wasn't bad enough, they had to sink it even further with that awful reggae music. Not exactly mood-setting music for a horror movie, eh mon? I guess if you never saw the commercial (or trailer, I suppose) you may think this is some hot stuff. For my money, the commercial was way better.

After replacing contractions:

=====

When I saw the commercial for this, I was all about seeing it. Now, forgive me, but it is been so long since I have seen it that I do not recall how it went. Suffice it to say, the movie I saw bore no resemblance to the "movie" they sold me on. I was bored, annoyed, and incredibly disappointed by this movie. And if it was not bad enough, they had to sink it even further with that awful reggae music. Not exactly mood-setting music for a horror movie, eh mon? I guess if you never saw the commercial (or trailer, I suppose) you may think this is some hot stuff. For my money, the commercial was way better.

FIGURE A.2 – remplacement des expressions contractées dans un document

Before removing empty HTML tags:  
=====

Rather like Paul Newman and Steve McQueen with their racing car movies this has all the appearance of a "jollies" project for Robert Redford, as he gets to ski up hill and down dale in the Alpine sunshine.<br /><br />The story is as light as powdered snow with Redford's small-town boy David Chappellet (what kind of lead name is that?) who with his eyes on the prize of Olympic glory, gets up the nose of, in no particular order, his coach, father and team-mates. Women are a mere side-show in his insular world as evidenced by a fairly distasteful pick-up scene with an old girlfriend in his hometown and then his selfishly petulant pursuit of, heavens above, a free-thinking, independent woman, played by Camilla Sparv. The ski-ing sequences are fine with some good stunt-work involving numerous bumps and scrapes on the piste but their effectiveness is dimmed by our subsequent familiarity with top TV coverage of skiing events down to the present day. Plus I'm not convinced that the Winter Olympics has the same mass identification with the general public as the summer games so that when Redford eventually wins his gold medal in the final reel, I couldn't really be that excited for him one way or another.<br /><br />Of the actors, Redford, best profile forward, doesn't need to do much and indeed doesn't, while Gene Hackman does better with equally meagre material. Ms Sparv does well as the chief female interest well who treats Redford the way he's doubtless treated every other woman in his chauvinistic way.<br /><br />In truth though, there's a lack of dramatic tension throughout for which the action sequences don't fully compensate and you don't care a fig for any of the leading characters. One of those films where the actors probably enjoyed making it more than the viewers did watching it....

After removing empty HTML tags:  
=====

Rather like Paul Newman and Steve McQueen with their racing car movies this has all the appearance of a "jollies" project for Robert Redford, as he gets to ski up hill and down dale in the Alpine sunshine. The story is as light as powdered snow with Redford's small-town boy David Chappellet (what kind of lead name is that?) who with his eyes on the prize of Olympic glory, gets up the nose of, in no particular order, his coach, father and team-mates. Women are a mere side-show in his insular world as evidenced by a fairly distasteful pick-up scene with an old girlfriend in his hometown and then his selfishly petulant pursuit of, heavens above, a free-thinking, independent woman, played by Camilla Sparv. The ski-ing sequences are fine with some good stunt-work involving numerous bumps and scrapes on the piste but their effectiveness is dimmed by our subsequent familiarity with top TV coverage of skiing events down to the present day. Plus I am not convinced that the Winter Olympics has the same mass identification with the general public as the summer games so that when Redford eventually wins his gold medal in the final reel, I could not really be that excited for him one way or another. Of the actors, Redford, best profile forward, does not need to do much and indeed does not, while Gene Hackman does better with equally meagre material. Ms Sparv does well as the chief female interest well who treats Redford the way he is doubtless treated every other woman in his chauvinistic way. In truth though, there is a lack of dramatic tension throughout for which the action sequences do not fully compensate and you do not care a fig for any of the leading characters. One of those films where the actors probably enjoyed making it more than the viewers did watching it....

FIGURE A.3 – suppression des balises auto-fermantes d'un document

Before removing URLs:  
=====

Did Uwe Boll seriously just rip off the basic idea and dialogue from Se7en?! Why is it so fucking difficult for this douchebag to be original?! He even mentioned in an interview with Gametrailers that he chooses stuff like games to make into movies because the characters, plots, backstories and so on are already there and ready for him to screw with.<br /><br />Guess it isn't too much of a stretch for him to rip off another movie entirely...<br /><br />I mean, seriously, what the hell...? Here's something I made in Uwe's 'honor'...<br /><br /><http://zuucka.deviantart.com/art/Uwe-Boll-is-a-Douchebag-70369862>...

After removing URLs:  
=====

Did Uwe Boll seriously just rip off the basic idea and dialogue from Se7en?! Why is it so fucking difficult for this douchebag to be original?! He even mentioned in an interview with Gametrailers that he chooses stuff like games to make into movies because the characters, plots, backstories and so on are already there and ready for him to screw with. Guess it is not too much of a stretch for him to rip off another movie entirely... I mean, seriously, what the hell...? Here's something I made in Uwe's 'honor'...

FIGURE A.4 – suppression des URLs d'un document

Before cleaning sentence anchors:  
=====

This movie (even calling it a movie is an overstatement) is ridiculously horrible. Normally a huge fan of Eric Roberts in "B" list movies, this tragedy of a flick makes me question his real B list clout! And Charlie, please go back to hoping for a Diagnosis Murder revival rather than this.....you can't blame the nameless eye candy (uhhum...beauty pageant members) for participating in this weak movie, but YOU are a former TV star man! Pull yourself together. Don't even get me started on Stuart Pankin. For the sake of all that is good about Stuart, you should have seen this was not necessarily a real movie! Bryan Michael Stoller exemplifies absolute genius only in the fact that he was able to dupe anyone into investing in this picture (money or time).  
<br /><br />Really, this was no parody or spoof movie although it tries on a 2nd grade level. Mostly, it is poor writing and acting and camera work and editing and....well poor everything. I watched it because I read an article in some mag about agent MJ's involvement and my interest was peaked due to the lawsuit in which he was involved. I now wonder if the only reason they show him from the shoulders up in the movie is because he, like at the trial, showed up wearing pajama bottoms and barely lucid (wait a second, is he ever really considered lucid?...I digress). And Agent MJ? Is that the best they could come up with for a name for his character? Sheez. What a startling piece of originality! Or, maybe that was supposed to be funny? Putting Marriott into the movie was a nice touch at first, but overdone and annoying after all is said and done.<br /><br />Spare yourself the grief of watching.....don't say I didn't warn you....

After cleaning sentence anchors:  
=====

This movie (even calling it a movie is an overstatement) is ridiculously horrible. Normally a huge fan of Eric Roberts in "B" list movies, this tragedy of a flick makes me question his real B list clout! And Charlie, please go back to hoping for a Diagnosis Murder revival rather than this..... you cannot blame the nameless eye candy (uhhum... beauty pageant members) for participating in this weak movie, but YOU are a former TV star man! Pull yourself together. do not even get me started on Stuart Pankin. For the sake of all that is good about Stuart, you should have seen this was not necessarily a real movie! Bryan Michael Stoller exemplifies absolute genius only in the fact that he was able to dupe anyone into investing in this picture (money or time). Really, this was no parody or spoof movie although it tries on a 2nd grade level. Mostly, it is poor writing and acting and camera work and editing and.... well poor everything. I watched it because I read an article in some mag about agent MJ's involvement and my interest was peaked due to the lawsuit in which he was involved. I now wonder if the only reason they show him from the shoulders up in the movie is because he, like at the trial, showed up wearing pajama bottoms and barely lucid (wait a second, is he ever really considered lucid?...I digress). And Agent MJ? Is that the best they could come up with for a name for his character? Sheez. What a startling piece of originality! Or, maybe that was supposed to be funny? Putting Marriott into the movie was a nice touch at first, but overdone and annoying after all is said and done. Spare yourself the grief of watching..... do not say I did not warn you....

FIGURE A.5 – nettoyage syntaxique des phrases mal commencées/terminées d'un document

Before Unicode normalization and ASCII encoding:  
=====

In the early 00's, production companies had a short-lived craze for supernatural genre movies in France after "The Crimson Rivers" and "Brotherhood of the Wolf" turned out to be hits, so several movies were green-lit or saved from their "direct-to-video" fate. However, France, as opposed to the US, UK or Italy, has little tradition of fantasy B-movies and it turned out quickly that "Samourais", "Bloody Mallory" or the "Crimson Rivers" sequel were ill-advised attempts at recreating a kind of magic that had never existed in French cinema in the first place. As they flopped, producers have gone back to their usual fare: derivative farces or the umpteenth self-referential tribute to French New Wave by a former critic from "Les Cahiers du cinéma".<br /><br />"Brocéliande" could only have been green-lit during this short window, as it serves no other discernible purpose. It's your by-the-book

After Unicode normalization and ASCII encoding:  
=====

[ 'In', 'the', 'early', '00', 's', ',', 'production', 'companies', 'had', 'a', 'short-lived', 'craze', 'for', 'supernatural', 'genre', 'movies', 'in', 'France', 'after', '...', 'The', 'Crimson', 'Rivers', '...', 'and', '...', 'Brotherhood', 'of', 'the', 'Wolf', '...', 'turned', 'out', 'to', 'be', 'hits', '...', 'so', 'several', 'movies', 'were', 'green-lit', 'or', 'saved', 'from', 'their', '...', 'direct-to-video', '...', 'fate', '...', 'However', '...', 'France', '...', 'as', 'opposed', 'to', 'the', 'US', '...', 'UK', 'or', 'Italy', '...', 'has', 'little', 'tradition', 'of', 'fantasy', 'B-movies', 'and', 'it', 'turned', 'out', 'quickly', 'that', '...', 'Samourais', '...', '...', 'Bloody', 'Mallory', '...', 'or', 'the', '...', 'Crimson', 'Rivers', '...', 'sequel', 'were', 'ill-advised', 'attempts', 'at', 'recreating', 'a', 'kind', 'of', 'magic', 'that', 'had', 'never', 'existed', 'in', 'French', 'cinema', 'in', 'the', 'first', 'place', '...', 'As', 'they', 'flopped', '...', 'producers', 'have', 'gone', 'back', 'to', 'their', 'usual', 'fare', '...', 'derivative', 'farces', 'or', 'the', 'umpteenth', 'self-referential', 'tribute', 'to', 'French', 'New', 'Wave', 'by', 'a', 'former', 'critic', 'from', 'the', 'Les', 'Cahiers', 'du', 'cinéma', '...', '...', '...', 'Brocéliande', '...', 'could', 'only', 'have', 'been', 'green-lit', 'during', 'this', 'short', 'window', '...', 'as', 'it', 'serves', 'no', 'other', 'discernible', 'purpose', '...', 'it', 'is', 'your', 'by-the-book' ]

FIGURE A.6 – normalisation Unicode, et encodage ASCII des tokens d'un document

```

Before conversion of characters to lower case:
=====
I never trust the opinions of anyone regarding a film. That goes for critics as well. Sure, if it gets positive reviews that's OK and a plus, but most films that get critical rave I hate. I enjoyed this film for what it was, an entertaining film. It takes you out of your life for a couple hours and into a fictional character... that being Catherine Trammell. Sharon Stone is awesome in this role, just like she was in the first one. Anyone who says she is horrible in this film must have felt the same in the first one b/c she is back acting the same way she did in Basic Instinct 1. Catherine is hers and she plays her to perfection. Her one liners are great, much like in the first one. Who can forget in the first film when she tells the cops, "If you're gonna arrest me do it...otherwise get the f**k out of here!" Great scene, and believe me, she does it again in this one. I was captivated by her. Her outfits, the way she smoked her cigarettes, believe me, its worth the price just to see Stone's performance. I cannot wait for this film to be released on DVD, uncut, because I can only imagine how much better it is going to be. And yes, there are lots of twists, as in the first one, including the ending!

After conversion of characters to lower case:
=====
['i', 'never', 'trust', 'the', 'opinions', 'of', 'anyone', 'regarding', 'a', 'film', '.', 'that', 'goes', 'for', 'critics', 'as', 'well', '.', 'sure', '.', 'if', 'it', 'gets', 'positive', 'reviews', 'that', 'is', 'ok', 'and', 'a', 'plus', '.', 'but', 'most', 'films', 'that', 'get', 'critical', 'rave', 'i', 'hate', '.', 'i', 'enjoyed', 'this', 'film', 'for', 'what', 'it', 'was', '.', 'an', 'entertaining', 'film', '.', 'it', 'takes', 'you', 'out', 'of', 'your', 'life', 'for', 'a', 'couple', 'hours', 'and', 'into', 'a', 'fictional', 'character', '...', 'that', 'being', 'catherine', 'trammell', '.', 'sharon', 'stone', 'is', 'awesome', 'in', 'this', 'role', '.', 'just', 'like', 'she', 'was', 'in', 'the', 'first', 'one', '.', 'anyone', 'who', 'says', 'she', 'is', 'horrible', 'in', 'this', 'film', 'must', 'have', 'felt', 'the', 'same', 'in', 'the', 'first', 'one', 'b/c', 'she', 'is', 'back', 'acting', 'the', 'same', 'way', 'she', 'did', 'in', 'basic', 'instinct', '1', '.', 'catherine', 'is', 'hers', 'and', 'she', 'plays', 'her', 'to', 'perfection', '.', 'her', 'one', 'liners', 'are', 'great', '.', 'much', 'like', 'in', 'the', 'first', 'one', '.', 'who', 'can', 'forget', 'in', 'the', 'first', 'film', 'when', 'she', 'tells', 'the', 'cops', '.', 'if', 'you', 'are', 'going', 'to', 'arrest', 'me', 'do', 'it', '...', 'otherwise', 'get', 'the', 'f**k', 'out', 'of', 'here', '!', 'great', 'scene', '.', 'and', 'believe', 'me', 'she', 'does', 'it', 'again', 'in', 'this', 'one', '.', 'i', 'was', 'captivated', 'by', 'her', '...', 'her', 'outfits', '...', 'the', 'way', 'she', 'smoked', 'her', 'cigarettes', '...', 'believe', 'me', '...', 'its', 'worth', 'the', 'price', 'just', 'to', 'see', 'stone', 's', 'performance', '.', 'i', 'can', 'not', 'wait', 'for', 'this', 'film', 'to', 'be', 'released', 'on', 'dvd', '.', 'uncut', '.', 'because', 'i', 'can', 'only', 'imagine', 'how', 'much', 'better', 'it', 'is', 'going', 'to', 'be', '...', 'and', 'yes', '...', 'there', 'are', 'lots', 'of', 'twists', '...', 'as', 'in', 'the', 'first', 'one', '...', 'including', 'the', 'ending', '!']

```

FIGURE A.7 – conversion en minuscule des tokens d'un document

```

In [88]: 1 sentence = "this is a compounded-token, this is another compounded/token and this is yet another compounded-token"
          2
          3 print(remove_punctuation(word_tokenize(sentence)))

['this', 'is', 'a', 'compoundedtoken', 'this', 'is', 'another', 'compoundedtoken', 'and', 'this', 'is', 'yet', 'another', 'compoundedtoken']

```

FIGURE A.8 – suppression des caractères de ponctuation, avant le découpage de tokens composés

```

In [90]: 1 sentence = "this is a compounded-token, this is another compounded/token and this is yet another compounded-token"
          2
          3 print(split_on_characterset(word_tokenize(sentence), r'[\~_-]'))

['this', 'is', 'a', 'compounded', 'token', '...', 'this', 'is', 'another', 'compounded', 'token', 'and', 'this', 'is', 'yet another', 'compounded', 'token']

```

FIGURE A.9 – découpage des tokens composés



Before replacing digits with letters:  
=====

This movie is like Happiness meets Lost in Translation with a Sixth Sense ending (or maybe a Crying Game surprise), and the best soundtrack I've probably ever heard...if that all make sense.<br /><br />The first 30 seconds pretty much tells you you're in for a twisted ride. (I was surprised no one walked out right away during the Brooklyn premiere.) But from there, the film settles down into a talk-fest between two really damaged people, Daphne and Buddy.<br /><br />They're lonely, mess-up, and boy do they talk about sex. Daphne brings to life her most interesting tales of escorting, some are quite funny (Mr. Chang) some disturbing (the Harlan scenes with music that tells us what we see might now be what's going on, or what Daphne is really feeling), and because I have a friend who used to escort, I might add, most seem quite real. <br /><br />You Are Alone is multi-layered and mostly brilliant. Okay, maybe a couple minutes less of the talking, and I don't know that we'd have missed anything. <br /><br />Then again, I need to see it again knowing the ending.<br /><br />I like this movie.<br /><br />(The director asked people in the Brooklyn audience to write a review on IMDb because a lot of people read them. Request granted.)

Before replacing digits with letters:  
=====

[ 'this', 'movie', 'is', 'like', 'happiness', 'meets', 'lost', 'in', 'translation', 'with', 'a', 'sixth', 'sense', 'ending', '(', 'or', 'maybe', 'a', 'crying', 'game', 'surprise', ')', ' ', 'and', 'the', 'best', 'soundtrack', 'i', 'have', 'probably', 'ever', 'heard', 'if', 'that', 'all', 'make', 'sense', ' ', 'the', 'first', 'thirty', 'seconds', 'pretty', 'much', 'tells', 'you', 'you', 'are', 'in', 'for', 'a', 'twisted', 'ride', ' ', ' ', ' ', 'i', 'was', 'surprised', 'no', 'one', 'walked', 'out', 'right', 'away', 'during', 'the', 'brooklyn', 'premiere', ' ', ' ', 'but', 'from', 'there', 'the', 'film', 'settles', 'down', 'into', 'a', 'talk', 'fest', 'between', 'two', 'really', 'damaged', 'people', ' ', ' ', 'daphne', 'and', 'buddy', ' ', 'they', 'are', 'lonely', ' ', ' ', 'mess', 'up', ' ', ' ', 'and', 'boy', 'do', 'they', 'talk', 'about', 'sex', ' ', ' ', 'daphne', 'brings', 'to', 'life', 'her', 'most', 'interesting', 'tales', 'of', 'escorting', ' ', ' ', 'some', 'are', 'quite', 'funny', ' ', ' ', 'mr.', 'chang', ' ', ' ', 'some', 'disturbing', ' ', ' ', 'the', 'harlan', 'scenes', 'with', 'music', 'that', 'tells', 'us', 'what', 'we', 'see', 'might', 'now', 'be', 'what', 'is', 'going', 'on', ' ', ' ', 'or', 'what', 'daphne', 'is', 'really', 'feeling', ' ', ' ', ' ', 'and', 'because', 'i', 'have', 'a', 'friend', 'who', 'used', 'to', 'escort', ' ', ' ', 'i', 'might', 'add', ' ', ' ', 'most', 'seem', 'quite', 'real', ' ', ' ', 'you', 'are', 'alone', 'is', 'multi', 'layered', 'and', 'mostly', 'brilliant', ' ', ' ', 'okay', ' ', ' ', 'maybe', 'a', 'couple', 'minutes', 'less', 'of', 'the', 'talking', ' ', ' ', 'and', 'i', 'do', 'not', 'know', 'that', 'we', 'would', 'have', 'missed', 'anything', ' ', ' ', 'then', 'again', ' ', ' ', ' ', 'i', 'need', 'to', 'see', 'it', 'again', 'knowing', 'the', 'ending', ' ', ' ', ' ', 'i', 'like', 'this', 'movie', ' ', ' ', ' ', 'the', 'director', 'asked', 'people', 'in', 'the', 'brooklyn', 'audience', 'to', 'write', 'a', 'review', 'on', 'imdb', 'because', 'a', 'lot', 'of', 'people', 'read', 'them', ' ', ' ', 'request', 'granted', ' ', ' ', ' ' ]

FIGURE A.10 – remplacement des tokens désignant des chiffres par leurs équivalents en lettres

Before removing punctuation:  
=====

This movie is like Happiness meets Lost in Translation with a Sixth Sense ending (or maybe a Crying Game surprise), and the best soundtrack I've probably ever heard...if that all make sense.<br /><br />The first 30 seconds pretty much tells you you're in for a twisted ride. (I was surprised no one walked out right away during the Brooklyn premiere.) But from there, the film settles down into a talk-fest between two really damaged people, Daphne and Buddy.<br /><br />They're lonely, mess-up, and boy do they talk about sex. Daphne brings to life her most interesting tales of escorting, some are quite funny (Mr. Chang) some disturbing (the Harlan scenes with music that tells us what we see might now be what's going on, or what Daphne is really feeling), and because I have a friend who used to escort, I might add, most seem quite real. <br /><br />You Are Alone is multi-layered and mostly brilliant. Okay, maybe a couple minutes less of the talking, and I don't know that we'd have missed anything. <br /><br />Then again, I need to see it again knowing the ending.<br /><br />I like this movie.<br /><br />(The director asked people in the Brooklyn audience to write a review on IMDb because a lot of people read them. Request granted.)

After removing punctuation:  
=====

[ 'this', 'movie', 'is', 'like', 'happiness', 'meets', 'lost', 'in', 'translation', 'with', 'a', 'sixth', 'sense', 'ending', 'or', 'maybe', 'a', 'crying', 'game', 'surprise', 'and', 'the', 'best', 'soundtrack', 'i', 'have', 'probably', 'ever', 'heard', 'if', 'that', 'all', 'make', 'sense', 'the', 'first', 'thirty', 'seconds', 'pretty', 'much', 'tells', 'you', 'you', 'are', 'in', 'for', 'a', 'twisted', 'ride', 'i', 'was', 'surprised', 'no', 'one', 'walked', 'out', 'right', 'away', 'during', 'the', 'brooklyn', 'premiere', 'but', 'from', 'there', 'the', 'film', 'settles', 'down', 'into', 'a', 'talk', 'fest', 'between', 'two', 'really', 'damaged', 'people', 'daphne', 'and', 'buddy', 'they', 'are', 'lonely', 'mess', 'up', 'and', 'boy', 'do', 'they', 'talk', 'about', 'sex', 'daphne', 'brings', 'to', 'life', 'her', 'most', 'interesting', 'tales', 'of', 'escorting', 'some', 'are', 'quite', 'funny', 'mr', 'chang', 'some', 'disturbing', 'the', 'harlan', 'scenes', 'with', 'music', 'that', 'tells', 'us', 'what', 'we', 'see', 'might', 'now', 'be', 'what', 'is', 'going', 'on', 'or', 'what', 'daphne', 'is', 'really', 'feeling', 'and', 'because', 'i', 'have', 'a', 'friend', 'who', 'used', 'to', 'escort', 'i', 'might', 'add', 'most', 'seem', 'quite', 'real', 'you', 'are', 'alone', 'is', 'multi', 'layered', 'and', 'mostly', 'brilliant', 'okay', 'maybe', 'a', 'couple', 'minutes', 'less', 'of', 'the', 'talking', 'and', 'i', 'do', 'not', 'know', 'that', 'we', 'would', 'have', 'missed', 'anything', 'then', 'again', 'i', 'need', 'to', 'see', 'it', 'again', 'knowing', 'the', 'ending', 'i', 'like', 'this', 'movie', 'the', 'director', 'asked', 'people', 'in', 'the', 'brooklyn', 'audience', 'to', 'write', 'a', 'review', 'on', 'imdb', 'because', 'a', 'lot', 'of', 'people', 'read', 'them', 'request', 'granted' ]

FIGURE A.11 – suppression des caractères de ponctuation

Before removing stopwords:  
=====

This movie is like Happiness meets Lost in Translation with a Sixth Sense ending (or maybe a Crying Game surprise), and the best soundtrack I've probably ever heard...if that all make sense.<br /><br />The first 30 seconds pretty much tells you you're in for a twisted ride. (I was surprised no one walked out right away during the Brooklyn premiere.) But from there, the film settles down into a talk-fest between two really damaged people, Daphne and Buddy.<br /><br />They're lonely, mess-up, and boy do they talk about sex. Daphne brings to life her most interesting tales of escorting, some are quite funny (Mr. Chang) some disturbing (the Harlan scenes with music that tells us what we see might now be what's going on, or what Daphne is really feeling), and because I have a friend who used to escort, I might add, most seem quite real. <br /><br />You Are Alone is multi-layered and mostly brilliant. Okay, maybe a couple minutes less of the talking, and I don't know that we'd have missed anything. <br /><br />Then again, I need to see it again knowing the ending.<br /><br />I like this movie.<br /><br />(The director asked people in the Brooklyn audience to write a review on IMDb because a lot of people read them. Request granted.)

After removing stopwords:  
=====

['like', 'happiness', 'meets', 'lost', 'translation', 'sixth', 'sense', 'ending', 'maybe', 'crying', 'game', 'surprise', 'best', 'soundtrack', 'probably', 'ever', 'heard', 'make', 'sense', 'first', 'thirty', 'seconds', 'pretty', 'much', 'tells', 'twisted', 'ride', 'surprised', 'no', 'one', 'walked', 'right', 'away', 'brooklyn', 'premiere', 'settles', 'talk', 'fest', 'two', 'really', 'damaged', 'people', 'daphne', 'buddy', 'lonely', 'mess', 'boy', 'talk', 'sex', 'daphne', 'brings', 'life', 'interesting', 'tales', 'escorting', 'quite', 'funny', 'mr', 'chang', 'disturbing', 'harlan', 'music', 'tells', 'us', 'see', 'might', 'going', 'daphne', 'really', 'feeling', 'friend', 'used', 'escort', 'might', 'add', 'seem', 'quite', 'real', 'alone', 'multi', 'layered', 'mostly', 'brilliant', 'okay', 'maybe', 'couple', 'minutes', 'less', 'talking', 'not', 'know', 'would', 'missed', 'anything', 'need', 'see', 'knowing', 'ending', 'like', 'asked', 'people', 'brooklyn', 'audience', 'write', 'review', 'imdb', 'lot', 'people', 'read', 'request', 'granted']

FIGURE A.12 – suppression des stopwords

Sentence: I love this food  
POS TAGGING : [('I', 'PRP'), ('love', 'VBP'), ('this', 'DT'), ('food', 'NN')]  
Lemmatization without POS TAGGING nor normalization: ['I', 'love', 'this', 'food']  
Lemmatization after normalization: ['love', 'food']

Sentence: I'm in love with eating this food  
POS TAGGING : [('I', 'PRP'), ('m', 'VBP'), ('in', 'IN'), ('love', 'NN'), ('with', 'IN'), ('eating', 'VBG'), ('this', 'DT'), ('food', 'NN')]  
Lemmatization without POS TAGGING nor normalization: ['I', 'm', 'in', 'love', 'with', 'eating', 'this', 'food']  
Lemmatization after normalization: ['love', 'eat', 'food']

Sentence: I will eat this food lovingly  
POS TAGGING : [('I', 'PRP'), ('will', 'MD'), ('eat', 'VB'), ('this', 'DT'), ('food', 'NN'), ('lovingly', 'RB')]  
Lemmatization without POS TAGGING nor normalization: ['I', 'will', 'eat', 'this', 'food', 'lovingly']  
Lemmatization after normalization: ['eat', 'food', 'lovingly']

Sentence: I find this food lovable  
POS TAGGING : [('I', 'PRP'), ('find', 'VBP'), ('this', 'DT'), ('food', 'NN'), ('lovable', 'JJ')]  
Lemmatization without POS TAGGING nor normalization: ['I', 'find', 'this', 'food', 'lovable']  
Lemmatization after normalization: ['find', 'food', 'lovable']

FIGURE A.13 – lemmatization en utilisant le POS-Tagging

## Annexe B

### Snapshots des évaluations des classifieurs

```
LogisticRegression classifier prediction started at 2019-04-19 11:14:06.394955  
  
Time taken to complete prediction: 74.41712045669556 seconds  
  
Accuracy: 0.8995714285714286  
Confusion Matrix  
[[3110  373]  
 [ 330 3187]]  
  
Classification Report  
      precision    recall  f1-score   support  
  
   -1         0.90      0.89      0.90       3483  
    1         0.90      0.91      0.90       3517  
  
 micro avg       0.90      0.90      0.90      7000  
 macro avg       0.90      0.90      0.90      7000  
weighted avg       0.90      0.90      0.90      7000
```

FIGURE B.1 – Résultats et évaluations de Logistic Regression sur le jeu de test lors de l'apprentissage

```

LogisticRegression classifier prediction of project dataset started at 2019-04-19 11:53:10.198264
Time taken to complete prediction: 40.0956335067749 seconds

Accuracy: 0.895
Confusion Matrix
[[1770  230]
 [ 190 1810]]

Classification Report
      precision    recall  f1-score   support

-1         0.90      0.89      0.89       2000
 1         0.89      0.91      0.90       2000

 micro avg       0.90      0.90      0.90      4000
 macro avg       0.90      0.90      0.89      4000
weighted avg       0.90      0.90      0.89      4000

```

FIGURE B.2 – Résultats et évaluations de Logistic Regression sur le dataset de test du projet

```

LogisticRegression classifier prediction of IMDB dataset started at 2019-04-19 11:57:38.117498
Time taken to complete prediction: 105.42721581459045 seconds

Accuracy: 0.8505
Confusion Matrix
[[4107  893]
 [ 602 4398]]

Classification Report
      precision    recall  f1-score   support

-1         0.87      0.82      0.85       5000
 1         0.83      0.88      0.85       5000

 micro avg       0.85      0.85      0.85      10000
 macro avg       0.85      0.85      0.85      10000
weighted avg       0.85      0.85      0.85      10000

```

FIGURE B.3 – Résultats et évaluations de Logistic Regression sur le dataset IMDB



```

GaussianNB classifieur prediction started at 2019-04-19 11:25:33.893449
.....

Time taken to complete prediction: 73.48156118392944 seconds

Accuracy: 0.8272857142857143
Confusion Matrix
[[2814  669]
 [ 540 2977]]

Classification Report
              precision    recall  f1-score   support

     -1         0.84         0.81         0.82         3483
     1         0.82         0.85         0.83         3517

   micro avg         0.83         0.83         0.83         7000
   macro avg         0.83         0.83         0.83         7000
weighted avg         0.83         0.83         0.83         7000

```

FIGURE B.4 – Résultats et évaluations de Gaussian Naive Bayes sur le jeu de test lors de l'apprentissage

```

GaussianNB classifieur prediction of project dataset started at 2019-04-19 11:53:50.302724
.....

Time taken to complete prediction: 42.45877385139465 seconds

Accuracy: 0.844
Confusion Matrix
[[1666  334]
 [ 290 1710]]

Classification Report
              precision    recall  f1-score   support

     -1         0.85         0.83         0.84         2000
     1         0.84         0.85         0.85         2000

   micro avg         0.84         0.84         0.84         4000
   macro avg         0.84         0.84         0.84         4000
weighted avg         0.84         0.84         0.84         4000

```

FIGURE B.5 – Résultats et évaluations de Gaussian Naive Bayes sur le dataset de test du projet

```

GaussianNB classifier prediction of IMDB dataset started at 2019-04-19 11:59:23.560473
Time taken to complete prediction: 106.52276062965393 seconds

Accuracy: 0.772
Confusion Matrix
[[3634 1366]
 [ 914 4086]]

Classification Report

```

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| -1           | 0.80      | 0.73   | 0.76     | 5000    |
| 1            | 0.75      | 0.82   | 0.78     | 5000    |
| micro avg    | 0.77      | 0.77   | 0.77     | 10000   |
| macro avg    | 0.77      | 0.77   | 0.77     | 10000   |
| weighted avg | 0.77      | 0.77   | 0.77     | 10000   |

FIGURE B.6 – Résultats et évaluations de Gaussian Naive Bayes sur le dataset IMDB