IFT6285 Devoir 2 - Fouille de données Steam

Francis de Ladurantaye
Département d'informatique et de recherche opérationnelle (DIRO)
Université de Montréal
Montréal, QC, H1V 2A2
francis.de.ladurantaye@umontreal.ca

Lien vers le projet

1 Description de la tâche

Dans le cadre de ce devoir, il nous était demandé d'extraire de l'information utile d'un corpus tiré de Steam, une plateforme de jeux vidéos accessible sur le web. Le corpus en question est composé de 59309 revues de jeux vidéos écrites par des usagers de la plateforme, dont la grande majorité sont en langue anglaise.

Peu de contraintes ayant été imposées, nous étions libres d'analyser le corpus à notre convenance afin de choisir ce que nous désirions en extraire. De plus, le corpus ayant été distribué en format json, les revues étaient accompagnées d'informations supplémentaires que nous pouvions ou non utiliser pour l'accomplissement de la tâche. L'extraction d'information pouvait être faite de façon manuelle, semi-automatique ou automatique.

2 Analyse du corpus

L'analyse du corpus est une étape importante en traitement automatique du langage naturel (TALN), car elle permet de relever les particularités dudit corpus. À ce titre, une analyse rapide à permis de constater la présence de nombres d'éléments récurrents des corpus tirés du web tels que la présence de smileys, de mots mal orthographiés, de *slang* ou d'entrées utilisant d'autres langues que l'anglais.

Toutefois, bien que ces éléments avaient le potentiel d'entraîner des difficultés pour l'extraction automatique d'information utile, j'ai tout de suite réalisé que ce serait une autre particularité de ce corpus qui me poserait le plus de problèmes. Cette particularité est la suivante : les sauts de ligne présents au sein des revues sur la plateforme web étaient absents au sein du corpus.

Cet élément peut sembler de moindre importance, mais le fait est que l'absence des sauts de ligne entraîne la concaténation de mots normalement sans rapport qui nuisent à l'utilisation d'outils tels que Spacy [HM17]. De plus, les mots concernés sont par la suite difficiles à séparer comme les phrases du corpus ne respectent pas les règles habituelles quand aux lettres majuscules et minuscules. Le repérage ainsi que la correction de ces problèmes ont donc nécessité un quantité considérable de temps et d'énergie au sein de ce travail.

3 Prétraitement

Comme il ne s'agit pas ici d'une tâche de classification, il a été jugé superflu d'effectuer un prétraitement en profondeur du corpus. En effet, l'objectif étant d'utiliser des patrons afin d'extraire des informations pertinentes, un prétraitement suffisant pour l'utilisation d'outils tels que Spacy suffisait.

Le prétraitement a donc principalement consisté à remplacer les URLs par un token, puis à tenter de séparer les mots regroupés par le retrait des sauts de ligne afin d'aider Spacy dans son travail.

Preprint. Work in progress.

Les guillemets ont aussi été normalisés et des espaces ont été ajoutés après certaines ponctuations lorsque celles-ci étaient collées au mot qui suivait.

4 Méthodologie

L'approche utilisée pour accomplir la tâche a été de tenter d'automatiser le plus possible le processus d'extraction. Afin d'y arriver, les patrons ont été écrit avant même d'effectuer quelque forme de prétraitement que ce soit. Ainsi, il était possible d'analyser les sorties produites par ces patrons afin de repérer rapidement les problèmes à corriger au sein du corpus. En résumé, le processus de prétraitement s'est donc déroulé de façon incrémentale.

Une fois satisfait de la qualité des résultats obtenus par les patrons, une étape de nettoyage a pris place afin de retirer la majeure partie du bruit présent au sein des sorties. On pourrait donc dire que le processus s'est déroulé de façon semi-automatique, où les étapes non automatique comprennent l'analyse des résultats des patrons suite au prétraitement et la création d'expressions régulières pour procéder au nettoyage des sorties desdits patrons.

5 Patrons recherchés

Plutôt que de tenter d'extraire un spectre élargi d'informations, j'ai plutôt choisi de chercher à produire des résultats de qualité, avec des visées plus restreintes.

Les outils principaux utilisés pour l'extraction et le filtrage des résultats ont été Spacy et les scripts en invite de commande.

5.1 Recherche des noms de jeux

La première tâche que j'ai choisi d'accomplir fut celle d'extraire les noms de jeux. Le fait qu'il s'agisse d'un corpus de revues à propos de ces jeux n'est pas étranger à ce choix. Considérant cette tâche comme pouvant être la base de futures recherches d'information concernant lesdits jeux, j'ai considéré important de laisser le plus petit nombre de jeux échapper à ma recherche. À cette fin, j'ai utilisé trois patrons distincts afin d'obtenir ma liste finale.

5.1.1 Patron - [sujet] is a game

Le premier patron consiste à repérer les expressions sujettes des verbes conjugués *is* ayant *game* pour attribut (`dep == attr` dans Spacy), tel qu'il était proposé en exemple dans l'énoncé du devoir. Voici quelques exemples d'expressions repérées par ce patron :

- Ark Survival Evolved is a game
- Dying Light is a game
- This is a game
- That game is a game [about something]

On remarque rapidement que la simple extraction du sujet du mot *is* dans les expressions précédentes produit une large quantité d'informations non pertinentes telles que des pronoms ou des expressions répétant l'attribut *game* du verbe. Afin de pallier le problème sans devoir filtrer manuellement les sorties produites, j'ai choisi d'apposer deux restrictions sur ces sorties.

D'abord, l'expression sujette ne devait être ni un pronom, ni un déterminant en termes de *part-of-speech*, qui permet d'éviter de trouver *This* ou *It* en résultat. Ensuite, l'expression trouvée ne devait pas inclure l'attribut utilisé dans le patron (un nombre non négligeable de résultats contenait le mot *game*).

Ce patron, en tenant compte des restrictions apposées, a permis de produire 1107 résultats dont 681 entrées uniques. Avec les efforts de nettoyage par expressions régulières et la mise en minuscules, nous terminons ici avec 508 entrées (contenant de nombreuses variantes orthographiques) avec entre 5 et 6% de bruit, ce qui est raisonnable.

Le nettoyage a d'abord consisté à retirer les expressions entre parenthèses et ce qui suivait les virgules, en plus des certains mots récurrents au seins des sorties tels que *review*, *overall* et *overview*.

Ont ensuite été retirées certaines chaînes de nature numérique au sein des résultats, puis ont été rejetées toutes les entrées contenant les mots *this*, *that*, *all* et *i*. Enfin, ont été ignorés les résultats de deux mots débutant par *the* et toutes celles débutant par *a*, avant de rejeter toutes celles ne contenant pas au moins quatre caractères.

5.1.2 Patron - the [compound] franchise / the [compound] series

Ce patron repère les noms de jeux dont les multiples suites ont permis de les qualifier de série/franchise. Le nom du jeux cherché est ici le *compound* des mots *franchise* ou *series*. En voici quelques exemples :

- The Counter-Strike franchise
- The Assassin's Creed series
- The TV series
- The anime series

Le patron a permis de trouver au total 310 éléments dont 174 éléments uniques. Suite à un petit nettoyage des sorties par expressions régulières et à la mise en minuscules, on termine avec 109 éléments (avec des variantes orthographiques) dont une douzaine représentant du bruit ou des éléments partiels.

Cette fois-ci, peu de nettoyage fut nécessaire. Il a bien sûr fallu retirer les entrées concernant la télévision plutôt que les jeux vidéos, ce qui a entraîné le rejet des entrées contenant les mots *game*, *sci* (pour sci-fi), *television*, *movie* et *anime*. Autrement, seules les entrées de trois caractères ou moins ont été éliminées.

5.1.3 Patron - it's like/game like [pobj]

Ce troisième patron concernant les noms de jeux cherche à repérer les références à d'autres jeux au sein des revues. Un total de 109 éléments ont été extraits par ce patron :

- It's like Team Fortress 2
- A game like Dota
- It's like a playable version of the tv show
- game like the other penumbra games

Une fois le nettoyage effectué et la case ignorée, on se retrouve avec 55 éléments dont seulement trois ou quatre sont du bruit. Le nettoyage a consisté à séparer les éléments trouvés sur les mots *or*, *and*, *with*, *but*, *like* et les virgules, puis à retirer les entrées ne contenant aucune lettre majuscule. Enfin, les entrées trop courtes (trois caractères ou moins) et celles commençant par le mot *a* ou contenant les mots *the* et *game* ont aussi été retirées.

5.1.4 Combinaison des patrons

Une fois les patrons précédents combinés, nous nous trouvons avoir un liste de 633 résultats, donc une petite fraction (5 à 6%) consiste en autre chose que des noms de jeux. Compte tenu du fait qu'avant la combinaison nous avions au total $(508 + 109 + 55) = 672$ résultats, cela implique que seulement $6,5\%$ des éléments trouvés avaient été repérés par plus d'un patron, ce qui semble justifier le choix d'utiliser plus d'un patron pour cette tâche.

5.2 Recherche des types de jeux

La seconde tâche à laquelle j'ai choisi de m'attaquer est celle de repérer les types de jeux mentionnés par les joueurs de la plateforme. Bien qu'il est possible de trouver des listes de types de jeux, aucune d'elles ne peut prétendre être exhaustive, et il me paraît intéressant de repérer les termes utilisés par les joueurs eux-mêmes pour définir les jeux auxquels ils jouent.

5.2.1 Patron - is a [compound] game

Le type `compound` de Spacy permet de repérer les qualificatifs reliés à un mot, généralement un nom commun. Compte tenu de la structure du patron, il fallait s'attendre à trouver beaucoup de bruit au sein des résultats qui en résulteraient.

Au total, ce patron a permis de repérer un total de 1267 qualificatifs de jeux, dont 487 entrées uniques suite à la mise en minuscules. Toutefois, nombres d'entrées figurant dans les résultats faisaient plutôt référence à des noms de jeux ou à des marqueurs d'enthousiasme plutôt qu'à des types de jeux. En voici quelques exemples :

- is a really fun game
- is a fallout game
- is an awsom game

D'autres entrées représentaient de leur côté un ensemble de types de jeux plutôt qu'un seul :

- is a horror/survival/adventure/action game
- is a stealth/assasin game
- is a crafting / building game

Dans un premier temps, il a donc été nécessaire de filtrer les entrées représentant des marqueurs d'enthousiasme/appréciation et de séparer les résultats combinant plusieurs types de jeux. Les mot repérés pour la démonstration d'appréciation sont les suivants : *fun*, *awesome*, *good*, *great*, *fav*,*really* et *quality*, ainsi que leurs variantes orthographiques (à l'aide d'expressions régulières).

Ensuite, les entrées faisant référence à des franchises de jeux populaires, parmi lesquelles on trouve *civ*, *cod*, *creed*, *fallout*, *portal*, *star wars*, ainsi que les termes reliés à des plateformes ou logiciels comme *steam* ou *valve* ont aussi été retirées de la liste finale.

La liste des résultats contenant aussi un large éventail de variantes orthographiques du mot *this* qui, dû à la mauvaise orthographe, n'avaient pas été filtrées automatiquement par Spacy en tant que pronom ou déterminant. Pour ne donner qu'un aperçu des variantes présentes au sein de la liste, c'est en utilisant la commande suivante qu'elles ont pu être filtrées : `grep -Evi '[td]((h|j)i?|ih?)s'`.

Enfin, les entrées trop courtes, c'est-à-dire celles de deux caractères ou moins, ont été éliminées. Le choix de deux caractères a dans ce cas-ci été déterminé par le fait que de nombreux types de jeux n'ont que trois caractères, comme c'est le cas pour *fps*, *mmo*, *rpg* ou *AAA* (triple A, jeu à grand budget).

À la suite de ces nettoyages, la liste résultante inclut un total de 360 entrées une fois la case ignorée. La quantité de bruit qu'elle contient reste plus élevée que pour les deux patrons précédents, mais reste raisonnable lorsque l'on jette un coup d'oeil aux résultats. En effet, la liste finale permet aisément de trouver un grande variété de types de jeux qui ne semble pas inconnus à nos oreilles :

- battle royale
- city building
- coop
- detective
- driving
- dungeon
- first person
- horror
- hunting
- indie
- ...

6 Conclusion

Rétroactivement, dû au temps requis pour permettre aux patrons de traverser l'ensemble du corpus, il aurait été préférable d'analyser le corpus plus en profondeur avant de commencer à y appliquer les patrons. En effet, chacun des quatre patrons présentés prenait entre 12 et 15 minutes pour passer au travers du corpus. Il était cependant possible d'afficher les patrons trouvés au fur et à mesure afin de pouvoir détecter plus rapidement les problèmes du corpus, mais cela s'est finalement avéré quelque peu inefficace.

Dans l'optique où l'on cherche toujours à obtenir des résultats de meilleure qualité, des méthodes supplémentaires auraient pu être appliquées afin d'épurer plus encore les listes obtenues. Pour ne nommer qu'un exemple, la distance d'édition aurait pu être appliquée aux entrées des listes elles-mêmes afin de déterminer quelles entrées ne sont que des variantes orthographiques de d'autres. Cela aurait permis de réduire davantage la taille des listes obtenues car celles-ci contiennent de nombreux exemples de cette situation.

Malgré tout, les résultats produits par l'application de patrons sont satisfaisant, ce qui permet d'affirmer avec grande conviction que le traitement automatique du langage naturel est une approche tout à fait viable afin d'accomplir ce genre de tâche.

References

[HM17] Matthew Honnibal and Ines Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017.