Classification de textes d'opinions : une approche mixte n-grammes et sémantique

Matthieu Vernier, Yann Mathet, François Rioult, Thierry Charnois, Stéphane Ferrari, Dominique Legallois

Classification de textes d'opinions : une approche mixte n-grammes et sémantique

Matthieu Vernier[1], Yann Mathet[1], François Rioult[1], Thierry Charnois[1], Stéphane Ferrari[1] et Dominique Legallois[2]

[1] Laboratoire GREYC, Université de Caen
Matthieu.Vernier@etu.info.unicaen.fr,
{Yann.Mathet, François.Rioult, Thierry.Charnois,
Stephane.Ferrari}@info.unicaen.fr

[2] Laboratoire CRISCO, Université de Caen,
dominique.legallois@unicaen.fr

Résumé : Cet article présente la participation de l'équipe du GREYC à DEFT'07, en détaillant les différentes approches mises en place ainsi que les résultats obtenus. Plusieurs techniques ont été mises en œuvre, notamment une approche à base de n-grammes, et une chaîne de traitement linguistique de production d'indices. L'approche de type n-grammes a bénéficié de traitements linguistiques complémentaires tels que la lemmatisation et la synonymie, et constitue à elle seule un classifieur autonome. La chaîne de traitements alimente quant à elle un classifieur supervisé en lui fournissant des indices s'appuyant en particulier sur un lexique. Enfin, un autre classifieur a pour vocation de conjuguer les résultats obtenus par les deux traitements précédents.

Mots-clés : Fouille de données, classification, n-grammes, lemmatisation, synonymie, sémantique, chaîne de traitements linguistiques, lexique, classification supervisée par règles d'association.

1 Introduction

Le laboratoire GREYC présente une équipe à DEFT pour la deuxième année consécutive. Pour cette édition 2007, sa composition ainsi que les techniques mises en œuvre sont pour partie issues de l'édition précédente, et pour partie nouvelles.

Le GREYC est impliqué à la fois dans la fouille de données et dans les techniques du TAL, notamment au sein de l'équipe DODOLA qui offre un carrefour idéal à ces axes de recherche. Parallèlement, une collaboration de longue haleine existe entre ce laboratoire d'informatique et le laboratoire de linguistique CRISCO. C'est donc avec un grand intérêt qu'une petite dizaine de chercheurs de Caen ont tenté de relever le défi.

En 2006, la technique mise en place consistait en une chaîne de traitement linguistique implémentée au sein de la plate-forme LinguaStream, produisant des indices pour chaque phrase du texte traité, et alimentant un classifieur (*via* un apprentissage supervisé).

Pour le présent défi, plusieurs voies ont été abordées parallèlement, et ont parfois fonctionné en synergie :

1) Un classifieur autonome basé sur une technique de n-grammes et adapté aux spécificités du langage naturel. Ce classifieur est capable de produire les fichiers de résultats DEFT. Cette partie que nous appellerons technique « n-grammes » dans la suite de cet exposé a mobilisé deux chercheurs.
2) Une chaîne de traitements linguistiques, mise en place au sein de la plate-forme LinguaStream, et produisant pour chaque texte un certain nombre d'indices. Cette chaîne alimente un classifieur supervisé, capable lui aussi de produire des résultats. Cette partie du projet est en quelque sorte un réinvestissement des idées de l'année passée, même si, bien sûr, son contenu effectif est totalement inédit, la tâche du présent défi étant singulièrement différente du défi précédent. Cette partie, que nous nommerons désormais « chaîne LinguaStream », a mobilisé trois chercheurs.

3) Enfin, un classifieur supervisé se basant sur les résultats croisés des deux traitements précédents a vocation à capitaliser ces derniers, le but étant bien sûr d'obtenir un score supérieur à chacun des deux scores indépendants. C'est un seul chercheur qui a géré l'intégralité de cette partie cette année.

En amont de l'élaboration de ces différentes approches, nous avons initié notre travail de réflexion par une étude « manuelle » des différents corpus. Pour rendre notre analyse plus pertinente, un premier travail a consisté à procéder à un découpage automatique des corpus en classes. Par exemple, pour le corpus 1 qui dispose de trois classes (correspondant resp. aux notes de 0 à 2), nous avons produit 3 fichiers, chacun desquels contient les seuls textes associés à une classe particulière. Nous avons implémenté ce découpage en Java, au moyen d'un parseur SAX.

C'est à l'issue d'un travail de réflexion d'une quinzaine de jours sur ces différents sous-fichiers des 4 corpus que les voies de recherche ont pu être définies. Nous présentons chacune de ces dernières dans les parties suivantes de cet article, puis aborderons une analyse comparative dans une ultime partie.

2 Un classifieur à base de n-grammes

La technique des n-grammes consiste à observer les collocations contiguës sur une fenêtre de n tokens consécutifs d'un flux, et à essayer de tirer de ces observations des régularités relatives à un aspect particulier de ce flux[1]. Par exemple, certains n-grammes seront caractéristiques de tel type de corpus car très récurrents dans ce dernier, et beaucoup plus rares ailleurs. En préambule à cette partie, nous devons annoncer clairement que notre équipe n'était pas du tout familière de ce type de technique, et qui si le principe nous en a paru pertinent, il s'agit pour nous d'un premier essai. En conséquence, le contenu de cet article qui y est relatif est vraisemblablement incomplet, et probablement quelque peu naïf.

Dans l'objectif de DEFT, le flux d'entrée est un matériau linguistique (textes écrits en français), et nous essayons de catégoriser les différents textes de ce flux selon le jugement porté par leur auteur. Pour illustrer de façon très simplifiée l'hypothèse de cette approche, nous espérons trouver des n-grammes caractéristiques d'un jugement favorable, défavorable, ou enfin, le cas échéant, neutre. Par exemple, pour des articles relatifs à des critiques de livres, et après analyse automatique des corpus d'apprentissage, nous pourrions avoir des tri-grammes caractéristiques tels que :

« une vraie catastrophe » : catégorie 0

« roman assez moyen » : catégorie 1

« très belle œuvre » : catégorie 2

Ainsi, lors de l'analyse d'un texte du corpus de test, si l'on tombe sur le tri-gramme « très belle œuvre », nous serons tentés de ranger ce texte en catégorie 2. Bien sûr, il y a un risque qu'au cours de l'analyse d'un même texte, nous trouvions des n-grammes appartement à différentes catégories, rendant le choix plus difficile. L'idée que nous mettons en œuvre pour pallier cette difficulté est de deux ordres :

- Ne retenir pour chaque catégorie que les n-grammes les plus discriminants, c'est-à-dire ceux étant le moins susceptibles d'apparaître dans des textes appartenant à d'autres catégories.

- Pondérer les n-grammes, c'est-à-dire associer à chacun un poids d'autant plus important qu'il apparaît fréquemment dans sa catégorie cible relativement aux autres catégories. Puis, lors de l'analyse d'un texte, tenir autant de comptes qu'il y a de catégories, et, pour chacune des catégories, sommer les poids[2] de tous les n-grammes (de cette catégorie) trouvés dans le texte. De la sorte, nous obtenons une note globale pour chacune des catégories, que nous pouvons mettre en balance avec les notes globales obtenues pour les autres catégories.

[1] Cf. Stubbs M. & Barth I. (2003).

[2] Il s'agit bien d'une somme et non d'un produit, car il ne s'agit pas ici à proprement parler d'un calcul de probabilité, mais d'une collecte d'indices concordants. Si par exemple pour un texte à classer nous obtenons les trois indices pondérés 10, 2 et 8 pour la catégorie 0 et les deux indices pondérés 12 et 12 pour la catégorie 1, le produit donnerait 10*2*8=160 contre 12*12=144, alors que la somme donnera 10+2+8= 20 contre 12+12=24. On voit au travers de cet exemple que le produit aurait tendance à favoriser de nombreux petits indices au détriment de gros indices moins nombreux.

2.1 Apprentissage

2.1.1 Collecte des n-grammes d'un corpus pour une catégorie

Étant donné que nous avons préalablement réalisé une application permettant de découper un corpus en autant de sous corpus qu'il y a de catégories, il nous suffit à présent de réaliser un traitement effectuant la collecte des n-grammes de n'importe quel corpus, et de l'appliquer ensuite sur chacun des sous-corpus.

Ce traitement prend donc en entrée un fichier corpus XML, et produit une instance de type NGramCorpus rendant compte de tous les n-grammes présents dans le texte, et de leur fréquence relative. Il est bien sûr paramétrable quant à la longueur des n-grammes à prendre en compte (monogrammes, bigrammes, trigrammes, etc.).

Par ailleurs, une méthode statique a été mise en place dans cette classe permettant de faire un « merge » des n-grammes de deux corpus distincts, à partir de deux de ses instances. Il sera ainsi possible d'obtenir le NGramCorpus des textes appartenant aux catégories 0 et 1 à partir de ceux des textes appartenant à la catégorie 0 et de ceux de la catégorie 1. Ceci s'avèrera pratique par la suite.

2.1.2 Création d'une collection de n-grammes discriminants

Conformément aux souhaits que nous avons formulés précédemment, un traitement ultérieur a vocation à déterminer quels sont les n-grammes discriminants d'un corpus vis-à-vis d'un autre. Cette notion de vis-à-vis est très importante pour la tâche que nous avons à réaliser. En effet, trouver des n-grammes représentant un corpus (donc, appliqué ici à un sous-corpus, à une catégorie de textes) en toute généralité (c'est-à-dire par rapport à un corpus générique) serait bien moins performant que de trouver des n-grammes opposant ce corpus à un certain autre corpus.

Ce traitement, réalisé par la classe CorpusDiscriminator prend donc en entrée deux instances de NGramCorpus, et fait ressortir les n-grammes révélateurs du premier corpus par rapport au second, selon le principe suivant :

- Considérer chaque n-gramme du premier corpus.
- Pour chacun d'entre eux, regarder s'il est présent ou non dans le second corpus
- S'il est absent du second corpus, et que son nombre d'occurrences dans le premier corpus est supérieur à un certain seuil paramétrable (par exemple réglé sur 1 pour éviter les orphelins), lui attribuer le poids INFINITY
- S'il est présent dans le second corpus, lui associer un poids égal au rapport entre sa fréquence relative dans le premier corpus et sa fréquence relative dans le second corpus. Ne garder ce n-gram que si le poids ainsi calculé et supérieur à un certain seuil paramétrable.

Prenons un exemple : le trigramme « une vraie catastrophe » apparaît 12 fois dans le premier corpus, donnant lieu à une fréquence relative de 12/13247 (ce corpus comportant 13247 trigrammes), et seulement 2 fois dans le second corpus, donnant lieu à une fréquence relative de 2/17523 (ce second corpus, plus volumineux, comporte 17523 trigrammes). Ce trigramme se verra ainsi attribué un poids égal au rapport de ces deux fréquences relatives, soit (12/13247) / (2/17523), c'est-à-dire 7.93. Cela signifie que l'on a pratiquement 8 fois plus de chances de trouver ce trigramme dans un texte du premier corpus que du second. Si cette valeur est supérieure au seuil que nous avons fixé, ce trigramme sera donc conservé comme trigramme discriminant, et son poids de 7.93 lui sera associé.

Prenons un second exemple : le trigramme « très belle œuvre » apparaît 4 fois dans le premier corpus, et jamais dans le second. Si 4 est supérieur au seuil paramétrable, nous conservons ce trigramme et lui associons le poids INFINITY (on a une infinité de chances supplémentaires de trouver ce trigramme dans le premier corpus que dans le second), valeur fixée dans la pratique non pas à l'infini, ce qui interdirait la prise en compte d'autres n-grammes, mais à 15, après une série de tests.

2.1.3 Discriminer une catégorie

Dans cette approche, pour discriminer une catégorie, nous souhaitons collecter des n-grammes révélateurs de cette catégorie par rapport à **toutes les autres catégories**. Pour ce faire, dans le cas où il y a plusieurs catégories autres (cas des corpus 1 à 3 de DEFT), nous utilisons la méthode statique « merge » de toutes ces dernières.

Nous constituons donc d'une part le NGramCorpus de la catégorie à discriminer, et d'autre part le merge des NGramCorpus de chacune des autres catégories. A partir de ces deux NGramCorpus, nous obtenons donc le CorpusDiscriminator de la catégorie à discriminer. Par exemple, si nous souhaitons obtenir le CorpusDiscriminator de la catégorie 1 d'un corpus comportant 3 catégories (0, 1 et 2), nous écrivons l'instanciation suivante :

```
discriminatorCat[1] = new CorpusDiscriminator(ngramCat[1],
NGramCorpus.merge(ngramCat[0],ngramCat[2])) ;
```

Notre programme établit automatiquement le CorpusDiscriminator de chacune des catégories. Le travail d'apprentissage est à présent terminé.

2.2 Classification : choisir une catégorie

L'apprentissage étant réalisé, nous pouvons maintenant aborder la question de l'assignation d'une catégorie à un texte d'un corpus. Il s'agit simplement de parcourir le texte en question au moyen d'une fenêtre de longueur n pour un choix via des n-grammes, et pour chacun des n-grammes ainsi constitué, interroger chacun des CorpusDiscriminator de chacune des catégories. On somme alors, le cas échéant (c'est-à-dire lorsque les valeurs sont non nulles), les poids correspondants. Nous obtenons, une fois tout le texte parcouru, autant de sommes qu'il y a de catégories (3 pour les trois premiers corpus, 2 pour le dernier), qui correspondent chacune à l'indice de confiance que l'on peut accorder à la catégorie en question pour ce texte.

Nous pouvons alors assigner comme catégorie celle obtenant la somme de poids la plus élevée, ou, comme cela était aussi possible dans DEFT, proposer un indice de confiance en pourcentage à chacune des catégories, sans statuer de façon catégorique. Dans ce cas, l'indice de confiance pour une catégorie donnée est tout simplement le rapport entre le poids total de cette catégorie sur la somme des poids totaux des autres catégories.

Par ailleurs, si certains corpus se prêtent plus à un calcul via des bigrammes, d'autres des trigrammes, etc., il est fréquent qu'une combinaison de plusieurs traitements en n-grammes, par le cumul des poids de ces derniers, soit plus performante que l'application d'un seul d'entre eux. Ainsi, notre application propose le choix de la plage de n-grammes à appliquer au corpus traité. Par exemple, la plage [2, 3] signifie que l'on cumule les traitements bigrammes et trigrammes.

2.3 Les apports de la linguistique

Les premiers tests réalisés à ce stade montrent des résultats positifs, i.e. supérieurs à un tirage aléatoire, mais leur observation précise révèle parfois un manque de n-grammes discriminatoires lors du processus d'apprentissage. En d'autres termes, lorsque parmi les n-grammes du texte à catégoriser, plusieurs sont aussi présents dans le corpus d'apprentissage, nous obtenons des poids non nuls, et la catégorisation est souvent satisfaisante. Mais lorsque ceux-ci sont trop peu nombreux dans le corpus d'apprentissage, il arrive que certaines catégories obtiennent un poids nul, ou très faible, le choix se faisant alors sur une autre catégorie, souvent mauvaise. Ce phénomène est bien sûr d'autant plus manifeste que le corpus d'apprentissage est réduit.

Or les éléments présentés jusqu'ici pourraient se prêter indifféremment à différents types de flux, pour des classifications de différentes natures, pour autant que les n-grammes soient révélateurs du phénomène étudié. Pourtant, le matériau sur lequel nous nous penchons ici est de nature linguistique, ce qui lui confère un certain nombre de spécificités et de régularités dont nous pouvons tirer parti. En effet, il est fréquent que même si les formes linguistiques de surface diffèrent, les valeurs sémantiques soient pourtant très proches. Nous allons en effet améliorer notre approche en lui appliquant quelques traitements d'ordre linguistique, à la fois lors de l'apprentissage et de l'exécution.

2.3.1 Lemmatisation des corpus

L'idée de la lemmatisation est fondée sur une observation simple mais parfois très puissante : des éléments linguistiques ayant une valeur sémantique proche, mais différant quant à leur genre, leur nombre ou leur temps morphologique, verront leurs formes lemmatisées identiques.

« est un apport », « sera un apport » → « être un apport »

« la bonne idée », « les bonnes idées » → « le bon idée »

De la sorte, en lemmatisant à la fois les corpus d'apprentissage et d'exécution, nous donnons aux n-grammes une généricité permettant de les multiplier virtuellement : un n-gramme donné d'un corpus d'apprentissage aura valeur de tous les n-grammes donnant lieu à la même forme lemmatisée.

La contre partie de ce procédé est qu'en gagnant en généricité, nous créons corollairement un appauvrissement linguistique équivalent, à savoir que « la bonne idée » et « les bonnes idées » auront la même valeur (en fait, virtuellement, seront un seul et même tri-gramme), alors que le second aurait sans doute une valeur discriminante positive plus forte. Nous aurons l'occasion de discuter de ce point lors de l'analyse des résultats.

2.3.2 La synonymie

Dans le même esprit de gagner en généricité, i.e. de multiplier virtuellement la taille des corpus d'apprentissage, nous avons étudié la possibilité de tirer parti de la synonymie. En effet, de façon un peu naïve, le fait que deux mots soient synonymes fait que l'un peut se substituer à l'autre, si bien qu'à partir d'un n-gramme donné, on peut virtuellement générer nombre de n-grammes sémantiquement équivalents.

Dans les faits, plutôt que de générer profusion de n-grammes, nous allons simplement remplacer chaque mot, le cas échéant, par son représentant sémantique (choisi arbitrairement pour une classe de synonymes donnée). Nous appliquons ceci à la fois lors de l'apprentissage et de l'exécution.

Voici une illustration de l'attribution d'un représentant sémantique à une classe de synonymes :

« `bon` » → `bon`
« `excellent` » → `bon`
« `formidable` » → `bon`
« `extraordiaire` » → `bon`

On remarque que le représentant sémantique d'une classe donnée se représente lui même (cf. première ligne de l'illustration précédente).

Dans l'optique d'avoir une généricité maximale à moindres frais, nous avons tenté le recours à une ressource linguistique informatique préexistante, le dictionnaire des synonymes du Crisco (cf. Manguin, 2005). Malheureusement, les résultats se sont généralement effondrés, ou tout du moins amenuisés. La principale raison est semble-t-il le problème de la **polysémie**. Un terme donné étant (le plus souvent) polysémique, il en résulte que parmi l'ensemble de ses synonymes vont se trouver des mots dont le sens n'aura rien à voir avec celui qui nous intéresse. Par exemple pour le trigramme « apprécié ce livre », on souhaiterait élargir sa portée à des trigrammes tels que « aimé ce livre » ou « adoré ce livre ». Mais, par le recours au dictionnaire des synonymes, nous l'élargirons aussi à un trigramme tel que « évalué ce livre » (évaluer étant l'une des acceptions possibles d'apprécier), qui pour sa part ne code aucunement une appréciation positive… Le bruit ainsi généré prend le dessus sur le gain offert par la généricité.

Nous nous sommes finalement orientés vers un lexique des synonymes établi manuellement pour la tâche particulière de DEFT, et l'avons décliné en quatre versions relatives à chacun des corpus. Ce mini dictionnaire dédié des synonymes rend compte principalement des adjectifs et des verbes d'évaluation (aimer, détester, être d'accord, etc.) ainsi que des principaux objets dont il est question (œuvre, livre, film, etc.). Il a été établi de sorte à générer le moins de bruit possible, notamment en limitant son étendue aux seuls termes non (ou faiblement) polysémiques.

Nous avons profité de ce dictionnaire pour créer une classe « ponctuation » qui est le représentant de tous les signes de ponctuation d'un texte. Les signes « . », « , », « ; », etc. ont tous pour pseudo synonyme le représentant « ponctuation », si bien que des trigrammes initiaux tels que « belle œuvre , » et « belle œuvre . » seront identiques une fois passés dans le module des synonymes.

La difficulté de la polysémie mise à part, il reste comme écueil à cette piste le fait, une fois encore, que ce que l'on gagne en généricité, on le perd en spécificité. Le bon ratio est donc à trouver, qui dépend notamment de la taille du corpus d'apprentissage. Plus ce dernier est maigre, plus le recours à la généricité sera bénéfique, et vice-versa.

2.3.3 Le traitement de la négation

Enfin, nous nous sommes attaqués à la question de la négation dans nos textes dès l'origine de notre travail tant notre analyse initiale a révélé de façon flagrante combien souvent la négation pouvait inverser le sens d'une valeur sémantique locale.

Une première idée, que nous pouvons qualifier de traitement sémantique, consistait à inverser la valeur sémantique de ce qui est porté par la négation. Nous avions alors affaire à une double difficulté, la première étant de circonscrire ce sur quoi porte la négation (ce qui nécessite une analyse lexicale suffisamment robuste), la seconde d'être capable d'inverser la valeur sémantique correspondante (soit par un dictionnaire des antonymes, mais on retombe alors sur le problème de la polysémie, soit en alimentant les autres catégories, mais lesquelles et comment ?). Nous n'avons pas eu la possibilité de mettre en œuvre cette idée faute de moyens.

Une seconde idée, moins ambitieuse, consiste à éliminer des corpus toutes les parties de proposition qui sont sous le joug d'une négation. Le traitement assez basique que nous avons réalisé consiste à ne pas considérer la partie du flux comprise entre une marque de négation « ne » ou « n' » ou encore « pas », et la prochaine ponctuation, ceci à la fois dans le corpus de test que dans les corpus d'apprentissage. En supprimant de tels segments, on évacue le fait que la valeur sémantique à prendre en compte est difficile à appréhender. Cette fausse bonne idée a été un échec sur tous les corpus, pour une double raison. La première est qu'en évacuant une partie du corpus, on limite d'autant l'apprentissage (et l'exécution) : cela diminue virtuellement la taille des corpus. La seconde est qu'en fait, sans que l'on ait finalement trop à s'en préoccuper, le n-grammes disposant d'une fenêtre suffisamment large (trigrammes, quadrigrammes…) prennent d'eux même en compte la valeur sémantique de la négation : « ai pas aimé », « pas un bon roman », etc.

2.4 Application aux différents corpus et résultats

2.4.1 Présentation

Le traitement réalisé, entièrement automatique, peut être appliqué tel quel à tous les corpus. Néanmoins, nous avons procédé à des ajustements de paramètres selon les spécificités de ces derniers. Les principaux paramètres ajustables sont les suivants :

- Plage de valeurs n des n-grammes : permet de définir sur quels n-grammes appliquer le traitement (monogrammes, bigrammes, trigrammes, etc.), les poids étant cumulés lorsque la longueur de la plage de valeurs de n est supérieure à 1.
- MIN_COUNT_FOR_INFINITY : le nombre minimum nécessaire d'occurrences pour prendre en compte les n-grammes de poids « infini », c'est-à-dire n'apparaissant que dans le corpus à différencier. Fixé par exemple à 2, on ne les gardera que s'ils apparaissent au moins 2 fois.
- ELIMINATE_VALUE : idem, mais pour des n-grammes de poids non « infini », c'est-à-dire apparaissant à la fois dans le corpus à différencier et dans le corpus de comparaison.
- MIN_QUOTIENT : valeur minimale du rapport entre le nombre d'occurrences d'un n-gramme dans le corpus à évaluer et dans le corpus de comparaison. C'est donc en fait, par construction, le poids minimum des n-grammes finalement retenus.
- LEMMATISATION : ON/OFF. Choix d'appliquer ou non la lemmatisation.
- SYNONYMIE : ON/OFF. Choix d'appliquer ou non le traitement de la synonymie.
- Coefficient correcteur à appliquer à chaque catégorie : permet d'ajuster le poids d'une catégorie par rapport à ce que donne le calcul des n-grammes. Par exemple, si l'on constate lors des tests que le rappel de la catégorie 0 est déficitaire par rapport au rappel des autres catégories, on pourra gonfler ce dernier en lui assignant un coefficient correcteur tel que 1.1 (pour 10% d'augmentation des poids).

Nous avons effectué l'essentiel de nos paramétrages en nous basant sur un découpage du corpus d'apprentissage en deux, les premiers 90% pour apprendre, et 10% restants pour tester. Faute de temps, nous n'avons pu procéder qu'à un second test en toute fin d'échéance, avec resp. les derniers 80% et les premiers 20%. Les résultats ont alors été sensiblement différents, malheureusement. Nous avons alors pris le parti d'ajuster les différents paramètres en fonction de ces deux jeux de tests (l'idéal aurait été de faire 10 séries de tests en faisant tourner les découpages 90% -10%).

Mise en garde : les commentaires des sections suivantes sont basés sur les tests 90%-10%, et non sur les corpus de test et d'apprentissage finaux (ne disposant pas de la version notée du corpus de test final, il ne nous est pas possible de faire autrement). Les indications fournies ici seraient donc sans doute un peu différentes avec les corpus réels.

2.4.2 Paramétrage des 4 corpus

Corpus	Plage de n-grammes	Min-count for infinity	Eliminate value	Min quot.	Lemmatisation	Synonymie	Coefficients correcteurs
1	[1, 3]	3	1	1.5	ON	ON	Cat0 : 0.85 Cat1 : 1.35 Cat2 : 1
2	[2, 3]	3	2	1.5	ON	ON	Cat 0 : 1.15 Cat 1 : 1 Cat 2 : 1
3	[2, 3]	3	2	1.5	ON	OFF	Cat 0 : 1.08 Cat 1 : 1.07 Cat 2 : 0.78
4	[1, 3]	3	2	1.5	OFF	ON	Cat 0 : 1 Cat 1 : 1

2.4.3 Corpus 1

L'application de lemmatisation produit un bond du F-Score d'environ 10% (+0.1), ce qui est particulièrement remarquable. L'application des synonymes donne lieu quant à elle à un gain supplémentaire d'environ 0.6% (+0.006).

2.4.4 Corpus 2

L'application de lemmatisation produit un gain du F-Score d'environ 4.5% (+0.045), et celle des synonymes donne lieu quant à elle à un gain supplémentaire d'environ 1.1% (+0.011), ce qui n'est pas négligeable compte tenu du score déjà établi. C'est sur ce corpus que nos principes ont été les plus efficaces, les scores obtenus étant nettement supérieurs aux scores obtenus sur les trois autres.

2.4.5 Corpus 3

L'application de lemmatisation produit un gain du F-Score d'environ 3% (+0.03). Par contre, sur ce corpus, les synonymes donnent lieu à un baisse d'environ 1% du score, ce qui en fait un cas particulier. Notons que nos traitements se prêtent manifestement assez mal à ce corpus, sans doute du fait de sa petite taille. Par ailleurs, il est probable que notre choix de répartition 90% initiaux – 10% finaux ne se soit pas révélé opportun ici, car les résultats sur le corpus réel chutent de plusieurs points. Il aurait donc été prudent de ne pas procéder à des ajustements basés sur un nombre de textes non significatif (10% d'un petit corpus, qui plus est divisé ensuite en 3 catégories…).

2.4.6 Corpus 4

Contre toute attente, et fait unique, ce corpus donne de moins bons résultats avec le processus de lemmatisation. Cela peut provenir selon nous de deux choses :

- la première, statistique, est due à la grande taille du corpus d'apprentissage. Le nombre de n-grammes étant « naturellement » important, le fait de lemmatiser apporte relativement moins de nouveaux n-grammes virtuels. Le bruit généré par cette lemmatisation prend alors le dessus sur le maigre gain en rappel.
- La seconde est de nature linguistique, liée à la nature même du corpus manipulé : il est probable qu'ici, à la fois le temps des verbes (notamment la différence entre passé, présent et futur), ainsi et surtout que la personne (« je » versus « nous ») aient une importance prépondérante, alors même que le processus de lemmatisation les gomme.

La synonymie permet quant à elle d'obtenir un léger gain d'environ 1%.

2.4.7 Conclusion

Nous avons donc observé des différences significatives de résultats entre les corpus, mais aussi la nécessité d'adapter les valeurs des paramètres d'ajustement d'un corpus à l'autre. Il s'avère que la taille du corpus d'apprentissage et surtout que sa nature linguistique entrent en ligne de compte pour ces différents ajustements (cf. 4. Résultats et comparaison des approches).

3 Un classifieur basé sur une chaîne de traitements linguistiques

3.1 Analyse linguistique

La deuxième méthode consiste à repérer et à exploiter un certain nombre d'indices linguistiques qui marquent la présence d'une évaluation positive ou négative dans un énoncé. Elle se base notamment sur des travaux[3] qui avaient été menés sur un thème proche et qui avaient montré la faisabilité d'une telle approche[4]. L'expertise linguistique et certaines ressources ont pu être réinvesties partiellement dans le cadre de DEFT. Toutefois, ce genre d'approche linguistique aurait nécessité une expertise propre aux corpus proposés pour laquelle nous manquions de temps et de moyens compte-tenu de l'ampleur de la tâche.

Nous détaillons par la suite quels sont les types d'indices retenus et quelle est la méthode pour pondérer la valeur évaluative de ces indices. En fin d'analyse, l'objectif est d'obtenir pour chaque corpus, un ensemble de scores qui viennent alimenter un processus de classification basé sur l'extraction automatique de motifs dans une matrice.

3.1.1 Différentes catégories d'indices

Termes évaluatifs.

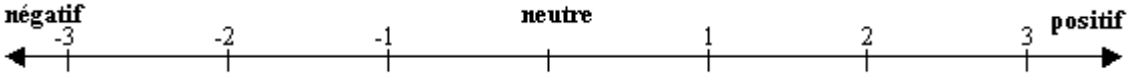
Au niveau lexical, un grand nombre de termes, quelle que soit leur catégorie grammaticale, portent une valeur évaluative intrinsèque.

	Positif	**Négatif**
Noms	*chef d'œuvre, réussite, beauté, perfection, merveille, …*	*nullité, absence, faiblesse, ersatz, navet, déception, ...*
Adjectifs	*beau, superbe, extraordinaire, intéressant, parfait, …*	*nul, incorrect, bâclé, laid, …*
Verbes	*réussir, plaire, …*	*décevoir, fruster, perdre son temps*
Adverbes	*heureusement, magnifiquement, clairement, judicieusement, …*	*malheureusement, hélas, …*

L'accumulation de ces lexies dans l'énoncé fournit autant d'indices susceptibles de mettre en lumière l'opinion de l'auteur. Toutefois, il est possible de rencontrer ces termes dans un contexte différent de l'évaluation. Pour éviter un phénomène de « bruit », la valeur donnée à ces indices est très peu élevée (1 pour les positifs, -1 pour les négatifs). Il aurait sans doute été envisageable, dès cette étape, d'opérer une gradation entre ces divers termes.

[3] Cf. Legallois D & Ferrari S. (2006).

[4] Un nombre appréciable de travaux sur le discours évaluatif a été mené récemment en linguistique anglo-saxonne. Parmi ceux-ci : Martin J. & White P. (2005), Hunston S. & Thompson G. (2000), Bednarek (2006).



mauvais, infâme	*insuffisant, indigeste …*	*léger, fade…*	*bon, joli…*	*intéressant, superbe …*	*remarquable, divin …*

Faute d'une analyse plus poussée de la concordance de ces termes avec une évaluation réelle et donc, de courir le risque d'attribuer trop d'importance à un terme, nous nous contentons de les considérer au même niveau.

Cadres expérientiels.

Les corpus 1 et 2 sont consacrés à des textes d'opinions sur des objets culturels. De ce fait, nous avons pu réexploiter certains cadres expérientiels décrits par Legallois et Ferrari (2006). Une analyse de l'évaluation d'un objet culturel est vite confrontée à un problème inhérent à la constitution de l'objet même : on peut évaluer différents aspects ou *qualia* ; par exemple, le contenu, le style, la satisfaction ou la déception par rapport à des attentes, etc. L'évaluation peut porter également sur l'auteur du livre, sur l'histoire. Autrement dit, la forme de l'expression d'un jugement est naturellement configurée par rapport à des *cadres expérientiels*. Par exemple, le cadre de l'affect :

Ex : « *Le guépard* […] nous chavire le cœur à jamais »,
« Jane pleure. Et nous aussi nous pleurons. »

Dans notre analyse, nous retenons deux cadres dont les termes relatifs dénotent l'évaluation :

- l'**emprise** des objets évalués sur le lecteur :
« l'auteur plonge le lecteur dans la mythologie »,
« Alain Fleischer se met en marche pour envoûter le lecteur pendant plus de quatre cents pages » ,
« *Les associés* […] n'emporte jamais réellement l'adhésion du spectateur », …
« Il vampirise notre intérêt par sa bonhomie bienveillante et son côté bougon sympathique. »
- les **attentes** satisfaites ou non du lecteur :
« On reste sur notre faim »
« On peut regretter le classicisme du choix des auteurs »
« On déplorera par contre le saucissonnage artificiel et purement commercial de la série »
« […] font de ce film une agréable surprise estivale

Ces indices sont également annotés par un score propre qui, selon les cas, est égal à 1 ou –1.

3.1.2 Différents « poids » d'indices

Objets du domaine.

L'observation des différents corpus nous permet de constater une certaine régularité quant aux objets sur lesquels sont portés une évaluation. Il est possible de prévoir les termes désignant ces objets. L'hypothèse est alors de se dire : « Lorsqu'un terme évaluatif ou expérientiel concerne directement un terme du domaine, il est beaucoup plus probable que l'on soit en présence d'une évaluation réelle ».

Ex : « un article *intéressant* »
« un *beau* film »
« le roman nous *entraîne* dans l'intimité d'une famille bourgeoise »

Ce type d'indices nous semble plus convaincant que la simple présence des mots « intéressant », « beau », « entraîne », etc. qui peuvent intervenir hors-contexte évaluatif.

De plus, nous catégorisons deux types d'objets du domaine : les termes qui désignent **un objet général** particulièrement important, et ceux uniquement relatifs à **une partie de l'objet**. Nous considérons, dans une critique de film par exemple, que critiquer « un acteur » a moins d'impact sur l'opinion générale du texte que s'il s'agissait d'une critique portant sur l'objet « film ».

	Termes généraux (coef. 4)	**Termes partiels (coef. 2)**
Corpus 1	*film, roman, album, livre, spectacle, divertissement, comédie, …*	*personnage, histoire, scénario, acteur, dialogue, musique, décor, …*
Corpus 2	*jeu, titre, version, opus, …*	*niveau, mode, son, gameplay, univers, graphisme, prise en main, …*
Corpus 3	*article, papier, rapport, contribution, travail, étude, recherche, …*	*résultats, approche, méthode, outil, application, expérience, …*
Corpus 4	hypothèse non considérée sur ce corpus.	

Modificateurs d'intensité.

Nous avons vu précédemment que certains adverbes pouvaient être intrinsèquement évaluatifs (*judicieusement, clairement,* …) ; une autre caractéristique d'un certain nombre d'adverbes est de permettre à un auteur de moduler l'opinion qu'il souhaite faire partager.

Ex : « Un film particulièrement réussi »
« Un papier véritablement intéressant »

Ce rôle est également tenu par certains adjectifs, comme dans :
« un pur bonheur »
« un véritable échec »

La présence de ces modificateurs d'intensité associée à un indice évaluatif lexical augmente (ou diminue si l'indice initial est négatif) le poids de l'indice selon un coefficient de 2. Ainsi :

« un pur bonheur » : pur → coefficient (intensité) : 2
bonheur → évaluation intrinsèque : 1
score de l'indice → 2

Combiné avec la règle précédente, il est possible d'obtenir :

« un papier véritablement mauvais » : papier → coefficient (terme général du domaine) : 4
véritablement → coefficient (intensité) : 2
mauvais → évaluation intrinsèque : -1
score de l'indice → -8

Marques de négation.

Selon le même principe, nous tentons de tenir compte des tournures négatives pour inverser la valeur de l'indice repéré (coefficient –1).

« des personnages sans réelle saveur » : personnage → coefficient (terme partiel du domaine) : 2
sans → coefficient (négation) : -1
réel → coefficient (intensité) : 2
saveur → évaluation intrinsèque : 1
score de l'indice → -4

« approche ne me semble guère probante » : approche → coefficient (terme partiel du domaine) : 2
[ne … guère] → coefficient (négation) : -1
probant → évaluation intrinsèque : 1
score de l'indice → -2

Toutefois un certain nombre de tournures négatives ne sont pas considérées. En particulier, lorsque la marque de négation ne se situe pas à proximité de l'indice.

Marques de concession.

Après analyse du corpus 3, il nous a semblé intéressant d'envisager la pondération des indices inclus dans les tournures de phrases concessives.

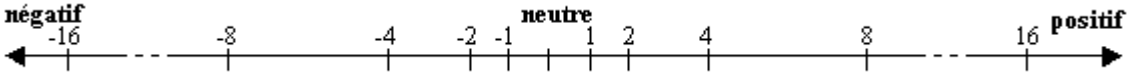
Ex : « la section 4 […] est intéressante mais ne propose aucune solution »

Dans cet exemple, l'adjectif « intéressant » laisse suggérer une évaluation positive, or l'aspect évaluatif de cette proposition est ambigu voire légèrement négatif. Dès lors, pour éviter de donner de

l'importance à des indices qui ne sont que les prémices d'une tournure concessive, nous annulons le score produit par ceux-ci. Cette hypothèse n'est testée que sur le corpus 3 pour lequel les tournures concessives sont relativement fréquentes et témoignent, a priori, de la nature des textes : des soumissions peuvent difficilement être complètement en dehors des attentes du relecteur, et une soumission peut toujours être améliorée.

Séquences lexico-grammaticales.

La combinaison de ces séquences lexico-grammaticales permet d'envisager une échelle d'indices résumée par la figure suivante :



score	indice	score	indice
-16	« un film véritablement très mauvais », …	+1	« poétique », « pas mal », …
-8	« un roman très ennuyeux », …	+2	« jolis effets spéciaux », « pas complètement inintéressant», …
-4	« l'introduction est particulièrement imprécise », …	+4	« beau film », « casting particulièrement réussi », …
-2	« approche guère probante », …	+8	« une BD très drôle », …
-1	« nul », «aucun intérêt »	+16	« cet album est une véritable grande réussite » …
0	« l'article est intéressant mais […] »		

3.1.3 Calcul du score à l'échelle du texte et à l'échelle de parties de texte

L'objectif de cette analyse est de produire, pour tous les textes d'un corpus, un score positif et un score négatif sur l'ensemble de l'énoncé en sommant les scores des indices trouvés. Du point de vue de l'analyse du discours, il nous a semblé cohérent de préciser également des scores propres à certaines parties du discours qui peuvent marquer plus fortement l'évaluation ou ayant des chances de refléter au mieux l'opinion associée à l'énoncé. En général, le premier et le dernier paragraphe (« introduction » et « conclusion ») ont ainsi un score qu'il peut être intéressant de préciser indépendamment du score général. L'hypothèse émise est que l'auteur aura tendance à annoncer la couleur de son opinion dès les premiers instants de l'énoncé, et qu'il pourra éventuellement synthétiser ses arguments en fin de texte. Cependant, les parties considérées sont variables selon les corpus. Après l'analyse préalable des différents corpus, nous avons constaté une récurrence de sections particulières sur un bon nombre de textes d'un corpus.

Ainsi, le **corpus 2** contient une partie sous-titrée par « Note Générale : ». Une telle section est susceptible de bien résumer la teneur des propos de l'auteur et est assimilable à une conclusion.

Certains textes du **corpus 3**, précisent explicitement par un sous-titre, l'objet d'une critique : « Commentaire », « Originalité », « Référence », « Importance », « Exactitude », « Rédaction ». D'emblée, l'analyse par classe de ce corpus nous montre qu'une critique portée sur la rédaction d'un article n'est absolument pas révélatrice de l'acceptation ou non de celui-ci. C'est pourquoi nous considérons comme une partie, l'ensemble des rubriques à l'exception de celle concernant la rédaction. Par ailleurs, la partie « Commentaire » semble faire l'objet d'une critique d'aspect général. Nous examinons donc indépendamment les indices contenus dans cette dernière.

Dans le **corpus 1**, le premier paragraphe représente le titre de l'œuvre. Nous considérons donc que l'introduction est constituée par l'union des deux premiers paragraphes. Les critiques étant assez longues, la conclusion est figurée arbitrairement par les deux derniers paragraphes.

Le **corpus 4** étant très souvent constitué de textes courts (un seul paragraphe), nous déterminons là aussi de façon arbitraire, que les deux premières phrases figurent l'introduction, et les deux dernières la conclusion.

Ainsi, nous obtenons différents scores d'indices positifs et négatifs pour chaque texte. Avec l'apport de la méthode d'extractions de motifs fréquents dans une matrice, nous espérons à ce stade que ces différents scores permettront une amélioration des résultats.

3.2 Mise en œuvre des traitements linguistiques

Nous utilisons *LinguaStream*, une plate-forme dédiée au TAL qui permet, dans une même chaîne de traitements, d'utiliser différents formalismes déclaratifs afin de marquer des objets linguistiques. Nous nous appuyons ici en particulier sur une grammaire Prolog (composant DCG Marker) et sur des expressions régulières. Afin de minimiser le besoin en ressources et le temps de calcul inhérent à ce genre d'analyse automatique, un traitement a préalablement scindé les fichiers XML des différents corpus à notre disposition. En début de chaîne, nous disposons d'un fichier XML pour chaque texte.

La chaîne de traitements de la figure 1 montre les différents composants utilisés pour notre analyse. Cette chaîne est exécutée automatiquement pour chaque texte. Après une segmentation en mots et une catégorisation grammaticale, quatre types d'expressions régulières enrichissent le fichier XML de base afin de marquer certaines parties du texte (cf. 3.1.3).

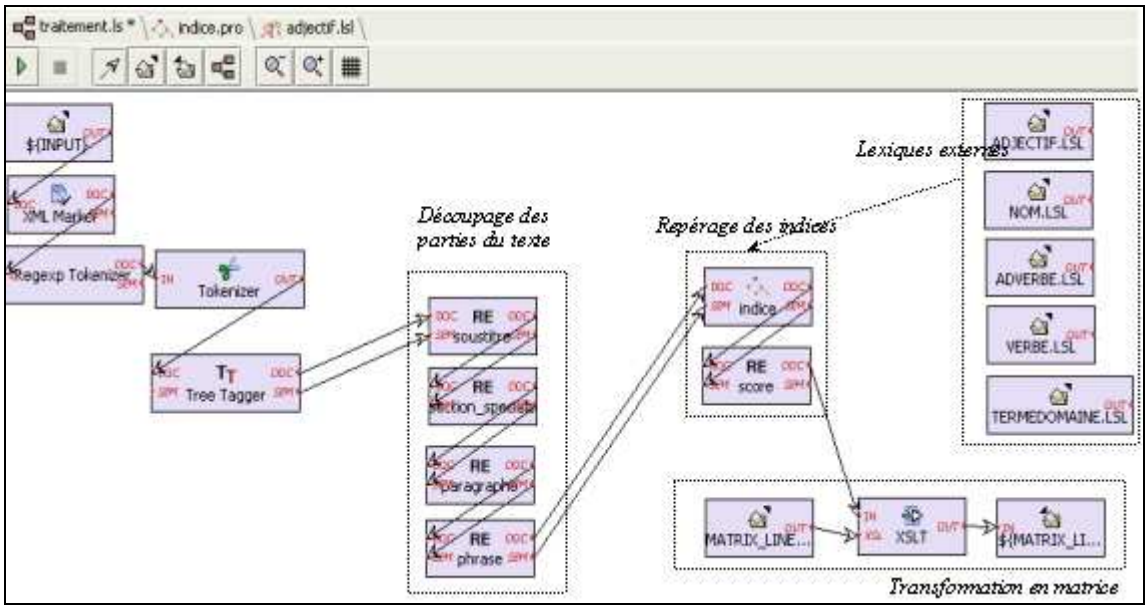


Fig. 1 – chaîne de traitements dans LinguaStream

Le composant « central » de notre approche est élaboré à partir du composant DCG Marker de LinguaStream. Il s'agit d'une grammaire Prolog qui opère un balisage des indices selon les règles décrites précédemment. Cette grammaire utilise des ressources lexicales externes constituées des formes lemmatisées des termes qui nous sont utiles. Pour chacun de ces termes, il est possible d'associer certains attributs qui correspondent aux caractéristiques propres que nous souhaitons leur donner.

En fin de chaîne, un dernier composant applique une transformation XSL au fichier XML enrichi par les marquages successifs. Les règles de transformation permettent de comptabiliser et regrouper les différents scores de l'énoncé. Au final, la chaîne produit un fichier texte constitué d'une ligne tabulée résumant l'évaluation des différentes parties du discours qu'il a nous semblé bon de considérer. Un post-traitement permet de reconstituer une matrice complète à partir de toutes les lignes produites.

3.3 Un classifieur supervisé

Le savoir-faire du GREYC en fouille de données concerne l'utilisation de méthodes à base de motifs ensemblistes. Les bases de données recensant des objets décrits par des attributs booléens, un motif est une conjonction d'attributs. Nous disposons depuis une dizaine d'années d'algorithmes performants pour

extraire les motifs fréquents (les motifs présents dans un nombre minimum d'objets) et construire les règles d'association. De la forme $X \rightarrow Y$, ces règles sont mesurées par une fréquence et une confiance, qui indiquent le nombre d'objets contenant $X \cup Y$ et la probabilité conditionnelle d'apparition de *Y* connaissant celle de *X*. Lorsque ces règles concluent sur un attribut de classe, elles peuvent être utilisées pour construire un classifieur automatique.

Plusieurs méthodes existent pour effectuer de la classification supervisée à partir de règles associations. Historiquement, la première et la plus simple est CBA (cf. Liu et *al.*, 1998)(Classification Based on Association). Cette méthode extrait les règles d'association de fréquence et confiance minimales désirées par l'utilisateur, et ordonne ces règles suivant leur confiance. Lorsqu'un nouvel exemple se présente, la première règle qui peut s'appliquer propose une valeur de classe.

Ce procédé a été raffiné par la méthode CMAR (cf. Li et *al.*, 2001) (Classification based on Multiple class-Association Rules) qui ne se contente plus d'une seule règle pour prendre la décision de classification. Les règles sont cette fois-ci mesurées par un indice de corrélation fourni par un χ^2 normalisé. On évite également la redondance entre les règles en ne conservant que celles qui sont à prémisse minimale. Un nouvel exemple sera classé à l'issue d'un vote réalisé par toutes les règles qui s'appliquent, selon leur pondération.

3.3.1 Notre méthode

Pour nos expériences, nous avons implémenté une méthode proche de CMAR, mais qui utilise des règles *disjonctives* ou *généralisées*. Contrairement aux règles d'association classiques, les règles généralisées sont de la forme $X \rightarrow \vee Y$ et concluent sur une disjonction d'attributs plutôt que sur une conjonction. Il est ainsi possible d'obtenir des règles *positives* (qui, en concluant sur un attribut de classe, entérinent la possibilité que l'exemple à classer appartienne à cette classe, si elle coïncide avec la prémisse) et des règles négatives (qui, en concluant sur la négation d'un attribut de classe, excluent la possibilité de classe correspondante) (cf. Antonie et Zaïane, 2004). Les règles positives sont de la forme $X \rightarrow c \vee Y$ (*c* est un attribut de classe) et se reformulent $X \overline{Y} \rightarrow c$. Les règles négatives de la forme $X c \rightarrow \vee Y$ et se reformulent $X \overline{Y} \rightarrow \overline{c}$. Dans les deux cas, elles s'appliquent pour classer tout exemple qui contient le motif *X*, mais aucun des attributs de *Y*.

Selon le modèle de CMAR, les règles sont pondérées par une mesure de χ^2. Pour un nouvel exemple, les règles positives voient leur pondération s'ajouter au score, les règles négatives soustraient leur pondération. Au final, la classe avec le meilleur score est désignée.

3.3.2 Aménagements pour les données du défi

La répartition des classes sur les différents corpus est très hétérogène. Dans cette configuration, les méthodes de classification à base d'association sont peu efficaces, car les classes dominantes fournissent plus de règles que les autres. Nous avons donc réalisé l'apprentissage sur un échantillon équilibré de chaque corpus.

D'autre part, nous avons limité la conclusion des règles à un singleton pour les corpus 1 et 2, car cela fournissait le meilleur résultat. En revanche, dans le corpus 3 qui contient peu d'objets, nous n'avons trouvé que peu de règles : nous avons alors dû extraire des règles généralisées dont la conclusion comportait jusqu'à trois attributs. Pour le très fourni corpus 4, le problème inverse s'est posé : le temps de calcul nécessaire à la constitution d'un classifieur fiable était insurmontable et nous avons renoncé à proposer une solution avec cette méthode.

4 Résultats et comparaison des approches

4.1 Tableau des résultats

Une analyse rapide des résultats obtenus (cf. Fig . 2) montre que la méthode « n-grammes » est la plus efficace sur les corpus 1, 2 et 4. De plus, les résultats sur le corpus 2 sont sensiblement meilleurs que les autres.

L'origine des différences dans les résultats nous semble variée : elle peut être liée à la nature des corpus fournis ou aux méthodes choisies. Certaines caractéristiques propres à chacun des corpus peuvent influer différemment selon les cas.

	« n-grammes »	*« analyse linguistique »*	*«approche combinée »*
Corpus 1	F-Score : 0,577 Préc. : 0,583 | Rappel : 0,571	F-Score : 0,457 Préc. : 0,444 | Rappel : 0,472	F-Score : 0,532 Préc. : 0,533 | Rappel : 0,532
Corpus 2	F-Score : 0,761 Préc. : 0,782 | Rappel : 0,741	F-Score : 0,506 Préc. : 0,493 | Rappel : 0,520	F-Score : 0,715 Préc. : 0,705 | Rappel : 0,726
Corpus 3	F-Score : 0,414 Préc. : 0,414 | Rappel : 0,414	F-Score : 0,474 Préc. : 0,476 | Rappel : 0,472	
Corpus 4	F-Score : 0,673 Préc. : 0,676 | Rappel : 0,669		

Fig. 2 – Tableau des résultats obtenus pour les différentes approches

4.2 Nature des corpus.

Corpus 1. La variété des objets critiqués aurait demandé une expertise humaine plus poussée pour bien considérer les usages linguistiques propres à ces différents genres d'énoncés[5]. On peut donc penser que les résultats de l'approche « linguistique » sont améliorables. Cette diversité conjuguée à la relative grande taille du corpus, permettant un entraînement, rend la méthode par « n-grammes » plus intéressante.

Corpus 2. Les tests de jeux vidéos ciblent un public précis, et le niveau de langue moins soutenu implique une variété lexicale et syntaxique plus réduite que pour les autres corpus. De plus, la présence de paragraphes récurrents présentant les différentes parties du jeu (graphismes, jouabilité, …) contribue à donner à ces textes un aspect « formulaire ». Par ces contraintes, le rédacteur de la critique est guidé à exprimer précisément les différentes facettes de son opinion, voire à réitérer son jugement. Ces considérations sont particulièrement rentabilisées dans le cas de l'utilisation des « n-grammes ».

Corpus 3. Un début de classification manuelle réalisée en amont a montré la difficulté, pour un humain, de déterminer la classe d'un texte. Une soumission peut avoir reçu une bonne critique mais ne pas être acceptée, ou inversement un article moyen peut tout de même être accepté. La variabilité du taux de sélection d'articles à une conférence nous semble être un paramètre important à prendre en compte ici. De ce fait, l'automatisation de cette tâche nous a paru dès le départ comporter une difficulté relativisant les faibles résultats pour ce corpus. La taille restreinte de ce corpus est un facteur qui peut expliquer le score plus faible de l'approche par « n-grammes » pour laquelle un entraînement est nécessaire.

Corpus 4. Il est à noter que certains textes exprimant une opinion sur un amendement à une loi et non directement sur cette dernière, le résultat enregistré peut être contraire à celui attendu, même par une expertise humaine. Il en résulte des résultats dans l'absolu un peu inférieurs à la réelle efficacité du traitement. Néanmoins, la méthode n-grammes réalise un score tout à fait intéressant, tirant bénéfice semble-t-il de la très grande taille du corpus. La méthode Linguastream n'a quant à elle pas pu être appliquée pour les raisons évoquées précédemment.

4.3 Nature des méthodes.

Une des propriétés importantes qui différencient les deux approches tient dans leur façon de discriminer les classes de textes et en particulier la classe intermédiaire (classe « 1 ») : *l'approche « n-grammes »* est capable de discriminer les trois classes de texte à partir des n-grammes révélateurs d'une classe par rapport aux deux autres classes. *l'approche « linguistique »*, quant à elle, cherche à dégager une forte présomption d'opinion négative ou positive à partir d'indices. La classe intermédiaire est, par conséquent, plus difficile à discriminer. La question est de savoir comment établir des indices concrets de

[5] Les travaux de Legallois D. & Ferrari F. (2006) s'intéressent essentiellement à l'évaluation des critiques de livres. Certaines hypothèses retenues ont été testées sur les autres objets de façon « aveugle ».

« neutralité » ou de savoir à quel moment il n'y a pas suffisamment d'indices majoritairement présent, et donc par défaut de considérer le texte comme étant de classe 1.

Par ailleurs, la différence dans la nature de ces méthodes nous permet d'envisager leur enrichissement mutuel. La constitution des lexiques de l'approche « linguistique » peut ainsi être améliorée par les n-grammes révélateurs d'une classe. Réciproquement, une expertise linguistique permettrait une évaluation de certains n-grammes et d'évacuer ceux qui paraissent non pertinents d'un point de vue sémantique.

Enfin, deux améliorations peuvent être envisagées. D'une part, la prise en compte de la catégorie syntaxique du n-gramme (syntagme nominal ou verbal) pourrait améliorer les résultats s'il s'avérait qu'une catégorie était plus pertinente qu'une autre. D'autre part, il serait intéressant d'étudier les effets d'une lemmatisation ciblée (par exemple ne lemmatiser que les verbes, que les noms, etc.) afin de voir l'impact qu'elle a sur chacune des catégories grammaticales sur les différents corpus.

5 Conclusion

L'équipe du GREYC a pu mettre à l'œuvre des compétences multiples pour relever ce défi, menant notamment à la mise en place d'un traitement à base de n-grammes, d'une chaîne de traitements linguistiques « LinguaStream », ainsi que d'un classifieur supervisé.

A l'issue des tests finaux, des différences notables apparaissent entre les approches. En l'état (nos travaux respectifs n'ont pu être menés que sur 2 mois), on constate que l'approche par n-grammes est celle qui produit les meilleurs résultats lorsque les corpus d'apprentissage sont suffisamment fournis, soit sur trois des quatre corpus du défi. Corollairement, l'approche plus directement sémantique est d'autant plus intéressante que l'apprentissage est effectué sur un corpus réduit, ce que l'on constate sur le corpus 3.

La combinaison des deux approches qui a été menée jusqu'à présent s'est contentée de s'appuyer sur les résultats pris indépendamment de ces deux dernières. Nous constatons un résultat dégradé par rapport à la meilleure méthode, l'autre méthode apportant plus de bruits que d'indices pertinents. Une combinaison des approches en amont de la classification supervisée nous paraît judicieuse ; elle pourra suivre les quelques idées présentées dans la partie 4.

Références

ANTONIE M.-L., ZAÏANE O. (2004), *An Associative Classifier based on Positive and Negative Rules*, ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'04), Paris, France.

BEDNAREK M. (2006), *Evaluation in Media Discourse*, Continuum.

HUNSTON S., THOMPSON G. (eds) (2000), *Evaluation in Text. Authorial Stance and the Construction of Discourse*, Oxford University Press.

LEGALLOIS D., FERRARI S. (2006), *Vers une grammaire de l'évaluation des objets culturels*, Schedae, prépublication n°8, fascicule n°1, pages 57-68.

LI W., HAN J., PEI J. (2001), *CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules*, IEEE International Conference on Data Mining (ICDM'01), San Jose, USA.

LIU B., HSU W., MA Y. (1998), *Integrating classification and association rules mining*, International Conference on Knowledge Discovery and Data Mining (KDD'98), New York, USA, pages 80-86.

MANGUIN J.L. (2005) *La dictionnairique Internet : l'exemple du dictionnaire des synonymes du CRISCO*, CORELA – Cognition, Représentation, Langage, Numéro spécial.

MARTIN J., WHITE P. (2005), *The Language of Evaluation: Appraisal in English*, Palgrave Macmillan Hardcover.

STUBBS M., BARTH I. (2003), *using recurrent phrases as text-type discriminators : a quantative method and some findings* in Functions of language 10 :1, 61-104.

WIDLÖCHER A., BILHAUT F. (2005), *La plate-forme LinguaStream : un outil d'exploration linguistique sur corpus*, In Actes de TALN 2005, Dourdan, France, pp. 517-522.