Rapport Projet: Approche Contenu pour recommendation de cours Ã Polytechnique

Claude Demers-Belanger (1534217) & Mikael Perreault (1741869)

November 30, 2018

Introduction

Dans la cadre du cours LOG6308, nous avons conçu un système de recommandation de cours pour la Polytechnique de Montréal qui prend comme entrée un cours qui aurait été apprécié par l'utilisateur et lui retourne un certain nombre de cours recommandés. Nous avons, pour l'instant, fixé le nombre de recommandations à 5 cours.

Motivation

Plusieurs raisons nous ont poussé à réaliser ce projet.

- Créer un outil de comparaison et de recommandation de cours. En ce moment, les outils de recherche de cours sont limités. Ils consistent uniquement en des outils de recherche par mots clés.
- Obtenir un moyen de recommander des cours hors cursus. Présentement, les étudiants ont facilement accès à la liste de cours de leur cursus, mais parfois, certains étudiants pourraient vouloir suivre des cours hors cursus, plus centrés sur leurs intérêts.
- En faisant ce projet, nous avons mis en place les premières étapes pour le design d'un système de recommandations de cours, mais nous avons aussi imaginé quelles seraient les étapes suivantes pour avoir un système complet et efficace.
- Apprendre les rudiments de l'analyse textuelle. Ces concepts ont été vaguement mentionnés dans le cours, mais nous avions un désir de pousser nos connaissances davantage pour se développer une intuition.

Dataset

Le set de données original est composé des descriptions de tous les cours de U de M, UQAM, HEC et Polytechnique. Chaque ficher .txt de description comprend le titre du cours et sa description. Nous avons remarqué, lors de nos analyse, certaines irrégularités dans les descriptions en ce qui attrait au traitement des accents. Certains mots avaient des accents retirés simplement, d'autres avaient la lettre complètement retirée.

Pour notre projet, nous avons décidé de conserver seulement les cours de polytechnique pour limiter le temps de calcul. Aussi, nous avons retiré tous les cours dont la description était de moins de 20 mots. Ce qui équivalait à 2 écarts types sous la moyenne du nombre de mots par description. Pour plus de détails sur la moyenne et l'écart type du nombre de mot, voir la section Analyse Exploratoire des Données.

De plus, lors de la création de la matrice termes - documents (termes - cours dans notre cas) un stemming a été réalisé pour retirer les marques de pluriels et de feminin. Cependant, le stemming ne semblait pas régulier,ceci probablement dû au fait que, dans les descriptions originales, certains mots avaient été coupé de façon irrégulière. Nous n'avons pas tenu compte des irrégularités dans nos calculs. Dans un travail futur, une analyse en détails pourrait être faite sur ce sujet.

En résumé, nous utiliserons pour nos calculs la matrice termes (ligne) cours (colonne) de polytechnique sans les cours qui ont été retiré en raison des descriptions trop courtes. De plus, pour simplifier la matrice, nous avons retiré les mots qui ne figuraient dans aucune description de Polytechnique.

Dans la section suivante, nous ferons un exploration de la matrice pour tenter de mieux comprendre le dataset.

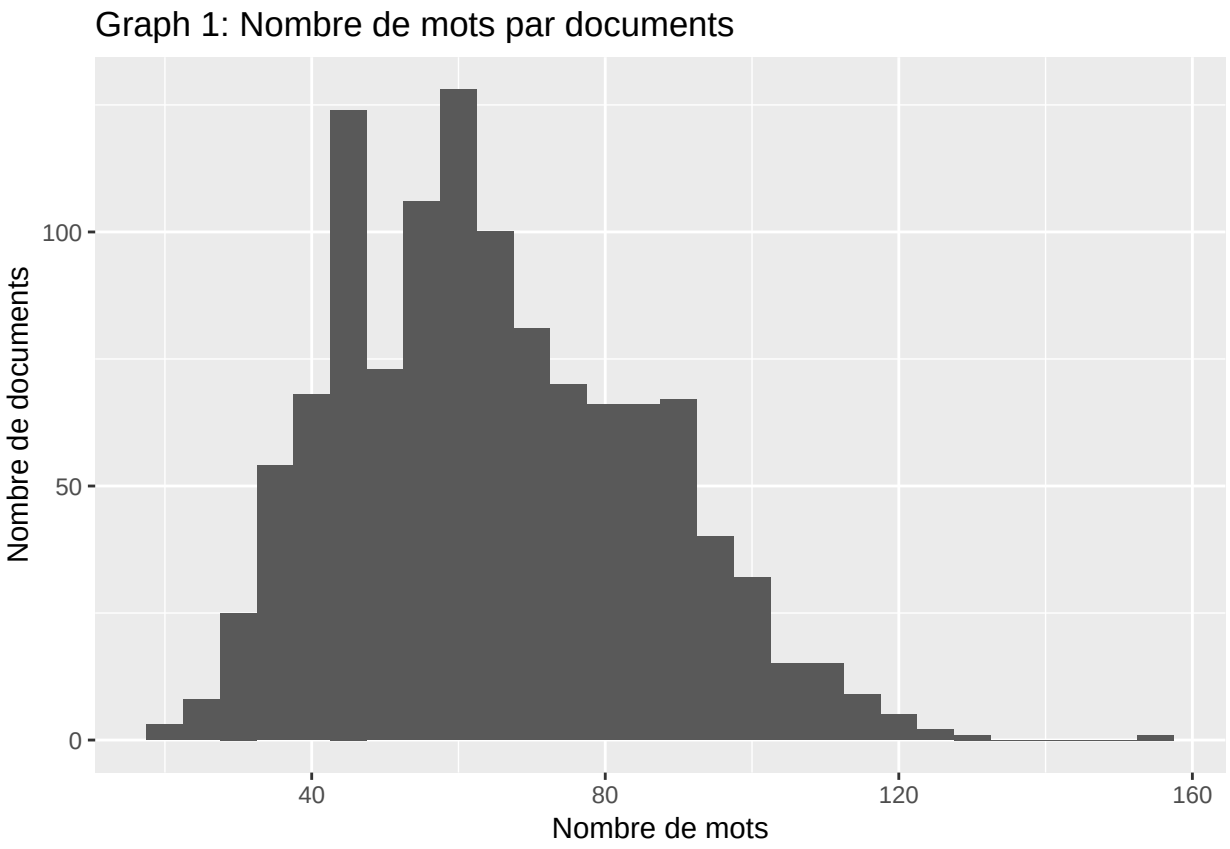
Analyse Exploratoire des Données

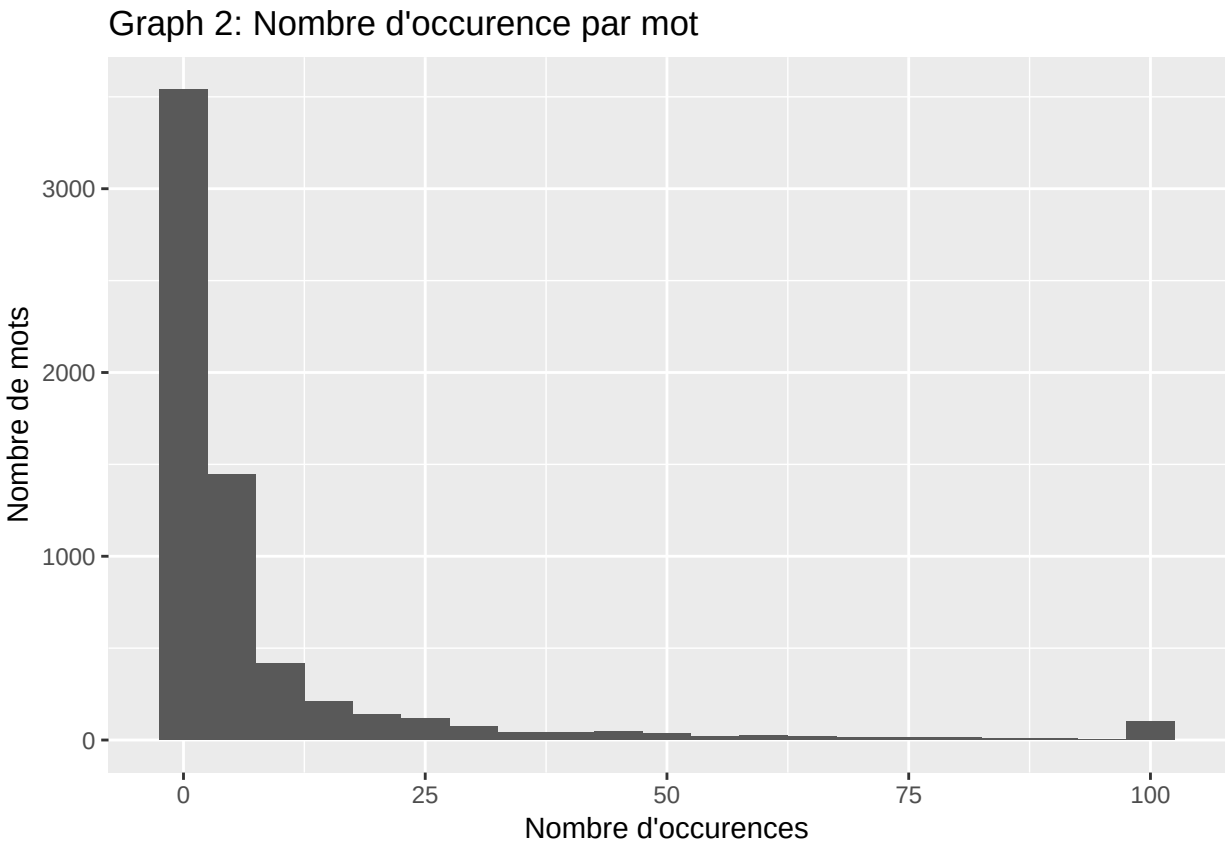
Trouvons le terme qui revient le plus souvent et la moyenne de fois qu'il revient par document. C'est le terme **"de"** qui revient le plus souvent dans les documents. Il revient 11332 fois et en moyenne **9.7020548** par document. En réduisant la matrice termes-documents originale avec seulement les cours de Polytechnique, on conserve **30%** des termes et **7%** des documents.

On analyse ensuite la matrice terme document des cours de poly et on trouve qu'en moyenne les cours ont une descriptions de **64.422089 mots** et un écart type de **21.029194 mots**. Sachant ceci, nous avons donc décidé de conserver seulement les cours qui ont plus de (Moyenne - 2 écart type) mots, donc tous les cours de plus de 20 mots environ.

Ceci retire seulement **9 cours** de notre matrice. Nous aurons donc, au total, **1159** cours avec en moyenne **64.7963762 mots**.

Regardons deux graphiques. Le premier est un histogramme du nombre de mots par documents etle second, un histogramme du nombre d'occurrences par mot.





Pour mieux voir les occurrences par mot, nous avons ajouté aux mots qui revenaient 100 fois tous les mots qui avaient plus de 100 occurences (**94 mots**).Ceci a été fait pour améliorer l'affichage des histogrammes. Voici la liste des mots qui reviennent le plus fréquemment et le nombre de fois qu'ils apparaissent;

Table 1: Les mots les plus populaire

	n
de	11308
un	1873
null	1159
titrecour	1159
descriptioncour	1159
projet	739
dan	571
system	549
analys	502
concept	501
aux	425
travail	392
applic	376
method	365
stage	365
etudi	337
techniqu	330
gestion	310
commun	299

	n
programm	262

Cadre Théorique

Pour le design de notre solution, nous nous sommes initialement basés sur l'article "Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications" par Achakulvisut et autres.[1] Dans l'article, on mentionne plusieurs transformations de matrice qui seront ensuite utilisées dans une méthode "Latent Semantic Analysis" (LSA). Ces transformations et la méthode LSA seront traitées lors des trois sections suivantes.

TF-IDF

La transformation TF-IDF de la matrice termes-cours permet de mettre l'emphase sur des termes plus rares par rapport aux termes qui se retrouvent dans pratiquement tous les documents.

L'équation suivante montre comment nous avons calculé le TF-IDF:

$$TF - IDF_{i,j} = (1 + log f_{i,j}) * log \frac{n}{f_i + 1}$$

où

$TF\text{-}IDF_{i,j}$ = TF-IDF du terme i pour le document j
$f_{i,j}$ = Fréquence du terme i dans le document j
f_i = Nombre de document contenant le terme i
n = Nombre total de documents

On utilise la transformation du TF avec le log en se basant sur l'article scientifique mentionné plus haut.

Log-Entropy

La transformation suivante que nous avons utilisée est la transformation log-entropy. On doit d'abord calculer l'entropie global du terme i (g_i) à l'aide de la formule suivante:

$$g_i = 1 + \sum_j \frac{p_{i,j} * log_2(p_{i,j})}{log_2(n)}$$

où

$$p_{i,j} = \frac{f_{i,j}}{\sum_j f_{i,j}}$$

On calcule ensuite l'entropie du terme i pour le document j ($l_{i,j}$):

$$l_{i,j} = log_2(1 + log f_{i,j}) * g_i$$

Sachant que:
$f_{i,j}$ = Fréquence du terme i dans le document j
n = Nombre total de documents
g_i = Entropie global pour le terme i
$l_{i,j}$ = Entropie du terme i pour le document j

[1] Achakulvisut T, Acuna DE, Ruangrong T, Kording K (2016) Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications. PLoS ONE 11(7): e0158423. doi:10.1371/journal.pone.0158423

LSA

1) Pourquoi utiliser LSA?

Puisque le but de notre projet est de présenter une méthode qui pousse plus loin la compréhension des descriptions de document qu'une méthode classique termes-termes, nous avons opté pour un traitement LSA (Latent Semantic Analysis). Les raisons de l'utilisation de cette technique mathématique puissante sont nombreuses et en voici quelques unes :

- En mettant l'emphase sur les dimensions latentes de notre matrice termes-documents, LSA nous permet de dégager le contexte des mots. Il s'agit d'une technique bidirectionnelle en ce sens qu'elle permet de dégager les contextes les plus vraisemblables pour un mot donné ainsi que les mots les plus vraisemblables pour un contexte donné.
- Cette technique nous permet de réduire l'effet du problème des mots à double sens. Par exemple, si on se base sur la co-occurences des termes, le mot "contrainte" peut autant produire des recommandations de cours de matériaux (contrainte mécanique) que des cours d'informatique (programmation par contraintes). La compréhension du contexte nous permet d'amoindrir ce problème.
- Cette méthode fonctionne par elle-même: on a besoin d'aucune métadonnée, aucun dictionnaire ni aucun graphe sémantique ni syntaxique.
- Finalement, on émet l'hypothèse que les recommandations LSA seront caractérisées par une sérendipité et une robustesse accrue en raison de l'abstraction des concepts découlant de la nature même de la méthode.

2) Les étapes de LSA

En somme, LSA est une combinaison de méthodes que nous avons déjà vues dans le cours. Voici les principales étapes:

1) Générer la matrice termes-documents et l'homogénéiser (ex: word stemming).
2) Pondération des termes selon TF-IDF comme expliqué précédemment.
3) Décomposition SVD (Singular Value Decomposition: On décompose la matrice termes-documents en trois matrices, soit W, S et P. W est la représentation vectorielle des valeurs orthogonales et factorisées des lignes de la matrice originale alors que P est l'analogue pour les colonnes. S est une matrice diagonale contenant des valeurs d'échelle qui font en sorte que, lorsque ces trois matrices sont multipliées ensemble, le produit est égal à la matrice originale. Plus précisément :

$$X = WSP'$$

4) Réduction de dimension et recommandations : Afin de faire ressortir les facteurs latents, on doit procéder à une réduction de dimensions. De cette manière, lorsque les matrices réduites seront multipliées ensemble, le produit sera égal à une approximation de la matrice originale au sens des moindres carrés. Avec les matrices réduites $\hat{W}$, $\hat{S}$, $\hat{P}'$, on a donc:

$$\hat{X} = \hat{W}\hat{S}\hat{P}'$$

Avec la matrice $\hat{X}$, on peut maintenant procéder aux recommandations en générant des corrélations par rapport aux colonnes de cette matrice (pour dégager les cours similaires)

Modèles

Logique derrière les modèles

La logique des modèles a été basée en partie sur l'article mentionné dans la partie cadre théorique [2]. Nous sommes partis de la matrice termes-documents pour créer des recommandations de base, ensuite nous avons appliqué des transformations (TF-IDF et Log-Entropy) à cette matrice pour générer de nouvelles recommandations. Par la suite, nous avons utilisé ces trois matrices comme entrées pour mettre dans une fonction LSA pour générer 3 nouvelles listes de recommandations.

Ces recommandations prennent comme entrée un cours qu'un étudiant aurait apprécié. Ensuite, on effectue un calcul de corrélation de Spearman pour trouver les cours les plus similaires dans la matrice termes-documents ou une des matrices modifiées. Pour faciliter le sondage, nous avons décidé de générer seulement 5 recommandations par méthode (voir section Résultats pour plus d'informatiosn sur le sondage).

Présentation des modèles

1) Comparaison des termes

Pour le modèle de comparaison terme-terme, nous faisons simplement une comparaison des cours entre eux avec une corrélation de spearman. Ensuite, on détermine les 5 cours les plus similaires pour chacun des cours. Avec l'entrée d'un cours, on choisit donc la ligne correspondante à ce cours. Nous discuterons plus en détails des recommandations dans la section suivante. Cependant, voici l'exemple du code qui générera les recommandations pour le cours de thermodynamique en comparant terme à terme sans transformation de la matrice originale.

```
correlation.termes=cor(t(as.matrix(m.poly)),method="spearman")
correlation.termes[is.na(correlation.termes)]=-1
neighbors.termes <-t(apply(correlation.termes,1,max.nindex.corr))
id.cours1 <- which(colnames(m.poly)=="MEC1210") #Thermodynamique
recommandations.terme <- colnames(m.poly)[neighbors[id.cours1,]]
recommandations.terme
```

```
## [1] "PHS1104" "GCH1510" "M-252"   "MEC3215" "MTR2211"
```

2) TF-IDF

Pour le modèle TF-IDF, on utilise la même logique mais en utilisant une matrice modifiée TF-IDF pour faire les corrélations. Voici le code pour générer la matrice et une partie de cette matrice:

```
log.m <- log(m.poly)
log.m[is.infinite(log.m)] <- 0
tf.idf <- (1 + log.m) * log (n.courses/(rowSums(m.poly > 0)+1))
tf.idf[m.poly==0]=0

tf.idf[1:20,1:8]
```

```
## 20 x 8 Matrix of class "dgeMatrix"
##                      AE3100        AE3205        AE3300        AE3400
## null         -0.0008624408 -0.0008624408 -0.0008624408 -0.0008624408
## huileus       0.0000000000  0.0000000000  0.0000000000  0.0000000000
## four          0.0000000000  0.0000000000  0.0000000000  0.0000000000
## prefix        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## polytechniqu  0.0000000000  0.0000000000  0.0000000000  0.0000000000
## travaux       0.0000000000  0.0000000000  0.0000000000  0.0000000000
```

[2] Achakulvisut T, Acuna DE, Ruangrong T, Kording K (2016) Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications. PLoS ONE 11(7): e0158423. doi:10.1371/journal.pone.0158423

```
## criter        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## fourni        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## rglage        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## dela          0.0000000000  0.0000000000  0.0000000000  0.0000000000
## bioconcentr   0.0000000000  0.0000000000  0.0000000000  0.0000000000
## econo         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## hough         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## regional      0.0000000000  0.0000000000  0.0000000000  0.0000000000
## lumir         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## gouvern       0.0000000000  0.0000000000  0.0000000000  5.2635533741
## tribologi     0.0000000000  0.0000000000  0.0000000000  0.0000000000
## perfectionn   0.0000000000  0.0000000000  0.0000000000  0.0000000000
## hitchcock     0.0000000000  0.0000000000  0.0000000000  0.0000000000
## fractal       0.0000000000  0.0000000000  0.0000000000  0.0000000000
##                      AE3900       AE3900A       AE3900B        AE4000
## null         -0.0008624408 -0.0008624408 -0.0008624408 -0.0008624408
## huileus       0.0000000000  0.0000000000  0.0000000000  0.0000000000
## four          0.0000000000  0.0000000000  0.0000000000  0.0000000000
## prefix        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## polytechniqu  0.0000000000  0.0000000000  0.0000000000  0.0000000000
## travaux       0.0000000000  0.0000000000  0.0000000000  0.0000000000
## criter        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## fourni        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## rglage        0.0000000000  0.0000000000  0.0000000000  0.0000000000
## dela          0.0000000000  0.0000000000  0.0000000000  0.0000000000
## bioconcentr   0.0000000000  0.0000000000  0.0000000000  0.0000000000
## econo         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## hough         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## regional      0.0000000000  0.0000000000  0.0000000000  0.0000000000
## lumir         0.0000000000  0.0000000000  0.0000000000  0.0000000000
## gouvern       0.0000000000  0.0000000000  0.0000000000  0.0000000000
## tribologi     0.0000000000  0.0000000000  0.0000000000  0.0000000000
## perfectionn   0.0000000000  0.0000000000  0.0000000000  0.0000000000
## hitchcock     0.0000000000  0.0000000000  0.0000000000  0.0000000000
## fractal       0.0000000000  0.0000000000  0.0000000000  0.0000000000
```

3) Log-Entropy

Log-Entropy reprend la même logique que le modèle précédent mais avec une transformation log-entropy. Voici le code pour générer la matrice et une partie de cette matrice:

```
pij <- m.poly / rowSums(m.poly)
log2.pij <- log2(pij)
log2.pij[is.infinite(log2.pij)] <- 0
global.entropy <- 1 + rowSums((pij * log2.pij * pij)/log2(n.courses))
log.entropy <- log2(1 + m.poly) * global.entropy

log.entropy[1:20,1:8]
```

```
## 20 x 8 Matrix of class "dgeMatrix"
##                AE3100    AE3205    AE3300    AE3400    AE3900   AE3900A
## null         0.9991372 0.9991372 0.9991372 0.9991372 0.9991372 0.9991372
## huileus      0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## four         0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## prefix       0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
```

```
## polytechniqu 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## travaux      0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## criter       0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## fourni       0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## rglage       0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## dela         0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## bioconcentr  0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## econo        0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## hough        0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## regional     0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## lumir        0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## gouvern      0.0000000 0.0000000 0.0000000 0.9543766 0.0000000 0.0000000
## tribologi    0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## perfectionn  0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## hitchcock    0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## fractal      0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##                AE3900B    AE4000
## null         0.9991372 0.9991372
## huileus      0.0000000 0.0000000
## four         0.0000000 0.0000000
## prefix       0.0000000 0.0000000
## polytechniqu 0.0000000 0.0000000
## travaux      0.0000000 0.0000000
## criter       0.0000000 0.0000000
## fourni       0.0000000 0.0000000
## rglage       0.0000000 0.0000000
## dela         0.0000000 0.0000000
## bioconcentr  0.0000000 0.0000000
## econo        0.0000000 0.0000000
## hough        0.0000000 0.0000000
## regional     0.0000000 0.0000000
## lumir        0.0000000 0.0000000
## gouvern      0.0000000 0.0000000
## tribologi    0.0000000 0.0000000
## perfectionn  0.0000000 0.0000000
## hitchcock    0.0000000 0.0000000
## fractal      0.0000000 0.0000000
```

4) LSA

Pour les modèles LSA, nous changerons simplement les matrices d'entrée par les matrices avec les différentes transformations. Nous utiliseront le package LSA de R. On utilise la fonction LSA() qui décompose en facteurs latents et la fonction as.textmatrix() qui prend les facteurs latents en entrée et retourne une matrice qui sera utilisée pour faire les corrélations.

Voici le code pour LSA avec la matrice d'entrée terme-document originale qui gènère une matrice post LSA:

```
X.lsa <- as.textmatrix(lsa(m.poly,dims=50))
X.lsa[1:20,1:8]
```

```
##                    AE3100        AE3205       AE3300        AE3400
## null          8.165613e-01  0.9287668724  0.986030457  0.8439828336
## huileus       7.343004e-04 -0.0103724012 -0.007725750  0.0224131656
## four         -3.755100e-03 -0.0029196718 -0.002621446 -0.0020240824
## prefix       -4.273663e-04  0.0028437920  0.002896917 -0.0022376627
## polytechniqu -1.649466e-02 -0.0197239211 -0.029527996 -0.0787595842
```

```
## travaux       4.987936e-02  0.0853848650  0.142204012  0.0644985774
## criter        9.000380e-03 -0.0192875711 -0.003297627  0.0013498924
## fourni       -1.026616e-03  0.0062022060  0.006928182 -0.0006575164
## rglage        1.038961e-02 -0.0009100416 -0.007168425  0.0674207875
## dela          7.921415e-04 -0.0016020271 -0.007491581 -0.0024856266
## bioconcentr  -9.336277e-04 -0.0046784288 -0.002832542  0.0016010661
## econo         7.184972e-04  0.0022944834  0.002350895 -0.0003780666
## hough         1.516151e-03 -0.0029529066 -0.004133759 -0.0049473828
## regional      8.389577e-04 -0.0032714773 -0.002318561  0.0020664138
## lumir         5.813205e-05  0.0064089031  0.005164021 -0.0042104236
## gouvern       9.743515e-04  0.0279891269  0.014676646  0.0686851358
## tribologi     2.803766e-03  0.0034244822  0.006259324  0.0054547969
## perfectionn  -3.527035e-03  0.0139264436  0.031909745  0.0129594558
## hitchcock    -6.782678e-04 -0.0024623710  0.002836799  0.0027777667
## fractal       1.257841e-02 -0.0085897748 -0.010503865 -0.0074096552
##                     AE3900       AE3900A       AE3900B        AE4000
## null          1.0472770647  1.0472770647  1.0472770647  1.0062718151
## huileus      -0.0305468944 -0.0305468944 -0.0305468944  0.0018315706
## four          0.0002977427  0.0002977427  0.0002977427 -0.0037040491
## prefix        0.0038005404  0.0038005404  0.0038005404  0.0039550879
## polytechniqu  0.0287669558  0.0287669558  0.0287669558  0.0422716383
## travaux       0.1561116184  0.1561116184  0.1561116184  0.1813529451
## criter       -0.0392656742 -0.0392656742 -0.0392656742  0.0428618645
## fourni       -0.0108240426 -0.0108240426 -0.0108240426 -0.0014671222
## rglage       -0.0039388961 -0.0039388961 -0.0039388961  0.0362903968
## dela          0.0000999339  0.0000999339  0.0000999339 -0.0015213596
## bioconcentr  -0.0036717822 -0.0036717822 -0.0036717822  0.0026578213
## econo         0.0007325618  0.0007325618  0.0007325618  0.0003922691
## hough        -0.0061644256 -0.0061644256 -0.0061644256  0.0068513891
## regional     -0.0007854185 -0.0007854185 -0.0007854185  0.0074061616
## lumir         0.0149994384  0.0149994384  0.0149994384 -0.0053718721
## gouvern       0.0035374353  0.0035374353  0.0035374353  0.0352438256
## tribologi    -0.0005545037 -0.0005545037 -0.0005545037 -0.0034527011
## perfectionn  -0.0462678165 -0.0462678165 -0.0462678165 -0.0201699339
## hitchcock    -0.0022786089 -0.0022786089 -0.0022786089 -0.0016862276
## fractal       0.0007788709  0.0007788709  0.0007788709 -0.0034538575
```

5) TF-IDF -> LSA

Voici le code pour générer la matrice TF-IDF LSA qui prend comme entrée la matrice TF-IDF:

```
lsaSpace.tfidf <- lsa(tf.idf,dims=50)
X.lsa.tfidf <- as.textmatrix(lsaSpace.tfidf)
X.lsa.tfidf[1:20,1:8]
```

```
##                     AE3100        AE3205        AE3300        AE3400
## null         -0.0003352624 -0.0005555302 -0.0004846415 -0.0008293961
## huileus       0.0551433816 -0.0138169818 -0.0949674077 -0.0273934356
## four          0.0036951760 -0.0170533463 -0.0361128947 -0.0240639096
## prefix       -0.0054366931  0.0056634746  0.0190212652 -0.0267488681
## polytechniqu  0.0064900599 -0.0097400621 -0.0701775897 -0.0651260230
## travaux      -0.0135356329  0.0017970357  0.0464726597  0.1632907798
## criter       -0.0258545438 -0.1303977962 -0.3119991311 -0.0574586552
## fourni       -0.0019176491  0.0101485358 -0.0016056314 -0.0102056410
## rglage        0.0182216836  0.1607528261 -0.0044859199  0.4480153226
```

```
## dela          0.0476571159 -0.0025229803 -0.0340975032 -0.0210305542
## bioconcentr   0.0103980378 -0.0298214181 -0.0973804296  0.0186448563
## econo         0.0007197553 -0.0004685778  0.0031958524 -0.0117374134
## hough         0.0053202426 -0.0613571066 -0.0433214821 -0.0234454441
## regional      0.0677295088  0.0136036585 -0.0875412566  0.0100918895
## lumir         0.0016488316  0.0419461151 -0.0037515666 -0.0064678232
## gouvern       0.0650257666  0.2938389561  0.1270066508  0.4895953564
## tribologi     0.0076003083  0.0133618267  0.0115817662  0.0408707848
## perfectionn   0.0365389793  0.0583764508 -0.0574571139  0.0817595604
## hitchcock    -0.0014138469 -0.0160212859  0.0640861179  0.0544301527
## fractal      -0.0153813433 -0.0314394610 -0.0149012248 -0.0029950080
##                     AE3900       AE3900A       AE3900B       AE4000
## null         -0.0009885194 -0.0009885194 -0.0009885194 -0.001074518
## huileus       0.1097822356  0.1097822356  0.1097822356 -0.076417342
## four          0.0113237377  0.0113237377  0.0113237377 -0.071363970
## prefix       -0.0021486929 -0.0021486929 -0.0021486929  0.010535118
## polytechniqu  0.0158190509  0.0158190509  0.0158190509  0.152483150
## travaux       0.1769623857  0.1769623857  0.1769623857  0.044983214
## criter        0.0669411827  0.0669411827  0.0669411827  0.237009662
## fourni       -0.1154584398 -0.1154584398 -0.1154584398 -0.055888579
## rglage        0.0199374308  0.0199374308  0.0199374308  0.093080473
## dela         -0.0057430028 -0.0057430028 -0.0057430028  0.012966587
## bioconcentr  -0.0316839016 -0.0316839016 -0.0316839016  0.168663556
## econo        -0.0004983737 -0.0004983737 -0.0004983737  0.029400086
## hough         0.0166130197  0.0166130197  0.0166130197  0.071194466
## regional      0.0567521406  0.0567521406  0.0567521406 -0.048205886
## lumir        -0.0507774478 -0.0507774478 -0.0507774478  0.008571772
## gouvern       0.0114672909  0.0114672909  0.0114672909  0.088152265
## tribologi     0.0133481113  0.0133481113  0.0133481113 -0.021252722
## perfectionn  -0.2094381448 -0.2094381448 -0.2094381448 -0.430851586
## hitchcock    -0.1057447147 -0.1057447147 -0.1057447147  0.047240274
## fractal       0.0463382684  0.0463382684  0.0463382684  0.031720908
```

6) Log-Entropy -> LSA

Voici le code pour générer la matrice Log-Entropy LSA qui prend comme entrée la matrice Log Entropy:

```
lsaSpace.entropy <- lsa(log.entropy, dims=50)
X.lsa.ent <- as.textmatrix(lsaSpace.entropy)
X.lsa.ent[1:20,1:8]
```

```
##                     AE3100       AE3205       AE3300        AE3400
## null          0.7441498096  0.874888321  0.827320179  1.0604524685
## huileus       0.0067356181  0.002475173 -0.003402578  0.0063297577
## four         -0.0007833269 -0.003194106 -0.002587797 -0.0033940333
## prefix       -0.0021801482  0.001663038  0.001918855 -0.0028120754
## polytechniqu -0.0046971678 -0.020522902 -0.005202331 -0.0338047774
## travaux       0.0664201710  0.070590449  0.096609244  0.0402022074
## criter        0.0319519883 -0.025310338  0.034400620 -0.0243825844
## fourni        0.0019722802  0.006818856  0.004608478  0.0003664759
## rglage        0.0173703935  0.005878950 -0.003436918  0.0381415115
## dela          0.0020315736 -0.005315902 -0.004970365 -0.0094627007
## bioconcentr   0.0001375267 -0.006400572 -0.006712731  0.0009278136
## econo         0.0003904856  0.003924093  0.003932111 -0.0001422878
## hough         0.0017644463 -0.004138666 -0.003531877 -0.0031175062
```

```
## regional      0.0037049791 -0.007514428 -0.001249813 -0.0049404594
## lumir        -0.0038805660  0.003728684  0.001177303 -0.0027005576
## gouvern       0.0034989409  0.054074550  0.017702639  0.0864238990
## tribologi     0.0008730663  0.001788126  0.006637931  0.0089262455
## perfectionn   0.0144469904  0.042953781 -0.001207360  0.0264592715
## hitchcock     0.0018273449 -0.006443045  0.002888981  0.0041332773
## fractal       0.0065355026 -0.008503041 -0.006782726 -0.0008199747
##                      AE3900       AE3900A       AE3900B        AE4000
## null          1.080260e+00  1.080260e+00  1.080260e+00  1.0429591833
## huileus      -6.781427e-03 -6.781427e-03 -6.781427e-03  0.0037634975
## four         -1.272503e-03 -1.272503e-03 -1.272503e-03 -0.0079546532
## prefix        4.119997e-03  4.119997e-03  4.119997e-03  0.0003954601
## polytechniqu  1.878881e-02  1.878881e-02  1.878881e-02 -0.0053701086
## travaux       1.340690e-01  1.340690e-01  1.340690e-01  0.0789840431
## criter       -5.201361e-02 -5.201361e-02 -5.201361e-02  0.0242747195
## fourni       -1.189755e-02 -1.189755e-02 -1.189755e-02 -0.0008726018
## rglage       -2.308122e-03 -2.308122e-03 -2.308122e-03  0.0191460289
## dela         -2.455963e-03 -2.455963e-03 -2.455963e-03 -0.0074708819
## bioconcentr   1.024664e-04  1.024664e-04  1.024664e-04  0.0091985361
## econo         9.851443e-05  9.851443e-05  9.851443e-05  0.0002476979
## hough         1.462118e-04  1.462118e-04  1.462118e-04  0.0100638324
## regional     -3.224077e-03 -3.224077e-03 -3.224077e-03  0.0122243443
## lumir         1.449596e-02  1.449596e-02  1.449596e-02 -0.0056308600
## gouvern       1.309686e-02  1.309686e-02  1.309686e-02  0.0202761585
## tribologi    -1.664825e-04 -1.664825e-04 -1.664825e-04 -0.0034784592
## perfectionn   1.362125e-02  1.362125e-02  1.362125e-02 -0.0099537455
## hitchcock    -5.053906e-04 -5.053906e-04 -5.053906e-04 -0.0002206394
## fractal      -9.924343e-03 -9.924343e-03 -9.924343e-03  0.0054463030
```

Résultats

Dans cette section, on traitera de la méthodologie et les principaux résultats seront présentés. ## Méthodologie Notre expérience peut être segmentée en 5 étapes que voici :

1) Calcul des matrices: Pour nos 6 modèles les matrices de recommandation sont générées. En ce qui concerne les méthodes LSA, on doit fixer le nombre de dimensions de la réduction. Habituellement, si on avait à notre disponibilité un "ground truth", on aurait fixé comme hyperparamètre ce nombre de dimensions optimal et on l'aurait déterminé par validation croisée. Comme ce n'est pas le cas, on pose ce nombre de dimensions à 50 de façon arbitraire et on le fera varier dans les analyses post-traitement.

2) Une fois les matrices compilées, on calcule les corrélations de Spearman entre les cours et, pour un cours cible, on sort les 5 cours avec corrélations maximales pour les recommandations.

3) Sélection de 6 cours tests de domaines différents: comme on dispose de 1200 cours, on rétréciera notre ensemble de test pour effectuer le sondage. Étant donné que ce sont des cours que nous avons suivis et aimés, nous serons en mesure de juger les recommandations adéquatement. Les cours choisis sont les suivants:

- MEC2115: Méthodes expérimentales et mesures en mécanique
- AR320: Aérodynamique II
- MEC1210: Thermodynamique
- MTH1006: Algèbre Linéaire
- INF2010: Structures de données et algorithmes

- IND4704: Théorie de la décision

Au total, il y a eu 5 recommandations pour 6 méthodes et pour 6 cours, donc 180 notes au total. Pour réduire le temps du sondage, Mikael a évalué les cours MEC1210, MTH1006 et MEC2115 alors que Claude a évalué INF2010, AR320 et IND4704.

4) Pour les recommandations de ces 6 cours, on effectue un sondage dans lequel on octroie des notes allant de 1 à 5 pour chaque recommandation. Faute de temps, les deux seuls répondants à ce sondage sont Mikael et Claude. Les qualités qui sont jugées dans ce sondage sont la pertinence et la nouveauté des recommandations. Plus précisément, voici la signification des notes sur laquelle l'équipe s'est entendue:

- 1: Aucun lien avec le cours cible
- 2: Même domaine, mais pas les mêmes concepts (ex: un cours de calcul est recommandée pour un cours d'algèbre linéaire)
- 3: Mêmes concepts, différents domaines (ex: un cours de thermodynamique en génie chimique est recommandée pour un cours de thermodynamique en génie mécanique)
- 4: Cours similaires en tous points, mais redondant.
- 5: Cours similaires en tous points et qu'on serait interessé à suivre pour étendre nos horizons découlant du cours cible.

Par ailleurs, nous avons pensé à créer notre propre "ground truth", mais on s'est arrêté à deux problèmes principaux. D'une part, on aurait pu créer une métrique de précision basée sur le nombre de cours dans le même programme que l'algorithme réussit à prédire. Toutefois, comme notre but est de prédire des cours hors cursus, cette métrique ne serait pas un indicateur de succès. D'autre part, la seule méthode légitime pour se créer un "ground truth" aurait été de parcourir toutes les descriptions de cours et de sélectionner manuellement des descriptions qui seraient voulues selon nos critères subjectif. Or, comme il aurait fallu passer au travers de tout le corpus pour avoir une vision exhaustive des cours qui devraient optimalement être recommandées, nous nous sommes dit qu'un sondage accomplirait le même objectif, tout en réduisant le temps d'analyse (environ 1200 descriptions de cours).

5) Analyses post-traitement. Ces analyses sont des variations et des études plus poussées par rapport à la méthode de calcul initiale. On parle ici de variation du nombre de dimensions des méthodes LSA et une méthode de focus sur les dimensions cibles qui sera expliquée prochainement.

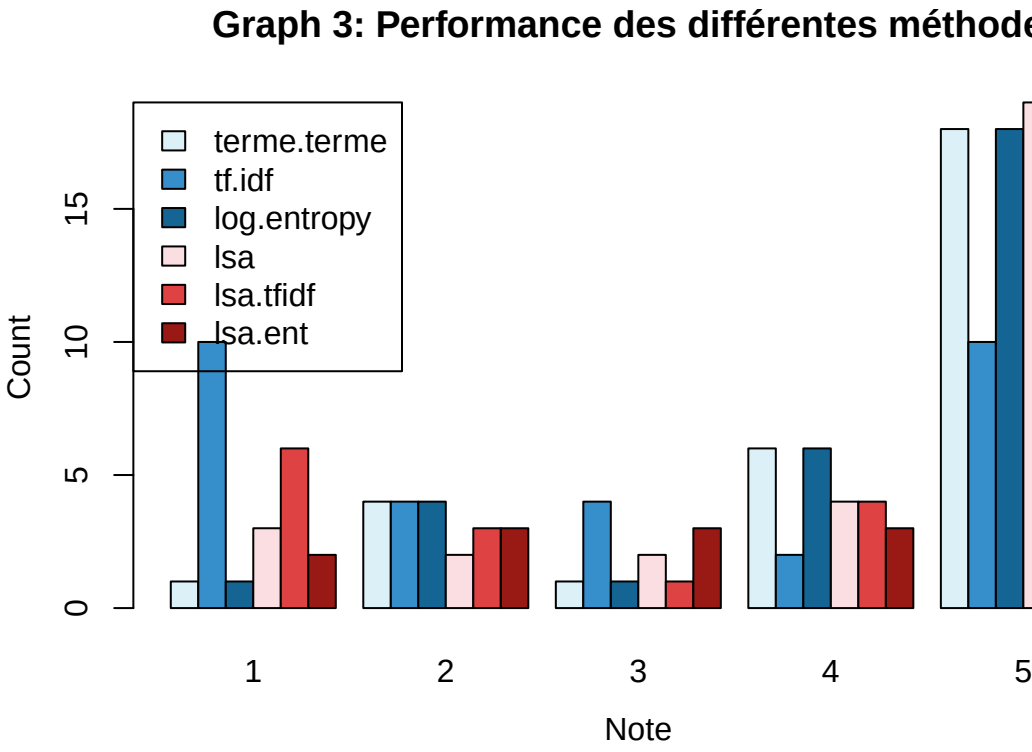
Présentation des résultats

D'abord, voici un exemple de recommandations en fonction des méthodes pour le cours de thermodynamique (l'ordre de haut en bas est l'ordre décroissant des corrélations) :

```
##      terme.terme tf.idf     log.entropy lsa       lsa.tfidf  lsa.entropy
## [1,] "PHS1104"   "GCH6112A" "PHS1104"   "GCH1510" "GCH2525"  "PHS1104"
## [2,] "GCH1510"   "ELE6216"  "GCH1510"   "PHS1104" "MET6208"  "GCH1510"
## [3,] "M-252"     "GBM6125"  "M-252"     "M-254"   "GCH1510"  "MEC3215"
## [4,] "MEC3215"   "ICM4316"  "MEC3215"   "GBM2620" "GCH3510"  "MTR2211"
## [5,] "MTR2211"   "GLQ3205"  "MTR2211"   "MTR2211" "GCH6912A" "GBM2620"
```

Pour ce cours, on remarque d'abord que les recommandations pour chacune des méthodes sont assez similaires. De plus, on observe aussi que, malgré le fait que ce soit un cours de génie mécanique, ce sont plutôt des cours de génie physique et génie chimique qui apparaissent comme recommandation principale. Ceci est logique, puisque la thermodynamique est une matière qui relève directement de ces domaines.

Montrons maintenant un graphique qui synthétise les résultats du sondage (ces derniers sont présentés intégrale-



ment dans le fichier resultat.csv) :

Dans la figure ci-dessus, l'axe des abscisses représente les notes qui ont été octroyées aux cours. Sur l'axe des ordonnées, on trouve le nombre de fois que ces notes ont été données. Les bandes bleues ont été utilisées pour illustrer les méthodes non-LSA (corrélation faite directement sur la matrice termes-documents, moyennant parfois certains traitements) tandis que les bandes rouges sont utilisées pour les méthodes LSA (matrice termes-documents parfois traitée et donnée en entrée dans une fonction LSA). A priori, on observe que les recommandations sont très bonnes et ce, pour la plupart des méthodes.

Les moyennes des cours pour chaque méthode sont compilées dans le tableau suivant:

```
##         terme.terme tf.idf log.entropy lsa lsa.tfidf lsa.ent Moy.Totale
## MEC1210         4.6    1.6         4.6 4.6       4.8     4.6       4.13
## MTH1006         4.2    3.6         4.2 5.0       2.6     4.4       4.00
## MEC2115         3.6    2.4         3.6 4.0       1.4     3.0       3.00
## INF2010         4.2    4.0         4.2 3.6       3.8     3.6       3.90
## AR320           5.0    3.6         5.0 5.0       5.0     5.0       4.77
## IND4704         3.6    2.4         3.6 2.6       4.6     4.2       3.50
```

On voit dans ce tableau que les deux notes totales les plus hautes sont les cours de thermodynamique (MEC1210) et Aérodynamique (AR320) alors que les deux notes les plus basses sont les cours d'instrumentation (MEC2115) et de prise de décision industrielle (IND4704). Une explication plausible est que ces méthodes performent bien pour des domaines techniques. Par exemple, en aérodynamique, les mêmes termes technique reviennent plus souvent (ex: profil d'aile, écoulement de Bernouilli, couche limite) et donc, les méthodes non-LSA performent mieux. En ce qui concerne les cours, avec les moins bonnes notes, on remarque qu'il s'agit de cours plutôt généraux, la théorie de la mesure ainsi que la précision étant commune à plusieurs domaines.

Ensuite, on présente un tableau synthétisant les moyennes des notes des méthodes pour chaque répondant:

```
##             Moy.Mikael Moy.Claude Moy.Totale
```

```
## terme.terme       4.13       4.27       4.20
## tf.idf            2.53       3.33       2.93
## log.entropy       4.13       4.27       4.20
## lsa               4.53       3.73       4.13
## lsa.tfidf         2.93       4.47       3.70
## lsa.ent           4.00       4.27       4.13
```

On observe deux points importants. D'une part, les méthodes log.entropy et terme-terme donnaient exactement les mêmes résultats. D'autre part, les méthodes impliquant le TF-IDF en pré-traitement donnent nettement les résultats les moins bons; nous essaierons d'investiguer pourquoi dans la section suivante. # Analyse Dans cette section, on procèdera à différentes analyses post-traitement sur les résultats obtenus à la section précédente.

TF-IDF

Afin d'investiguer la raison pour laquelle les méthodes impliquant le TF-IDF donnent les moins bons résultats, on sort les mots avec les TF-IDF maximaux (calculés avec la normalisation logarithmique de l'article [1]) pour chacun de nos cours tests:

```
##         [,1]          [,2]       [,3]        [,4]          [,5]
## MTH1006 "vectoriel"   "matric"   "espac"     "homogen"     "vecteur"
## MEC1210 "parfait"     "pure"     "entropi"   "substanc"    "cycl"
## MEC2115 "informatis"  "acquisit" "huileus"   "prefix"      "fourni"
## AR320   "arodynamiqu" "hlicoptr" "coulement" "potentiel"   "profil"
## INF2010 "arbr"        "file"     "algorithm" "sequentiel"  "manipul"
## IND4704 "decis"       "jeux"     "collect"   "prise"       "huileus"
##         [,6]      [,7]          [,8]      [,9]          [,10]
## MTH1006 "lineair" "propr"       "huileus" "prefix"      "fourni"
## MEC1210 "melang"  "huileus"     "prefix"  "fourni"      "dela"
## MEC2115 "dela"    "bioconcentr" "econo"   "hough"       "lumir"
## AR320   "thori"   "huileus"     "prefix"  "fourni"      "dela"
## INF2010 "graph"   "huileus"     "prefix"  "fourni"      "dela"
## IND4704 "prefix"  "fourni"      "dela"    "bioconcentr" "econo"
```

Or, on voit que les termes "huileus", "prefix" et "fourni" se retrouvent dans les tops de tous les cours. Regardons maintenant les mots à TF-IDF maximaux dans tout le corpus. On sort les 100 mots ayant les TF-IDF maximaux et minimaux :

```
##              Bottom            Top
## 1              null     prcontraint
## 2         titrecour             jat
## 3   descriptioncour         surtens
## 4                de         chausse
## 5                 un    manufacturir
## 6               dan            houl
## 7            analys             ada
## 8            applic        temporis
## 9           concept           lumir
## 10              aux             sst
## 11           system      formabilit
## 12         techniqu             ctl
## 13          travail              cp
## 14          rapport             cfd
## 15           method       decrochag
## 16           projet         huileus
## 17              sou            pass
```

```
## 18        introduct        cologiqu
## 19           notion            dgel
## 20          gestion     arolasticit
## 21          problem  microcontrleur
## 22          princip         nerveux
## 23          structur             mef
## 24              tude  biomicrosystem
## 25          programm       diphasiqu
## 26              base          dsagrg
## 27            utilis        polyphas
## 28            calcul       radiofrqu
## 29              type           meubl
## 30       industriel        anticorp
## 31              geni          osseux
## 32            etudi       clavardag
## 33              etud        arospati
## 34          fonction             iao
## 35              heur           langu
## 36          recherch           breve
## 37            direct     turbomoteur
## 38             mesur  heterostructur
## 39              cour  nanotechnologi
## 40          developp         ontolog
## 41             model               f
## 42          mecaniqu    trigonometri
## 43             evalu             tec
## 44          logiciel           siecl
## 45          dynamiqu         sensori
## 46              moin           patch
## 47           pratiqu          dmodul
## 48          propriet             irc
## 49             ainsi  optolectroniqu
## 50          processu          camion
## 51             donne      arolastiqu
## 52       professeur       enterpris
## 53                se         chainag
## 54              norm              by
## 55       laboratoir    hypersustent
## 56           present         shannon
## 57                ca           clamp
## 58               non      cancerigen
## 59            realis        calibrag
## 60            commun       informati
## 61                 s           galoi
## 62           modelis      trajectoir
## 63             equip    biophotoniqu
## 64           product      doublement
## 65           physiqu       petrolier
## 66             effet             sti
## 67         ingenieri             lin
## 68           control         inciner
## 69              autr              gi
## 70             outil           modbu
## 71              form          prsenc
```

```
## 72           integr        urbanism
## 73             mise            hook
## 74            systm         estuair
## 75          definit             mcu
## 76         numeriqu         heritag
## 77             entr             fly
## 78   caracteristiqu       perglisol
## 79         environn            mare
## 80          activit     geomecaniqu
## 81       traitement        multitag
## 82          pendant            cuve
## 83            solut       biosystem
## 84             oral      clairement
## 85           redact        opratoir
## 86           niveau        extrieur
## 87         descript         composs
## 88           theori          papeti
## 89          chimiqu           parit
## 90            simul        directif
## 91            stage        geochimi
## 92           proced        luminair
## 93          comport         metteur
## 94        entrepris      flexibilit
## 95           qualit      sdimentair
## 96        materiaux            poli
## 97             temp          enfant
## 98         supervis            parc
## 99             plan  neuromusculair
## 100             fin sedimentologiqu
```

Par exemple, on remarque que le terme “huileus” survient à la 16e position des plus grands TF-IDF dans le corpus. Aussi, on sait que la normalisation logarithmique fait en sorte que des termes n’ayant pas de valeur dans la matrice termes-documents (i.e. ne se trouve pas dans le document) peuvent avoir une valeur dans la matrice TF-IDF. Conséquemment, on comprend pourquoi des termes ayant de hauts TF-IDF globaux peuvent avoir du poids dans les recommandations par rapport à un cours ne possédant même pas ces termes dans sa description, ce qui vient assurément fausser les résultats pour les méthodes impliquant TF-IDF (TF-IDF et LSA-TF-IDF). Notons que la méthode LSA classique, par exemple, n’est pas affectée par ce problème, puisque c’est la fonction LSA qui fait son propre prétraitement TF-IDF.

Par ailleurs, pour remédier à ce problème, les auteurs de l’articles supprimaient les termes ayant un TF-IDF supérieur à une certaine borne et cette borne était déterminée par validation croisée. Cependant, étant donné que nous ne possédons pas de “ground truth”, on ne peut pas déterminer cette borne. Alors, pour simplifier le tout, nous allons regénérer les TF-IDF maximaux de deux façons différentes: 1)En posant nulle les valeurs des termes dans la matrice TF-IDF qui ne possènent pas de valeur dans la matrice termes-documents 2) En calculant un TF-IDF classique sans normalisation logarithmique. Pour les résultats qui suivent, ces deux méthodes donnent exactement la même sortie, on les traitera donc comme une seule. Voici les TF-IDF maximaux de nos cours avec la correction :

```
##         [,1]          [,2]         [,3]       [,4]         [,5]
## MTH1006 "vectoriel"   "espac"      "lineair"  "matric"     "homogen"
## MEC1210 "cycl"        "parfait"    "pure"     "entropi"    "substanc"
## MEC2115 "mesur"       "informatis" "acquisit" "mecaniqu"   "instrument"
## AR320   "arodynamiqu" "coulement"  "hlicoptr" "potentiel" "profil"
## INF2010 "algorithm"   "arbr"       "file"     "donne"      "sequentiel"
## IND4704 "decis"       "prise"      "jeux"     "collect"    "actuariel"
```

```
##          [,6]      [,7]      [,8]             [,9]          [,10]
## MTH1006 "vecteur" "plan"    "propr"          "schmidt"     "orthonormal"
## MEC1210 "melang"  "gaz"     "thermodynamiqu" "otto"        "brayton"
## MEC2115 "experim" "instrum" "conditionneur"  "metrologiqu" "mesurag"
## AR320   "thori"   "hlice"   "joukowski"      "doublet"     "rotor"
## INF2010 "manipul" "graph"   "structur"       "retrait"     "parcour"
## IND4704 "repet"   "lectr"   "ahp"            "multicriter" "nash"
```

On voit maintenant qu'il n'y a plus de termes irréguliers. De plus, on conclut aussi que TF-IDF fait un bon travail pour identifier les termes significatifs pour chacun des cours. En regénérant d'autres recommandations avec cette correction, on conclut que le TF-IDF modifié donne exactement les mêmes résultats que termes-termes tandis que LSA-TF-IDF donnent les mêmes résultats que LSA classique.

Variations LSA

En premier lieu, nous allons générer des recommandations pour les trois méthodes LSA avec un nombre de dimensions réduites de 100 au lieu de 50 afin d'étudier quelles sont les différences. Dans le but d'abréger cette section et puisque répondre au sondage était long et fastidieux, nous allons présenter un comparatif des moyennes pour le cours MEC2115 d'intrumentation:

```
##           Moy.50 Moy.100
## lsa          4.0     3.8
## lsa.tfidf    1.4     3.0
## lsa.ent      3.0     3.0
```

On voit que les résultats sont très rapprochés pour les méthodes LSA et LSA-Entropy, mais on observe une différence marquée pour LSA-TF-IDF. Toutefois, rappelons qu'il s'agit du TF-IDF sans correction, avec la normalisation logarithmique. Lorsqu'il est corrigé, les résultats sont identiques à LSA. Pour ce qui est de la nature des recommandations, les cours recommandés sont quasi identiques pour LSA-Entropy, mais diffèrent significativement pour LSA (tout en restant de qualité similaire). En somme, on conclut que l'effet du nombre de dimensions n'a pas beaucoup d'importance sur la qualité des recommandations pour ce problème.

En second lieu, on voulait voir si les dimensions réduites conservées par LSA étaient dépendantes de l'ordre des cours et des termes dans la matrice termes-documents initiale. Pour ce faire, on a pris un cours qui donnait beaucoup de recommandations hors cursus. Le cours de thermodynamique (MEC1210) génère beaucoup de recommandations de cours en génie chimique physique, ce qui est normal et voulu considérant la nature du cours. Nous avons donc positionné les 50 cours de mécanique et les 50 termes ayant les plus haut TF-IDF en mécanique dans les 50 premières lignes et colonnes de la matrice termes-documents. De cette manière, on voulait voir si davantage de recommandations en génie mécanique allait être générées. La réponse est catégorique: les recommandations sont en tout point identiques peu importe comment les lignes et colonnes de la matrice initiale sont ordonnées. Ainsi, on conclut que les dimensions latentes font fi de cet arrangement et arrivent véritablement à dégager le sens global de l'ensemble de données initial.

Corrélations termes-termes

Finalement, dans le but d'essayer de comprendre quels sont les étapes intermédiaires de LSA pour dégager la sémantique des textes, nous avons décidé de faire un comparatif des corrélations entre les termes avant et après l'application de LSA.

Donc, 13 termes représentant bien nos cours tests ont été choisis et les termes les plus similaires basés sur la corrélation ont été retournés dans la matrice termes-documents. On obtient ceci :

Termes les plus similaires (après LSA)

```
##            [,1]         [,2]         [,3]           [,4]
## vectoriel  "scalair"    "vecteur"    "epissur"      "bimodaux"
## lineair    "non"        "differenti" "troncatur"    "interpol"
## nergi      "solair"     "variateur"  "lectriqu"     "ractiv"
```

```
## chaleur    "transfert"  "convect"    "carnot"        "echangeur"
## mesur      "etalonnag"  "calibr"     "instrument"    "facteur"
## donne      "langag"     "algorithm"  "acquisit"      "base"
## algorithm  "tri"        "heuristiqu" "graph"         "np"
## structur   "poutr"      "tectoniqu"  "mohr"          "solid"
## coulement  "laminair"   "visqueux"   "turbul"        "turbulent"
## potentiel  "couch"      "ail"        "visqueux"      "navier"
## viscosit   "diffusivit" "oxygen"     "robinet"       "darci"
## dcision    "cohsion"    "runion"     "motion"        "prouv"
## industriel "conjoint"   "moi"        "coordonnateur" "relie"
##            [,5]
## vectoriel  "interferometr"
## lineair    "guide"
## nergi      "varignon"
## chaleur    "conduct"
## mesur      "mesurag"
## donne      "successif"
## algorithm  "procedural"
## structur   "atomiqu"
## coulement  "potentiel"
## potentiel  "ecoul"
## viscosit   "laminair"
## dcision    "itinrair"
## industriel "exig"
```

Ensuite, on applique LSA sur la matrice termes-documents et on effectue le même calcul des termes les plus similaires. On obtient ceci :

Termes les plus similaires (après LSA)

```
##            [,1]         [,2]           [,3]             [,4]
## vectoriel  "scalair"    "bimodaux"     "interferometr" "spectraux"
## lineair    "non"        "ordr"         "linearis"      "algebriqu"
## nergi      "lectricit"  "prsentent"    "lectrocut"     "scuritair"
## chaleur    "convect"    "conduct"      "transfert"     "echangeur"
## mesur      "etalonnag"  "instrum"      "conditionneur" "mesurag"
## donne      "sql"        "data"         "entit"         "productiqu"
## algorithm  "np"         "asymptotiqu"  "tri"           "campeur"
## structur   "evolutif"   "contemporain" "valenc"        "repuls"
## coulement  "hlice"      "joukowski"    "doublet"       "rotor"
## potentiel  "visqueux"   "turbul"       "laminair"      "fluid"
## viscosit   "laminair"   "visqueux"     "schwarz"       "conduis"
## dcision    "motion"     "prouv"        "runion"        "cohsion"
## industriel "smed"       "die"          "exchang"       "allege"
##            [,5]
## vectoriel  "fusionn"
## lineair    "entier"
## nergi      "neutr"
## chaleur    "ebullit"
## mesur      "informati"
## donne      "relationnel"
## algorithm  "gloutonn"
## structur   "attract"
## coulement  "portanc"
## potentiel  "idal"
```

```
## viscosit   "absolu"
## dcision    "prliminair"
## industriel "lean"
```

A priori, on observe plusieurs similarités entre ces matrices, mais aussi certaines différences. Par exemple, si on prend le terme "algorithm", les termes qui diffèrent d'une matrice à l'autre sont "heuristique", "graph" et "procedural" pour la matrice avant LSA versus "asymptotique", "campeur", "gloutonn". Par conséquent, on remarque que les mots liés à une corrélation basés sur la co-occurrence de termes (avant LSA) sont exclusivement dans le domaine du mot cible. En revanche, les mots liés à une corrélation sur la matrice post-LSA sont des termes qui pourraient être utilisés dans un autre contexte. Ici, on comprend que LSA a bien associé le contexte des termes "campeur" et "gloutonn" (on parle ici de l'algorithme sac de "campeur" et de "greedy algorithm") avec le contexte de "algorithm" et ce, sans qu'il y est nécessairement co-occurrence des termes. En effet, on voit qu'il n'y a jamais co-occurrence des termes "algorithm" et "campeur":

```
sum(m.poly[ which(rownames(m.poly)=="algorithm"),]==m.poly[which(rownames(m.poly)=="campeur"),]&m.poly[
```

```
## [1] 0
```

Conclusion

Pour conclure, on énoncera les principales conclusions de cette étude ainsi que les étapes futures du projet. ## Discussion et conclusions sur l'analyse Les conclusions sur l'analyse se divise en trois points.

Méthodes non LSA

Les méthodes non-LSA (termes-termes, TF-IDF, Log-entropy) fonctionnent très bien pour ce contexte. En revanche, il faut garder en tête qu'il s'agit d'un problème dans lequel les termes sont très techniques. Ceci signifie qu'il arrive fréquemment que des termes identiques soient utilisés pour exprimer un concept et donc, les méthodes de co-occurrences fonctionnent bien. Par exemple, si on prend le cours d'aérodynamque (AR320) qui possède la description suivante:

"TitreCours: Arodynamique II DescriptionCours: coulement stationnaire incompressible en 2D. coulement potentiel. Potentiel de vitesses. Fonction de courant. Singularits : sources, doublets, tourbillons. Potentiel complexe. Profils arodynamiques. Thorme de Kutta-Joukowski. Caractristiques arodynamiques des profils. Aile d'envergure finie. Thorie de la ligne portante. Viscosit. Couche limite laminaire sur une plaque plane. coulements compressibles. Onde de choc. Nombre de Mach. coulement supersonique. Arodynamique de l'hlicoptre. Thorie des hlices. Rotor d'hlicoptre."

La recommandation la plus populaire par les méthodes non LSA est le cours AE4300 dont la description est la suivante: "TitreCours: Arodynamique de l'avion II DescriptionCours: Interaction couche limite-coulement potentiel. Forces de trane. Calcul des caractristiques arodynamiques (portance, trane, moment) de profils d'ailes en coulement plan incompressible. Mcanismes et techniques de l'hypersustentation. tude fondamentale de l'hypersustentation et estimation de portance maximum. Mthode des lments de surface pour le calcul arodynamique des ailes. Ailes en coulement compressible subsonique. Arodynamique transsonique : thorie des profils d'aile, solutions gnrales d'coulements transsoniques, aile d'envergure finie et avion en coulement transsonique."

On voit donc que les cours sont extrêmement similaires, que ce soit à cause des termes utilisés que ce qu'ils proposent en général.

Méthodes LSA

En ce qui concerne les méthodes LSA, les résultats globaux sont moins bon, mais plus intéressant. Ceci s'explique par le fait qu'elles fourni des recommandations qui respectent la maxime "high risk, high reward". Effectivement, les recommandations étaient plus souvent osées, ce qui résultait parfois en des propositions inadéquates. Cependant, lorsque celles-ci étaient justes, elles étaient souvent intéressante et nouvelle. Par exemple, pour le cours d'algèbre linéaire (MTH1006) dont la description est la suivante:

“TitreCours: Algebre lineaire DescriptionCours: Plan et espace euclidiens. Vecteurs geometriques du plan et de l’espace. Produits scalaire, vectoriel et mixte. Droites et plans. Espaces vectoriels, sous-espaces vectoriels, independance lineaire, base, dimension. Bases orthogonales et orthonormales, procede de Gram-Schmidt. Transformations lineaires, matrices et changement de bases. Noyau, image et rang. Systemes d’equations lineaires homogenes, non homogenes et liens avec les matrices. Valeurs propres et vecteurs propres. Diagonalisation. Formes quadratiques et matrices symetriques. Applications a la geometrie : classification des equations du second degre (coniques et quadriques).”

Un des cours que LSA nous a retourné est MTH3215 dont la description est :

“TitreCours: Mathematiques pour les applicat. multimedias DescriptionCours: Interpolation, differentiation et integration numerique. Resolution numerique des equations algebriques. Methodes directes et iteratives pour les systemes d’equations algebriques lineaires et non lineaires. Modelisation mathematique. Erreurs de modelisation, de representation et de troncature. Theorie des equations aux derivees partielles. Methode de separation des variables. Series de Fourier.”

On voit ici que les cours sont beaucoup moins similaires que les cours d’aérodynamique, mais ils sont tout de même liés. On sort du domaine de l’algèbre linéaire pour découvrir le domaine du calcul numérique dans les applications multimédias. On émet l’hypothèse que ce choix est plus intéressant pour l’utilisateur qu’un autre cours d’aérodynamique. Cependant, il arrivait aussi quelques fois que la recommandation n’était pas du tout en lien avec le cours cible.

Robustesse des méthodes

Enfin, que ce soit en faisant varier le nombre de dimensions réduites ou en réarrangeant la matrice initiale, les recommandations se sont avérées très robustes et ce, pour toutes les méthodes. De plus, la pertinences des recommandations s’est avérée généralisée pour la plupart des cours, avec une légère distinction par rapport au niveau de technicalité du cours (meilleure recommandation pour des cours plus techniques).

Étapes Futures

À la suite de ce projet, nous aurions plusieurs étapes à compléter pour confirmer le uel des modèles est réellement le plus efficace, mais aussi pour rendre les recommandations encore meilleures. Voici la liste des étapes potentielles:

- Effectuer un sondage plus étendu. Pour l’instant, le sondage a été effectué seulement par Mikael et Claude. Avec plus de temps, nous devrions contacter plus d’étudiant de domaines différents pour juger de la qualité des recommandations. Nous aurions aussi un travail plus systématique nécessaire pour retirer les biais de ce sondage. Par exemple, mettre dans un ordre aléatoire les recommandations et les méthodes.
- Obtenir des données supplémentaires. Nous avions en tête d’obtenir des données pour ajouter un aspect collaboratif au système. Les données pourraient être la liste des cours suivis pour différents étudiants. De plus, avec ce genre de données, nous pourrions effectuer des méthodes de validation du style “Leave one out” en tentant de prédire les cours que les étudiants ont suivis.
- Ajouter des critères de hiérarchie de cours. Puisque notre système n’a aucune restriction pour les recommandations, le cours en entrée pourrait être un cours de maitrise et la recommandation, un cours de 1ˆere année de BAC. En créant une telle hiérarchie, on pourrait potentiellement augmenter la qualité des recommandations.
- Ajouter les cours d’autres universités. Pour le moment, nous couvrons seulement Polytechnique pour réduire le temps de calcul. Cependant, en utilisant tous les cours qui sont à notre diposition (UQAM, U de M et HEC), nous pourions améliorer la diversité des recommandations. Aussi, nous pourrions ajouter toutes les universités montréalaises pour avoir un système complet pour la grande région de Montréal. Certains cours dans d’autres universités pourraient grandement améliorer l’expérience de certains étudiants.

- Penser à des méthodes alternatives de validation des recommandations. Pour le projet, un sondage a été effectué. Cependant, une étape future serait définir si cette méthode est réellement la plus efficace.

Références

1. Achakulvisut T, Acuna DE, Ruangrong T, Kording K (2016) Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications. PLoS ONE 11(7): e0158423. doi:10.1371/journal.pone.0158423
2. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. Discourse Processes, 25, 259-284
3. Bergamaschi, S., Po, L., & Sorrentino, S. (2014). Comparing Topic Models for a Movie Recommendation System. WEBIST.