Actes du neuvième défi fouille de texte, DEFT2013, Les Sables-d'Olonne, France, 21 juin 2013. Pages 67-79.
Proceedings of the Ninth DEFT Workshop, DEFT2013, Les Sables-d'Olonne, France, 21st June 2013. Pages 67-79.

Participation de Orange Labs à DEFT 2013

Olivier Collin, Aleksandra Guerraz, Yannick Hiou, Nicolas Voisine
Orange Labs
2, avenue Pierre Marzin, 22307 Lannion Cedex, France
{olivier.collin,aleksandra.guerraz}@orange.com
{yannick.hiou,nicolas.voisine}@orange.com

RESUME

Cet article décrit notre participation à DEFT2013. Nous détaillons tout d'abord nos réponses et résultats pour les tâches 1 et 2 qui sont des problèmes de classification de recettes : classification en niveaux de difficulté (*facile, difficile…)* pour la tâche 1 et vers une typologie (*entrée, plat principal, dessert*) pour la tâche 2. Nous poursuivons ensuite par le traitement de la tâche 3 dont le but était de mettre en correspondance les titres des recettes en regard de leur description. Pour cette tâche, nous avons testé l'utilisation d'une distance sémantique entre mots, distance pré-calculée à partir de Wikipédia français.

ABSTRACT

Orange Labs participation to DEFT 2013

This paper describes our participation to DEFT2013. On the one hand we first detail our answers and results related to the tasks 1 and 2 which are classification tasks applied to recipes. Task 1 focused on the difficulty level of each recipe *(facile, difficile…)* and task 2 to the natural typology of recipes: *entrée, plat principal, dessert*. On the other hand, the goal of task 3 was to find the link between a recipe description and a title list. For this task, we applied a semantic distance calculus between words, pre-calculated on Wikipedia French corpus.

MOTS-CLÉS : classification supervisée, co-clustering, distance sémantique, Wikipédia, Khiops, TiLT
KEYWORDS : supervised classification, co-clustering, semantic distance, Wikipedia, Khiops, TiLT

1 Introduction

Le défi 2013 proposait 4 tâches, nous avons participé aux tâches 1, 2 et 3. Les deux premières tâches ont été traitées par des algorithmes de classification supervisée. La tâche 3 a été traitée par des calculs de similarité textuelle s'appuyant sur une métrique de distance entre mots. La tâche 4 n'a pas été traitée. Si les tâches 1 et 2 sont plutôt classiques, la nature des données et la confusion des étiquettes ont apporté une difficulté supplémentaire à ce type de tâche. Nous pensons avoir apporté un éclairage intéressant sur la tâche 2 par l'utilisation d'un outil de co-clustering permettant une factorisation utile des variables. En ce qui concerne la tâche 3, nous avons choisi d'explorer une métrique basée sur une distance sémantique inter mots introduite par Cilibrasi (Cilibrasi et Vitanyi 2006, 2007), (Cilibrasi *et al.*, 2009). Nous nous sommes placés dans un cadre applicatif que nous avons repris dans le challenge « AI Mashup Challenge » (Belaunde *et al.*, 2013). Ce cadre consiste à rapprocher un document (recette) d'un ensemble de titres issus de données vidéos (présentations vidéos

d'une recette).

Dans la section 2, nous décrivons le corpus et les prétraitements généraux que nous avons effectués pour les trois tâches. Notre contribution aux tâches 1 et 2 est présentée dans la section 3. La section 4, quant à elle, détaille l'approche que nous avons utilisée pour la tâche 3. Nous terminons enfin par une discussion sur les résultats et une conclusion sur notre participation au défi.

2 Prétraitements généraux

Le corpus d'entrainement est constitué de 13864 recettes de cuisine. Les recettes sont au format XML. Pour chaque recette, les champs « id », « titre », « type », « niveau », « coût », « ingrédients » et « préparation » sont renseignés. De plus, un fichier récapitulatif de ce corpus a été fourni. Il regroupe des informations collectées dans chaque recette ; on y trouve pour chaque recette un champ comportant les ingrédients normalisés (sans accent, en minuscule, en multi mots) et un champ composé du nombre de ces ingrédients normalisés. Au final, nous avons conservé chacun des champs des recettes mais nous n'avons pas utilisé les ingrédients normalisés. Nous n'avons pas non plus utilisé le fichier fournissant la liste des titres séparément des recettes.

Outre le prétraitement linguistique que nous avons effectué pour la tâche 3 (et décrit en tâche 3), nous avons dû tout d'abord traiter l'encodage des recettes. Cet encodage annoncé comme étant de l'UTF8 était ponctuellement bruité notamment par un encodage parasite de caractères HTML. L'encodage des données XML issues du WEB et notamment des flux RSS est très souvent bruité. Nous avons d'autre part converti en minuscules l'ensemble des données des corpus d'apprentissage et de test.

3 Tâche 1 et 2, généralités

Pour ces tâches, nous avons testé deux méthodes de classification différentes : une méthode utilisant le principe de maximum d'entropie et une autre qui exploite l'hypothèse Bayesienne naïve. Rappelons que la tâche 1 consistait à identifier à partir du titre et du texte de la recette son niveau de difficulté sur une échelle à 4 niveaux : *très facile, facile, moyennement difficile, difficile*. Quant à la tâche 2, elle consistait à classifier la recette selon le type de plat parmi *entrée*, *plat principal* et *dessert*. Des premiers essais de classification nous ont permis d'analyser les caractéristiques de chacune de ces tâches. La tâche 1 se caractérise par un étiquetage subjectif de la notion de difficulté d'un plat. Un calcul d'accord inter-annotateur aurait probablement montré une forte divergence des étiquettes. D'autre part, le corpus d'apprentissage est fortement déséquilibré et contient relativement peu d'exemples de la classe *difficile*. La tâche 2 se caractérise aussi par une confusion importante portant sur les classes *entrée* et *plat* alors que la classe *dessert* est plutôt bien différenciée des deux autres classes. Dans les deux cas, on se retrouve donc assez loin d'un corpus d'apprentissage idéal.

Concernant ces deux tâches, le premier essai nous a permis de réaliser une référence en utilisant une technique standard de type « Maximum d'Entropie » (Nigam *et al.*, 1999), notre but étant ensuite de tester un outil développé par Orange Labs : Khiops (Boullé, 2008). Pour réaliser notre classifieur de type Maximum d'Entropie, nous avons simplement téléchargé et adapté les entrées/sorties du code mis à disposition par (Tsuruoka *et al.*, 2009) à l'adresse

http://www.nactem.ac.uk/tsuruoka/maxent/.

Pour ces deux tâches, pour les autres essais, nous avons donc utilisé Khiops, un outil de préparation des données et de modélisation pour l'apprentissage supervisé et non-supervisé. Cet outil est basé sur une méthode de classification qui exploite l'hypothèse Bayesienne naïve (Boullé, 2007). Cette méthode estime les probabilités conditionnelles univariées. Elle effectue des discrétisations et groupements de valeurs optimaux pour les variables numériques et catégorielles. Elle recherche un sous-ensemble de variables consistant avec l'hypothèse Bayesienne naïve, en utilisant un critère d'évaluation selon une approche Bayesienne de la sélection de modèles et des heuristiques efficaces d'ajout/suppression de variables. Enfin, elle moyenne l'ensemble des modèles évalués en utilisant un lissage logarithmique de la distribution a posteriori des modèles. L'outil Khiops ne nécessite aucun paramétrage, ce qui constitue son point fort. Cet outil, développé par Orange Labs, est utilisé intensivement pour de nombreux problèmes d'analyse et de classification de données mais moins souvent sur des données textuelles. Il nous a permis de réaliser la classification supervisée des documents pour les tâches 1 et 2, ainsi qu'une analyse des variables pour chacun des modèles générés. Il offre, d'autre part, un algorithme de « co-clustering » que nous avons appliqué à des regroupements documents/mots. Ces regroupements sont à la fois utiles par la sémantique portée mais constituent aussi de nouvelles variables synthétiques permettant de réaliser, voire améliorer les tâches de classification supervisées. Khiops fournit d'autre part un ensemble d'outils d'analyse des variables qui permettent notamment d'ordonner et visualiser le poids relatif de chacune d'entre elles.

Nous n'avons utilisé aucun prétraitement de type linguistique (lemmatisation…) pour ces deux tâches. Par contre, différents types de variables ont été utilisés conjointement : des variables catégorielles comme le champ « coût » ainsi que des variables numériques comme le nombre de mots du champ « préparation ».

Pour pré-évaluer ces tâches, nous avons partagé le corpus d'apprentissage fourni en deux sous-corpus : 70% pour l'apprentissage et 30% pour le test.

Pour la tâche 1, nous observons un fort déséquilibre du nombre d'exemples par classe, la classe *difficile* étant particulièrement sous représentée.

3.1 Tâche 1

3.1.1 Essai 1

L'essai 1 utilise le classifieur de type « Maximum d'Entropie ». Les variables utilisées sont les variables catégorielles « coût » (*bon marché, moyen, assez cher*) et « type » (*entrée, plat, dessert*). Nous prenons également en compte comme variables les mots qui sont présents dans le titre, la liste d'ingrédients et dans la préparation. Certaines catégories de mots ont été filtrées en utilisant notamment la liste de « stop words » fournie par NLTK (http://nltk.org/). Ce sont notamment les articles, prépositions, pronoms personnels, adjectifs possessifs, négations, conjonctions, auxiliaires être et avoir. Nous avons en plus supprimé la plupart des termes représentant des valeurs numériques, des symboles de mesure de contenance, de poids… Les petits mots ne contenant qu'une ou deux lettres ont été aussi éliminés. Les variables numériques utilisées sont le « nombre de mot dans la

préparation » et le « nombre d'ingrédients ». Ces variables ont été discrétisées linéairement.

Les résultats sur notre sous-corpus de test sont très mauvais pour cette tâche : toutes les recettes sont classées comme "facile", soit un taux d'erreur de 66%.

Les fréquences relatives des classes sont probablement la cause de ce problème. Nous ne nous attendions donc pas à de bons résultats pour cet essai.

3.1.2 Essai 2

Pour la tâche 1 utilisant Khiops (essai 2), nous utilisons les variables catégorielles suivantes : « coût » (*bon marché, moyen, assez cher*) et « type » (*entrée, plat, dessert*). Les variables numériques utilisées sont : « nombre de mot dans la préparation » et « nombre d'ingrédients ». Nous prenons également en compte comme variables les mots qui sont présents dans le titre de la recette et dans sa préparation. Les variables « mots » liées aux ingrédients n'ont pas apporté une information utile pour cette tâche avec cet outil. Ce choix de variables a été réalisé suite à l'analyse de variables fournie par Khiops.

Les graphes suivants produits par l'environnement de Khiops illustrent cette analyse pour les variables les plus pertinentes pour notre tâche.

Nous observons ainsi une corrélation entre le niveau de difficulté d'une recette et son coût (figure 1). Cette figure illustre le fait que lorsqu'une recette est « *assez cher* » il est moins probable qu'elle soit très facile à réaliser.

FIGURE 1 – Distribution des valeurs de la variable « coût ».

Le nombre de mots de la préparation est aussi une variable importante (figure 2).

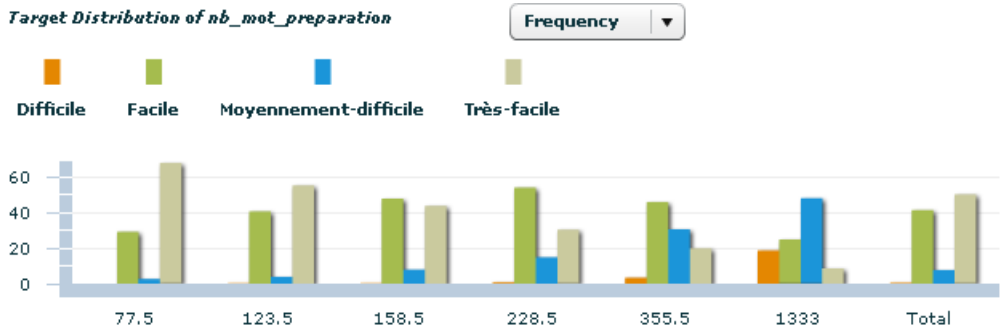

FIGURE 2 – Distribution des valeurs de la variable « nombre de mots de la préparation ».

Plus la recette est longue, plus la difficulté de la réalisation augmente.

Parmi les varibles sélectionnées par Khiops, nous trouvons des mots comme « poche » et

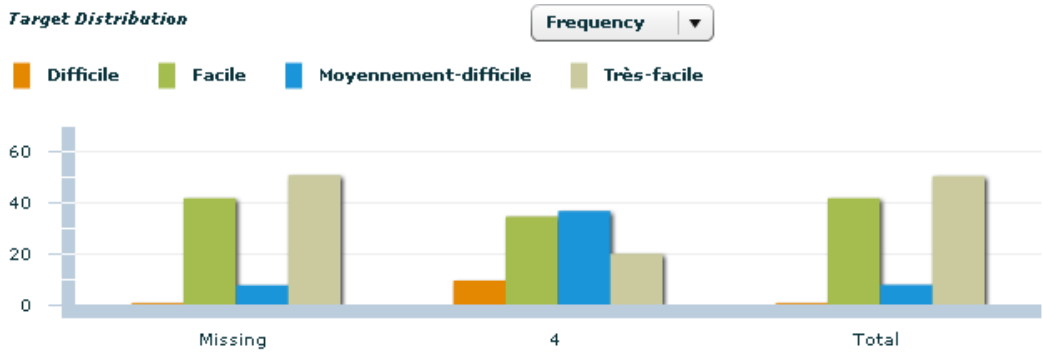

FIGURE 3 – Distribution pour le mot « poche ».

« salade ». Les figures 3 et 4 présentent la distribution de ces mots. Le mot « poche » est absent de la plupart de recettes. En revanche, quand il apparaît il est significatif : il est plus probable que la recette soit difficile à réaliser. Dans le corpus le mot « poche » apparaît dans les contextes « poche à douille » ou « poche à encre », il s'agit, en effet, de termes techniques. En ce qui concerne le mot « salade » (figure 4), il est plus probable qu'une recette soit facile à réaliser quand le « mot » salade apparaît.

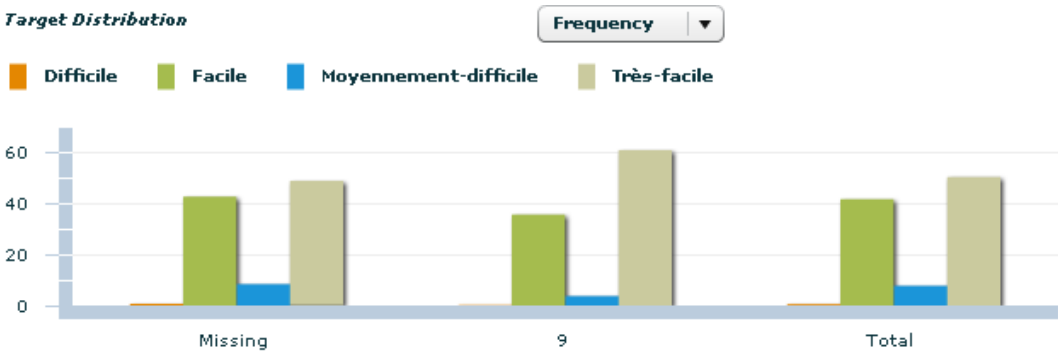

FIGURE 4 – Distribution pour le mot « salade ».

Outre la performance atteinte, cet environnement d'analyse s'avère très utile dans la compréhension du problème. Les tests de mise au point ont montré que Khiops proposait un résultat intéressant pour cette tâche. Mais, même si cet outil souffre moins du manque de données associées à la classe *difficile* et propose une modélisation et une analyse de variables pertinentes, la tâche reste délicate.

3.2 Tâche 2

3.2.1 Essai 1

L'essai 1 utilise le classifieur de type « Maximum d'Entropie », prenant à peu près les mêmes variables d'entrée que pour la tâche 1. Bien évidemment, la variable « type » n'est pas présente, mais nous avons utilisé la variable « niveau». Pour la mise au point de cet essai nous avons constitué un corpus d'apprentissage équi-réparti contenant 2268 exemples par classe, soit environ 50% des exemples disponibles. Les tests préliminaires ont donc été réalisés sur les exemples restants.

Contrairement à la tâche 1, les résultats sont utilisables. Nous avons notamment observé que des mots comme "tarte" et "quiche" sont « confondus » pour ces deux classes. Nous avons constaté que la classe « *dessert* » est plutôt bien discriminée, un *dessert* étant confondu avec un *plat* ou une *entrée* moins de 1% des cas. La confusion provient essentiellement des classes *entrée* et *plat principal*, une *entrée* étant confondue avec un *plat* presque une fois sur deux, un *plat* étant parfois confondu avec une *entrée* dans près de 8% des cas.

Pour notre test final, nous avons donc conservé une equi-répartition des données d'apprentissage de 3200 exemples par classes.

3.2.2 Essai 2

Pour la tâche 2 utilisant Khiops (essai 2), nous utilisons les variables catégorielles « coût » et « niveau » ainsi que les variables numériques « nombre de mot dans la préparation » et « nombre d'ingrédients », ainsi que les variables « mots » présents dans le titre et dans la préparation.

Pour cette tâche, nous utilisons Khiops pour une modélisation non-supervisée et supervisée. Nous avons donc aussi recours à sa fonctionnalité de co-clustering (Boullé, 2011). Les deux dimensions que nous utilisons sont les textes, et les mots. Après convergence de l'algorithme, nous obtenons 102 clusters de textes et 219 clusters de mots. Nous utilisons ensuite les clusters de textes pour la modélisation supervisée. Les recettes ne sont plus représentées par un ensemble de mots, mais sont classées dans un des 102 clusters. Nous utilisons également les variables catégorielles « coût » et « niveau » pour la modélisation supervisée et les variables numériques. Cependant ces variables n'ont pas de gros impact sur le modèle. La variable « cluster » a un pouvoir prédictif beaucoup plus significatif que les autres variables. L'analyse des regroupements de documents réalisés par le co-clustering permet d'expliquer ce résultat : les regroupements effectués de manière non-supervisée correspondent globalement à des regroupements d'entrées, de plat et de desserts. Si nous examinons plus finement les clusters de textes, nous observons en réalité une spécialisation par « cluster » dont voici quelques exemples :

Les recettes *Gâteau pomme-rhubarbe crousti-moelleux, Gâteau automnal, Gâteau aux fruits jaunes et aux pignons, Clafoutis corrézien* appartiennent au même « cluster », caractérisant un dessert de type gâteau. Nous trouvons également des clusters de desserts à base de fruits ou encore à base de chocolat.

De la même manière, les entrées *Velouté de printemps, Soupe aux marrons et aux carottes, Crème de céleri glacée à la menthe, Crème de cresson au caviar de Sevruga, Soupe d'endive à la bière* ont été regroupées au sein d'un même cluster qui semble correspondre à des entrées chaudes de type soupe ou crème.

Un autre exemple concerne des entrées à base de fruits de mer ou de poissons: *Pamplemousse de la mer, Mini-concombres farcis du pêcheur, Miettes de crabe et pamplemousse, Brochette des îles, Salade d'été au saumon fumé, feta, pommes et carottes…*

Pour les plats principaux, nous avons par exemple un cluster contenant *: Coq au vin de la mère Michèle, Bolognaise façon Mag, Bœuf mijoté façon bourguignon, Garbure landaise, Daube de canard au madiran, Daube de sanglier à la provençale …*

Finalement, examinons un cluster qui contient des entrées et des plats principaux et montre la difficulté de dissocier les classes *entrée* et *plat*: *Quiche à la truite fumée, aux courgettes et à la mozzarella (plat), Tarte à l'aubergine et à la brousse (entrée), Quiche à la tomate et aux oignons (plat), Quiche chorizo et tomate (entrée), Tarte moutardée au thon, tomates et poireaux (entrée), Tarte légère courgettes, jambon et chèvre gratiné (entrée), Tarte lardons champignons camembert (plat), Tarte à la tomate et au roquefort (plat), Quiche au poulet et à l'estragon (plat), Quiche au parmesan et tomates (entrée), Quiche au crabe (entrée).* On retrouve dans ces titres et dans ces recettes les termes « quiche » et « tarte » qui conviennent à un plat ou une entrée.

La figure 6 illustre la distribution des classes cibles (*entrée, plat, dessert*) pour un regroupement de cluster réalisé par Khiops. On visualise assez nettement la spécialisation réalisée par le co-clustering.

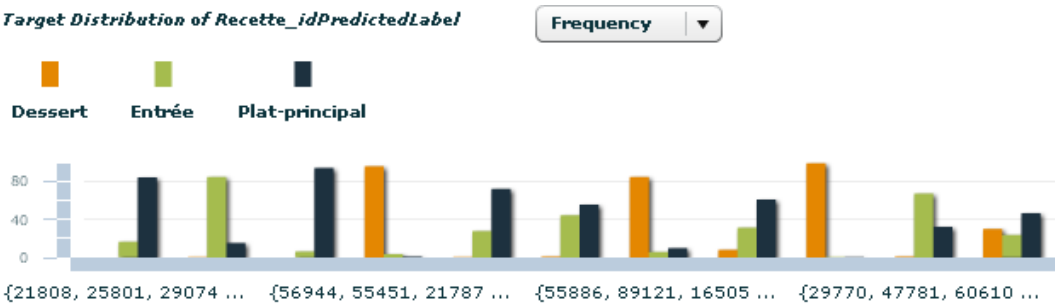

FIGURE 6 – Distribution de clusters

3.2.3 Essai 3

Le 3ème essai reprend le même algorithme que l'essai 2, mais le co-clustering a été réalisé non pas sur les données d'apprentissage mais sur l'ensemble des données d'apprentissage et de test, c'est à dire sur environ 16000 recettes.

4 Tâche 3

Cette tâche a été l'occasion de tester une forme de distance sémantique mot/mot généralisée de manière simple à une distance document/document. Nous nous sommes placés dans un cadre applicatif dans lequel les documents courts (titres) sont en quantité limitée mais gagnent à être ordonnés de manière pertinente.

4.1 Prétraitements

Pour cette tâche, l'espace de représentation des documents a été modifié. Nous avons utilisé notre outil de Traitement Automatique des Langues, TiLT (Heinecke *et al.*, 2008), pour effectuer une lemmatisation associée à un étiquetage syntaxique des documents. Certains traitements ont été réalisés pour tous les types de champs (titre, ingrédients, préparation). Les termes liés aux catégories suivantes été retirés pour tous les types de champs : pronoms, prépositions, déterminants, conjonctions, adverbes, interjections, ainsi que les quantités numériques de tous types. Dans tous les cas, nous avons aussi cumulé, dans un même vecteur, les formes obtenues ainsi que les lemmes correspondants, de manière unitaire, afin d'obtenir une liste de termes uniques.
Nous avons ensuite réalisé des filtrages différenciés en fonction des types de champs (titre, ingrédients, préparation). En ce qui concerne les titres, seules les formes ont été conservées, pas leurs lemmes. Pour les ingrédients, ce sont les formes et les lemmes qui ont été retenus. Les préparations sont aussi représentées par leurs formes et leurs lemmes mais elles ont subi un filtrage supplémentaire puisque nous avons aussi ôté les adjectifs et les verbes. En effet, le rapprochement entre titres et recettes ne fait pas ou peu intervenir adjectifs et verbes. Toutefois, nous avons conservé les adjectifs et les verbes des titres et des ingrédients car le contexte de lemmatisation générait des erreurs du type nom/adjectif telle que « saumon » qui aurait été filtré suite à une erreur l'étiquetant comme « adjectif ». Au final, les ingrédients et les préparations correspondantes sont regroupés en un seul vecteur constitué d'éléments uniques. L'utilisation des lemmes est surtout destinée à étendre la représentation de la recette afin d'augmenter l'intersection entre le vecteur ingrédients-préparation et les mots des titres. Les multi-mots détectés ne sont pas utilisés mais décomposés en mots simples. Le résultat de ce traitement appliqué, par exemple, à la recette « Risotto de quinoa aux champignons » donne la représentation suivante :
Titre : *risotto quinoa champignons*
Document (ingrédients et préparation) : c*hampignons champignon frais quinoa noix oignon sel poivre poivrer persil morceaux morceau sauteuse sauteur huile eau temps risotto poêle bouillon sauce*

Cet exemple montre un taux de recouvrement des mots du titre avec le contenu de la recette de 100%.

Cette représentation a été testée parmi d'autres et s'est avérée nécessaire et suffisante pour cette tâche. Nous limitons ainsi la taille des vecteurs à comparer ce qui limite le temps de calcul notamment le temps nécessaire pour appliquer la mesure de distance présentée au chapitre suivant.

4.2 Distance sémantique

Nous avons utilisé et adapté la mesure proposée par (Cilibrasi et Vitanyi, 2006). Pour rappel, si x et y sont deux termes, f(x) la fréquence de x , f(y)la fréquence de y, f(x,y) la fréquence de x ET y, et M une valeur de normalisation (nombre de documents), la « distance » est la suivante :

$$D(x,y) = \frac{max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log M - min\{\log f(x), \log f(y)\}}$$

Cette mesure vérifie les deux premiers critères d'une distance mathématique (symétrie et séparation) mais pas le critère d'inégalité triangulaire :

D(x,y) = D(y,x) (symétrie)
D(x,y) = 0 ⇔ x=y (séparation)
D(x,z) <= D(x,y)+D(y,z) non vérifié

Cette mesure donne donc une forme de proximité sémantique qui varie entre 0 et 1, 0 indiquant que x est « sémantiquement » proche de y et 1 que x est « sémantiquement très éloigné de y ». Les logarithmes et la constante de normalisation donnent une valeur qu'il faut plutôt interpréter comme une relation d'ordre : x est plus proche de y que de z.

Cette mesure a été initialement proposée en utilisant les comptages fournis par Google. Pour une utilisation concrète faisant appel à un accès intensif à des centaines de millions de distances, l'utilisation de requêtes vers Google n'est pas possible. Nous avons choisi de réaliser ce calcul à partir des documents de Wikipédia français. Une fois le calcul terminé, nous disposons d'une ressource de « distances sémantiques entre termes » accessible en temps réel.

Ce calcul a été réalisé en 2010 à partir d'une version de Wikipédia datant de fin 2009. Les documents utilisés sont les paragraphes issus naturellement des pages de Wikipédia lors du filtrage des méta-données de formatage des pages de Wikipédia. Le calcul fréquentiel f(x,y) est donc effectué dans le contexte d'une phrase ou d'un paragraphe. Après traitement, nous avons extrait environ 6 000 000 paragraphes et constitué une matrice de 193 999 258 distances associées à autant de couples de mots simples. La matrice utilisée ici n'intègre ni les entités nommées, ni les multi-mots.

C'est cette ressource que nous avons voulu tester sur la tâche 3, l'idée étant de calculer une distance globale entre chaque titre et chaque recette par le biais des distances individuelles entre chaque mot des titres et de la recette. Ces distances étant issues d'une ressource initialement indépendante de la tâche mais relativement importante par sa taille et la variété de ses contenus, il est intéressant de pouvoir la tester dans un domaine de spécialité.

Autre point important, ce type de mesure permet d'ordonner la liste des titres en fonction des termes de la recette même si les termes du titre n'apparaissent pas parmi les termes de la recette. Il suffit qu'un terme du titre possède une entrée dans notre matrice en relation avec un mot de la recette pour avoir une distance utilisable.

En ce qui concerne la recette du « Risotto de quinoa aux champignons », nous pouvons ainsi potentiellement récupérer les distances entre les mots du titre (*risotto quinoa champignons)* et tous les mots de la recette (c*hampignons champignon frais quinoa noix oignon sel poivre*

poivrer persil morceaux morceau sauteuse sauteur huile eau temps risotto poêle bouillon sauce).

Nous pouvons donc réaliser ce calcul pour chaque couple (titre, recette). Si T est le nombre de termes du titre et R le nombre de termes de la recette, le nombre de distances à considérer est T×R. Nous avons ensuite choisi une métrique globale faisant intervenir les distances unitaires et synthétisant la proximité entre deux documents, en l'occurence un titre et une recette. Des essais préliminaires ont montré qu'il était plus pertinent de retenir, pour chaque terme d'un titre, uniquement le terme de la recette le plus proche, donc de distance minimum. En ce qui concerne la recette précédente, nous obtenons donc naturellement une distance minimum de 0 pour les termes communs au titre et à la recette :

D_{min}(risotto,risotto) = 0.0
D_{min}(quinoa,quinoa) = 0.0
D_{min}(champignons,champignons) = 0.0

Pour d'autres titres, nous obtenons des valeurs comprises en 0 et 1 pour chaque terme en relation avec l'élément le plus proche de la recette. Par exemple pour le titre représenté par (gâteau chocolat noix coco) :

D (gâteau,*noix*) = min{ D (gâteau,*m*)| *m* mot de la recette}
= 0.182777617404
D (chocolat,*noix*) = min{ D (chocolat,*m*)| *m* mot de la recette}
= 0.192853987214
D (noix,*noix*) = min{ D (noix,*m*)| *m* mot de la recette}
= 0.0 (terme commun)
D (coco,*noix*) = min{ D (coco,*m*)| *m* mot de la recette}
= 0.131960109813

Pour le titre représenté par (gratin courgettes pain ail)

D (gratin, *champignons*) = min{ D (gratin,*m*)| *m* mot de la recette}
= 0.200901512394
D (courgettes, *sauteur*) = min{ D (courgettes,*m*)| *m* mot de la recette}
= 0.190081305148
D (pain, *persil*,) = min{ D (pain,*m*)| *m* mot de la recette}
= 0.17466164279
D (ail, *oignon*) = min{ D (ail,*m*)| *m* mot de la recette}
= 0.0799472060861

Ces valeurs sont donc le reflet du calcul issu du corpus de Wikipédia. Si on peut établir facilement le lien (ail, oignon) ou (coco, noix) il est plus difficile d'expliquer la relation (courgettes, sauteur) !

La mesure globale que nous avons ensuite retenue est la moyenne de ces distances minimales. Si T est un titre de N termes et R une recette, la distance globale est :

$$D(T,R) = \frac{1}{N} \sum_i^j D_{min}(T_i, R_j)$$

Nous obtenons une mesure globale de proximité qui vaut 0 si les termes du titre sont

contenus dans les termes de la recette. Cette mesure tend vers 1 si aucun des termes du titre n'est contenu dans la recette et si de plus ces termes sont « éloignés » des termes de la recette. La moyenne permet de normaliser la mesure par rapport à la longueur du titre. Nous avons bien une mesure qui permet d'ordonner les titres de manière relative dans tous les cas, même si aucun terme du titre n'apparaît dans la recette, ou presque… En effet, si cela est vrai pour la majorité des titres, certains titres dont les termes ne sont jamais apparus dans Wikipédia ne possèdent aucun lien avec d'autres termes. Un exemple est le titre « Tiou Yap ». Dans ce cas la distance à chaque recette est de 1 ce qui reste cohérent.

Les 3 essais portant sur cette tâche sont des variantes de cette mesure :

- le premier essai n'utilise pas la matrice de distance issue de Wikipédia mais fixe à 0 la distance d'un terme du titre apparaissant dans la recette et à 1 s'il n'apparaît pas. C'est une sorte de limite binaire par défaut de la distance utilisée.

- le deuxième essai utilise la métrique décrite calculée à partir de Wikipédia français.

- le troisième essai utilise aussi la métrique décrite mais nécéssite deux matrices. La première est toujours celle issue de Wikipédia français. La deuxième a été calculée à partir du corpus d'apprentissage de la tâche. Toutes les recettes (titre, ingrédients, préparation) ont servi à calculer une matrice de distances mot/mot relative à ces données. Au final, nous utilisons une forme de vote qui consiste à choisir la distance minimale issue de l'une ou l'autre des deux matrices. Nous faisons l'hypothèse que la matrice calculée sur les recettes des données d'apprentissage est plus spécifique et pertinente que celle de Wikipédia mais moins complète.

Pour des raisons de temps les essais 2 et 3 ont été appliqués sur les 50 premiers résultats de l'essai 1 ce qui correspond plutôt à un ré-ordonnancement du premier essai.

5 Résultats et discussions

5.1 Tâche 1

Pour notre premier essai, utilisant le classifieur de type Maximum d'Entropie, les résultats n'ont pas été satisfaisants.

Concernant l'essai 2 utilisant Khiops, notre 2ème place semble confirmer les qualités de cet outil. Khiops gère plutôt bien cette tâche que nous estimions difficile compte tenu du type et des quantités relatives des données d'apprentissage. Une technique discriminante appliquée à une sélection de variables effectuée par Khiops aurait peut être pu nous hisser plus haut !

5.2 Tâche 2

Pour cette tâche, deux constats sont à faire. Tout d'abord, les résultats des deux premiers essais sont assez proches avec des techniques différentes. L'algorithme de type Maximum d' Entropie s'en sort plutôt bien avec le jeu de données d'apprentissage fourni.

Par contre, la puissance de modélisation de Khiops ne semble pas avoir permis de surpasser les algorithmes concurrents. Nous sommes un peu surpris du positionnement, compte tenu des résultats habituels de Khiops. Son utilisation sur des variables majoritairement textuelles, ce qui n'est pas son domaine applicatif standard, nécessite peut être des avancées

algorithmiques ou bien des prétraitements amont plus adaptés.

Pour le 3ème essai nous avons réalisé un co-clustering sur les données d'apprentissage et de test. Cette technique améliore généralement, au moins légèrement, les performances d'un co-clustering réalisé uniquement sur les données d'apprentissage (essai 2). Compte-tenu des résultats inférieurs du 3ème essai, nous suspectons un problème technique lors de notre essai.

Toujours est-il que Khiops nous donne une vision synthétique des données qui peut être utile dans d'autres cadres applicatifs : sous-catégorisation plus spécialisée des types de plats ou rapprochement de recettes par rapport à peu d'exemples de recettes prototypiques. Il nous a aussi permis d'analyser rapidement le chevauchement des classes *entrée*/*plat* et des variables associées.

5.3 Tâche 3

Pour cette tâche, nous avions choisi de tester une métrique particulière. À la vue de nos résultats, il semblerait que ce choix soit pertinent. Mais si nous regardons de plus près les résultats des différents essais, ce choix est moins évident. En effet, la distance binaire naïve du 1er essai produit un meilleur résultat que le 2ème essai qui utilise les distances mot/mot générales issues de Wikipédia. Il semblerait donc que la prise en compte de ces valeurs réelles entre 0 et 1 agisse plutôt comme du bruit. On peut aussi penser que notre stratégie qui a consisté à utiliser un pré-ordonnancement des cinquante premiers titres avec une métrique naïve, puis à réordonner ces titres avec une deuxième métrique élimine des bonnes solutions qui sont en quelques sorte oubliées…

Le 3ème essai qui donne les meilleurs résultats en utilisant l'information spécifique au domaine (corpus d'apprentissage) nous encourage toutefois à continuer dans l'exploration de cette voie.

6 Conclusion

Cette participation à DEFT 2013 nous a permis de tester des algorithmes développés par Orange Labs sur des données qui peuvent paraître loin de nos préoccupations habituelles. La spécificité des problèmes de classification des tâches 1 et 2 nous intéressait justement par leur production peu classique. L'outil testé (Khiops) montre ses capacités à s'adapter à de nombreuses tâches et particulièrement ici à des variables majoritairement textuelles. Il a montré ses performances en classification supervisée sur la tâche 1. Il se montre un peu moins performant sur la tâche 2, bien que présentant des résultats tout à fait honorables. Ses capacités intégrées d'analyse non supervisée en font un outil globalement très performant et utile pour de nombreuses tâches. La tâche 3 a été l'occasion d'expérimenter une métrique document/document basée sur une métrique individuelle mot/mot. Ces premiers essais nous paraissent intéressants et demandent à être confirmés sur d'autres types de données, particulièrement sur des documents possédant peu de mots communs. Finalement, la tâche 3 s'est avérée être presque un cas d'usage que nous allons exploiter dans le cadre du « AI Mashup Challenge ».

Remerciements

Nous remercions Marc Boullé pour ses conseils sur les tâches de classification ainsi que pour son aide avec Khiops.

Références

BELAUNDE, M., PINSON F., COLLIN, O. (2013). Cooking Assistant mashup. *AI Mashup Challenge (ESCW 2013),* Accepted

BOULLÉ, M. (2007). Compression-Based Averaging of Selective Naive Bayes Classifiers. *Journal of Machine Learning Research,* 8:1659-1685.

BOULLÉ, M. (2008). Khiops: outil de préparation et modélisation des données pour la fouille des grandes bases de données. *In Extraction et gestion des connaissances, EGC'2008,* Sophia-Antipolis, France

BOULLÉ, M. (2011). Data grid models for preparation and modeling in supervised learning. *In Hands-On Pattern Recognition: Challenges in Machine Learning, volume 1, I. Guyon, G. Cawley, G. Dror, A. Saffari (eds.),* pages 99-130*,* Microtome Publishing.

CILIBRASI, R., VITANYI, P. (2006). Similarity of Objects and the Meaning of Words. *In Proceedings of the Third international conference on Theory and Applications of Models of Computation, TAMC'06*, Beijing, China, pages 21-45.

CILIBRASI, R., VITANYI, P. (2007). The Google similarity distance. *In IEEE Transactions on Knowledge and Data Engineering,* 19(3), 370–383 (2007). Preliminary version: "Automatic Meaning Discovery Using Google", Arxiv preprint cs.CL/0412098, 2004, arxiv.org

CILIBRASI, R., BALBACH, F. J., VITANYI, P., LI, M. (2009). Normalized Information Distance *Information Theory and Statistical Learning (*Franck Emmert-Streib, Matthias Dehmer) ISBN: 978-0-387-84815-0 (Print) 978-0-387-84816-7 (Online), pages 45-82, Chapitre 3

HEINECKE, J., SMITS, G., CHARDENON, C., GUIMIER DE NEEF, E., MAILLEBUAU, E., BOUALEM, M. (2008). TiLT : plate-forme pour le traitement automatique des langues naturelles. *Traitement automatique des langues ,* 49(2):17-41.

NIGAM, K., LAFFERTY, J., MCCALLUM, A. (1999). Using Maximum Entropy for text classification. *In Proceedings of the IJCAI Workshop on Information Filtering*, pages 61-67

TSURUOKA, Y., TSUJII, J., ANANIADOU, S. (2009). Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty. *In Proceedings of ACL-IJCNLP*, pages 477-485