Classification multi-label par fonctions de croyance

Multilabel classification using belief functions

R. Nassif[1,2] S. Destercke[1] M.H. Masson[3]
[1] Université de Technologie de Compiègne
[2] Université Libanaise
[3] Université de Picardie Jules Verne
Laboratoire Heudiasyc, UMR CNRS 7253, BP 20529 60205 Compiègne, roula.nassif@etu.utc.fr

Résumé :
La classification multi-label consiste à associer simultanément à chaque individu x une ou plusieurs étiquettes. L'ordonnancement d'étiquettes est un problème d'apprentissage dont le but est de relier des instances à un ordre total défini sur un ensemble d'étiquettes possibles. Une technique de passage de l'ordonnancement d'étiquettes à la classification multi-label a été developpée dans la littérature. En se basant sur cette technique, nous utilisons la théorie des fonctions de croyance afin d'élaborer une nouvelle méthode de classification multi-label. Afin d'évaluer notre approche, nous comparons les résultats obtenus sur les jeux de données classiques avec ceux obtenus par d'autres méthodes de la littérature.

Mots-clés :
Classification multi-label, ordonnancement d'étiquettes, fonctions de croyance, k-PPV évidentiels, étiquette calibrée, comparaison par paires.

Abstract:
In multilabel classification, the goal is to assign one or more labels to each instance x. Label ranking is a learning task where the goal is to map instances to a linear order on a finite set of predefined labels. An approach was developed in the literature to move from label ranking to multilabel classification. Based on this technique, we use the theory of belief functions to develop a new method for multilabel classification problem. To evaluate our approach, we compare the results with those obtained by other methods in the literature.

Keywords:
Multilabel classification, label ranking, belief functions, evidential k-NN, calibrated label, pairwise comparison.

1 Introduction

Ces dernières années ont vu émerger différentes extensions du problème de classification classique : parmi ces dernières se trouvent les problèmes de classification multi-label et les problèmes d'ordonnancement d'étiquettes. Soit $\pounds$ un ensemble fini d'étiquettes $\{\lambda_1,\lambda_2,...\lambda_l\}$ et soient X l'espace d'entrée et Y l'espace de sortie formé des sous-ensembles de $\pounds$. Sur la base d'un ensemble d'apprentissage $D = \{(x_i, y_i)/x_i \in X, y_i \in Y\}$ formé de n instances étiquetées, on recherche un classifieur permettant d'associer à toute instance $x \in X$ le y minimisant un certain critère d'erreur. Dans la *classification classique*, l'espace de sortie est formé par des singletons c.à.d. qu'une seule étiquette parmi les l est associée à chaque instance x. Dans la *classification multi-label*, une ou plusieurs étiquettes sont associées simultanément à x. Dans le cas du problème d'*ordonnancement d'étiquettes*, il s'agit d'apprendre pour chaque instance x un ordre total $\succ$ sur l'ensemble d'étiquettes $\pounds$. Par exemple, prenons le cas d'un document et supposons qu'on désire le classer selon deux critères, le premier étant la langue et le second le champ d'application. Etant donné un ensemble de trois langues possibles {française, anglaise, chinoise}, à tout document on ne peut associer qu'une seule étiquette parmi les trois étiquettes possibles. Ce type de problème est traité par la classification classique. Par contre, pour un ensemble de champs d'application *(mathématiques, physique, chimie, biologie, informatique)* $\{M, P, C, B, I\}$, un document peut relever de plusieurs catégories comme les mathématiques, physique et informatique, soit $y = \{M, P, I\}$, donc pour une instance on peut associer une ou plusieurs étiquettes. Finalement, du point de vue du problème d'ordonnancement, on peut associer à tout document un ordre sur l'ensemble des champs d'application. Par exemple un docu-

ment x peut appartenir plus à la catégorie M que P que I ... En utilisant l'écriture $\lambda_i \succ_x \lambda_j$ qui signifie que, pour une instance x, l'étiquette λ_i est préférée à l'étiquette λ_j, le résultat aura, par exemple, la forme suivante : $M \succ_x P \succ_x I \succ_x C \succ_x B$.

Ces deux problèmes ont de nombreuses applications. Le problème de classification multi-label est rencontré, par exemple, dans la caractérisation d'une image (qui peut à la fois contenir une ville, une montagne, une plage, ...), d'une musique (qui peut être à la fois douce, calmante, enthousiasmante) ou d'un film (qui peut appartenir à plusieurs genres). Le problème d'ordonnancement est rencontré, par exemple, dans la recommandation d'articles selon les préférences des consommateurs, l'évaluation du profil d'expression d'un gène (où les forces d'expression peuvent se voir comme un ordonnancement).

Nous traitons dans cet article du problème de classification multi-label en adoptant l'approche proposée par Fürnkranz et al. [3]. Cette approche appelée ordonnancement calibré, permet de transformer tout algorithme résolvant un problème d'ordonnancement en méthode pour la classification multi-label. Nous utilisons comme algorithme d'ordonnancement celui proposé par Masson et al. [5]. Ce dernier, fondé sur la théorie des fonctions de croyance, permet d'obtenir l'ordonnancement le plus plausible à partir d'informations fournies par différents classifieurs binaires.

Afin de présenter les résultats de la classification multi-label par ordonnancement calibré dans le cadre de la théorie des fonctions de croyance, nous proposons d'organiser cet article de la manière suivante : tout d'abord, nous commençons par quelques brefs rappels sur la théorie des fonctions de croyance, puis nous rappelons la méthode de l'ordonnancement d'étiquettes par fonctions de croyance introduite dans [5]. Ensuite, nous présentons la méthode de classification multi-label par ordonnancement calibré [3]. Dans le paragraphe 5, nous évaluons notre méthode en comparant les résultats obtenus sur des jeux de données classiques avec d'autres méthodes.

2 Quelques rappels sur la théorie des fonctions de croyance

La théorie des fonctions de croyance (ou théorie de Dempster-Shafer) [8] généralise à la fois la théorie des probabilités (conditionnement, marginalisation) et les approches ensemblistes (intersection, union, inclusion, etc.). Soit $\Omega = \{\omega_1, \omega_2, ..., \omega_k\}$ (cadre de discernement) un ensemble fini de réponses possibles à une question. Une fonction de masse sur Ω est une application $m : 2^\Omega \rightarrow [0, 1]$ telle que :

$$\sum_{A \subseteq \Omega} m(A) = 1. \quad (1)$$

Tout sous-ensemble A de Ω tel que $m(A) > 0$ représente un ensemble possible de valeurs pour ω, et la quantité $m(A)$ peut être interprétée comme la mesure de croyance exactement allouée à $\omega \in A$. A partir de m, on peut définir d'autres fonctions :

– Fonction de plausibilité : elle représente la partie maximale de croyance qui pourrait soutenir A.

$$pl(A) = \sum_{B \cap A \neq \emptyset} m(B) \quad \forall A \subseteq \Omega. \quad (2)$$

– Fonction de commonalité : elle représente la somme des masses allouées aux sur-ensembles de A, $q(\emptyset) = 1$,

$$q(A) = \sum_{B \supseteq A} m(B) \quad \forall A \subseteq \Omega. \quad (3)$$

Soient Ω et Θ deux cadres de discernement. On appelle *raffinement* de Θ vers Ω l'application $\rho : 2^\Theta \rightarrow 2^\Omega$ telle que :

– l'ensemble $\{\rho(\{\theta\}), \theta \in \Theta\} \subseteq 2^\Omega$ est une partition de Ω ;

– $\rho(A) = \cup_{\theta \in \Theta} \rho(\{\theta\}) \; \forall A \subseteq \Theta$

Θ est appelé un grossissement de Ω et Ω un raffinement de Θ. Pour transférer une masse m^Θ d'un grossissement Θ vers Ω, on utilise

l'opération d'extension vide suivante :

$$m^{\Theta\uparrow\Omega}(B) = \begin{cases} m^{\Theta}(A) & si \quad B = \rho(A), \\ 0 & \text{sinon.} \end{cases} \tag{4}$$

Soient m_1 et m_2 deux fonctions de masse sur Ω modélisant deux éléments d'évidence, on utilise la règle de combinaison conjonctive pour les combiner. La fonction de masse résultante est, $\forall A \subseteq \Omega$:

$$(m_1 \oplus m_2)(A) = \sum_{B \cap C = A} m_1(B) m_2(C). \tag{5}$$

Cette règle peut s'exprimer facilement à l'aide des commonalités, $\forall A \subseteq \Omega$:

$$(q_1 \oplus q_2)(A) = q_1(A) q_2(A). \tag{6}$$

A noter que, avant d'appliquer cette règle de combinaison, il faut ramener les informations sur le même cadre de discernement.

3 Ordonnancement d'étiquettes par fonctions de croyance

3.1 Ordonnancement d'étiquettes

Le problème d'ordonnancement d'étiquettes consiste à apprendre, à partir d'exemples, une application associant à toute instance $x \in X$, un ordre total $\succ_x$ sur $\pounds = \{\lambda_1, \lambda_2, ..., \lambda_l\}$. L'ordre total $\succ_x$ peut être représenté de manière équivalente par une permutation τ_x de l'ensemble des entiers $\{1, 2, ..., l\}$ de telle sorte que $\lambda_i \succ_x \lambda_j \Leftrightarrow \tau_x(i) < \tau_x(j)$ (la valeur $\tau_x(i)$ représente le rang de λ_i dans les préférences de x). Pour résoudre ce problème, différentes méthodes existent. Une approche particulière est celle dite de ***préférence par paires*** [2]. Elle consiste à apprendre, pour chaque couple d'étiquettes (λ_i, λ_j) tels que $i < j$, un classifieur binaire M_{ij} permettant de prédire pour une entrée x si λ_i est préférée ou non à λ_j. Pendant la classification, chaque instance x est alors soumise aux $l(l-1)/2$ classifieurs, la sortie du classifieur étant 1 si $\lambda_i \succ \lambda_j$ et 0 dans le cas contraire. A noter que n'importe quel classifieur binaire peut être utilisé dans ce cas, et que la sortie n'est pas nécessairement dans $\{0, 1\}$, elle est généralement comprise dans l'intervalle $[0, 1]$. Hüllermeier propose alors d'associer à chaque entrée x une relation floue de préférence R_x :

$$R_x(\lambda_i, \lambda_j) = \begin{cases} M_{ij}(x) & \text{si} \quad i < j, \\ 1 - M_{ij}(x) & \text{si} \quad i > j. \end{cases} \tag{7}$$

Moyennant ces relations de préférences, on calcule pour chaque étiquette λ_i, la fonction score $S_x(\lambda_i)$:

$$S_x(\lambda_i) = \sum_{j \neq i} R_x(\lambda_i, \lambda_j). \tag{8}$$

Pour une instance x, l'ordre total est obtenu en triant par ordre décroissant les fonctions $S_x(\lambda_i)$ calculées pour chaque élément de $\pounds$.

3.2 Ordonnancement d'étiquettes par fonctions de croyance

Plusieurs méthodes ont été proposées dans la littérature pour traiter la problématique d'ordonnancement d'étiquettes. Dans cet article, nous adoptons l'approche ***d'ordonnancement d'étiquettes dans le cadre des fonctions de croyance*** [5].

Chaque classifieur binaire évidentiel M_{ij} travaille sur un grossissement particulier Θ_{ij} de l'ensemble S contenant toutes les permutations possibles de $\pounds = \{1, 2, ..., l\}$. Nous avons $\Theta_{ij} = \{\theta_{ij}, \overline{\theta}_{ij}\}$ un grossissement binaire avec θ_{ij} l'ensemble des permutations $\tau \in S$ pour lesquelles λ_i est préférée à λ_j et $\overline{\theta}_{ij}$ l'ensemble des permutations $\tau \in S$ pour lesquelles λ_j est préférée à λ_i. On suppose que chaque classifieur évidentiel fournit, pour chaque instance x, la fonction de masse suivante :

$$\begin{cases} m_x^{\Theta_{ij}}(\theta_{ij}) = \alpha_{ij}, \\ m_x^{\Theta_{ij}}(\overline{\theta_{ij}}) = \beta_{ij}, \\ m_x^{\Theta_{ij}}(\Theta_{ij}) = 1 - \alpha_{ij} - \beta_{ij}. \end{cases} \tag{9}$$

A partir de ces masses, on en déduit les plausi-

bilités suivantes :

$$\begin{cases} pl_x^{\Theta_{ij}}(\theta_{ij}) = 1 - \beta_{ij}, \\ pl_x^{\Theta_{ij}}(\overline{\theta_{ij}}) = 1 - \alpha_{ij}. \end{cases} \quad (10)$$

Ces valeurs correspondent aux commonalités, donc on a :

$$\begin{cases} q_x^{\Theta_{ij}}(\theta_{ij}) = 1 - \beta_{ij} \\ q_x^{\Theta_{ij}}(\overline{\theta_{ij}}) = 1 - \alpha_{ij}. \end{cases} \quad (11)$$

Un ordre total sur les étiquettes s'obtient finalement par la méthode du *maximum de plausibilité*. Ayant $l(l-1)/2$ sources d'informations exprimées sur des référentiels différents, on cherche une fonction de masse $m_x^S(.)$ exprimée sur S. La fonction de masse totale est obtenue en combinant ces différentes masses étendues à S grâce à l'opération d'extension vide :

$$m_x^S(.) = \oplus_{i<j} m_x^{\Theta_{ij}\uparrow S}(.). \quad (12)$$

En utilisant les opérations sur les fonctions de croyance, la commonalité associée à une permutation τ de S peut s'écrire :

$$q_x(\tau) = \prod_{\tau(i)<\tau(j)} (1-\beta_{ij}) \prod_{\tau(k)>\tau(l)} (1-\alpha_{kl}), \quad (13)$$

où la notation $\tau(i) < \tau(j)$ en indice désigne l'ensemble des couples (λ_i, λ_j), $i<j$ pour lesquels λ_i est placée avant λ_j. La permutation τ étant un élément singleton de S, cette valeur n'est rien d'autre que la plausibilité $pl_x(\tau)$. Le choix d'un ordre particulier dans S peut consister à chercher la permutation τ^* de plus grande plausibilité. Pour maximiser $pl_x(\tau)$, il est équivalent de maximiser son logarithme. La permutation optimale s'exprime donc sous la forme suivante :

$$\begin{array}{c} \tau^* = \arg\max_\tau \sum_{\tau(i)<\tau(j)} \ln(1-\beta_{ij}) \\ + \sum_{\tau(k)>\tau(l)} \ln(1-\alpha_{kl}). \end{array} \quad (14)$$

Pour déterminer τ^*, on introduit pour chaque classifieur binaire une variable x_{ij} telle que :

$$x_{ij} = \begin{cases} 1 & \text{si} \quad \lambda_i \succ \lambda_j \quad \text{c.à.d.} \quad \tau(\lambda_i) < \tau(\lambda_j), \\ 0 & \text{si} \quad \lambda_j \succ \lambda_i \quad \text{c.à.d.} \quad \tau(\lambda_j) < \tau(\lambda_i). \end{cases}$$

On calcule les x_{ij} optimaux par résolution d'un programme linéaire en nombres entiers :

$$\begin{array}{c} \max_{x_{ij}\in\{0,1\}} \sum_{i<j} x_{ij} \ln(1-\beta_{ij}) \\ + \sum_{i<j} (1-x_{ij}) \ln(1-\alpha_{ij}). \end{array} \quad (15)$$

sous les contraintes :

$$\begin{cases} x_{ij} + x_{jk} - 1 \leq x_{ik} \quad \forall \quad i<j<k, \\ x_{ik} \leq x_{ij} + x_{jk} \quad \forall \quad i<j<k. \end{cases} \quad (16)$$

Ces contraintes permettent d'assurer la transitivité de la relation d'ordre recherchée. Finalement, à partir des valeurs x_{ij}, on peut déterminer la permutation optimale :

$$\begin{cases} x_{ij} = 1 \quad \Rightarrow \quad \tau^*(i) < \tau^*(j), \\ x_{ij} = 0 \quad \Rightarrow \quad \tau^*(i) > \tau^*(j). \end{cases} \quad (17)$$

4 Classification multi-label par ordonnancement calibré

Nous exposons une approche pour résoudre le problème de ***multi-label calibré*** qui associe à toute entrée $x \in X$, caractérisée par des attributs, un ensemble d'étiquettes pertinentes $P_x \subseteq \mathcal{L} = \{\lambda_1, \lambda_2, ..., \lambda_l\}$ déduit à partir d'un ordre sur $\mathcal{L}$.

4.1 Classification multi-label comme problème d'ordonnancement

Dans la classification multi-label, chaque exemple x est associé à un ensemble d'étiquettes pertinentes P_x et implicitement à un ensemble d'étiquettes non pertinentes $N_x = \mathcal{L} \setminus P_x$. Ce problème peut être vu comme un cas particulier de l'ordonnancement d'étiquettes. Obtenir P_x et N_x à partir d'un ordre $\succ$ sur $\mathcal{L}$ revient à choisir une étiquette λ_i et à considérer la séparation $P_x = \{\lambda_j / \ \lambda_j \succ \lambda_i\}$ et $N_x = \{\lambda_j / \ \lambda_i \succeq \lambda_j\}$. Pour résoudre le problème du choix de λ_i, Fürnkranz et al. [3] proposent d'ajouter une étiquette pivot λ_0 à $\mathcal{L}$.

Soit S^0 l'espace des permutations possibles de l'ensemble $\mathcal{L}'$ avec $\mathcal{L}' = \mathcal{L} \cup \lambda_0$ ($\mathcal{L}' = \{\lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_l\}$). Le modèle $h : \boldsymbol{X} \rightarrow S^0$ associant à tout exemple un ordre sur $\mathcal{L}'$ est appelé ordonnancement calibré. Le pivot λ_0 est ensuite utilisé pour séparer les étiquettes pertinentes (P_x) de celles non pertinentes.
Le problème est alors réduit à un problème d'ordonnancement de $l+1$ étiquettes. Le résultat

étant de la forme :

$$\lambda_{i_1} \succ \dots \lambda_{i_j} \succ \lambda_0 \succ \lambda_{i_{j+1}} \succ \dots \lambda_{i_l}. \quad (18)$$

Ceci induit à la fois un ordonnancement

$$\lambda_{i_1} \succ \dots \lambda_{i_j} \succ \lambda_{i_{j+1}} \succ \dots \lambda_{i_l}, \quad (19)$$

et une partition bipartite :

$$\begin{cases} P_x = \{\lambda_{i_1}, \dots, \lambda_{i_j}\}, \\ N_x = \{\lambda_{i_{j+1}}, \dots, \lambda_{i_l}\}. \end{cases} \quad (20)$$

4.2 Transformation de données multi-label en données d'ordonnancement

Un individu de l'ensemble d'apprentissage d'un problème d'ordonnancement est une instance $x \in X$, caractérisée par des attributs, et sa sortie y constituée d'une relation de préférence $R_x = \{(\lambda, \lambda')/\lambda \succ_x \lambda'\}$. Dans un problème de classification multi-label, deux ensembles N_x et P_x sont associés à chaque instance d'apprentissage x. A partir de ces ensembles, on peut facilement construire l'ensemble de préférences $R_x = \{(\lambda, \lambda')/\lambda \in P_x, \lambda' \in N_x\}$. En introduisant le pivot λ_0, l'ensemble de préférences devient alors :

$$R'_x = R_x \cup \{(\lambda, \lambda_0)/\lambda \in P_x\} \cup \{(\lambda_0, \lambda)/\lambda \in N_x\}$$

Nous traitons dans cette section la transformation d'un ensemble d'apprentissage multi-label vers un ensemble d'apprentissage traitant un problème d'ordonnancement. Afin de rendre plus claire la méthodologie, nous utilisons un exemple. Supposons que l'ensemble d'apprentissage associé au problème de classification multi-label d'un document *(paragraphe 1)* est formé de trois instances :

$x_1 : P_{x_1} = \{M, P, I\}$;
$x_2 : P_{x_2} = \{C, B\}$;
$x_3 : P_{x_3} = \{M\}$.

Comme nous l'avons vu dans la partie précédente, afin de résoudre le problème de classification multi-label, il faut ajouter un pivot λ_0 à $\pounds = \{M, P, C, B, I\}$. Suivant le paragraphe 3.1, les données d'apprentissage sont ensuite transformées pour obtenir un jeu de données par paire d'étiquette. Nous devons alors construire 15 classifieurs binaires M_{ij} ($M_{0M}, M_{0P}, \dots, M_{MP}, M_{MC}, \dots, M_{BI}\}$). Par exemple, pour le classifieur M_{0M}, pour x_1 et x_3, l'étiquette M appartient à P_{x_1} et P_{x_3}, d'où $M \succ_{x_1} \lambda_0$ et $M \succ_{x_3} \lambda_0$, cependant, pour x_2, l'étiquette M n'appartient pas à P_{x_2}, d'où $\lambda_0 \succ_{x_2} M$. x_1 et x_3 seront donc utilisés pour apprendre M_{0M}, mais pas x_2. Pour le classifieur M_{MP}, pour x_1, on ne sait pas laquelle des étiquettes $\{M, P\}$ est préférée, alors nous ne prenons pas en considération cet exemple dans la construction de la base d'apprentissage de ce classifieur. Voici quelques transformations :

Tableau 1 – Classifieur M_{0M}

	$\lambda_0 \succ_x M$	$M \succ_x \lambda_0$
x_1	0	1
x_2	1	0
x_3	0	1

Tableau 2 – Classifieur M_{MP}

	$M \succ_x P$	$P \succ_x M$
x_3	1	0

5 Etude expérimentale

5.1 Jeux de données utilisés

Pour tester notre méthode, nous utilisons trois bases de données {*emotions, scene, yeast*}, téléchargées de la librairie *MULAN* (http ://mlkd.csd.auth.gr/multilabel.html). Les données *emotions* portent sur la classification des chansons selon les émotions qu'elles évoquent, le jeu *scene* porte sur l'indexation sémantique de scènes et, finalement, la base *yeast* contient des données concernant des analyses fonctionnelles des gènes. Les caractéristiques de ces jeux sont données dans le

tableau 3, et tous les résultats fournis dans ce papier ont été calculés sur les examples de test (fournis dans les jeux de données initiaux).

Tableau 3 – Caractéristiques des données : nombre des exemples d'apprentissage et de test, nombre d'attributs, nombre de classes.

Nom	# App.	# Test	# Attr.	# Classes
emotion	391	202	72	6
scene	1211	1196	294	6
yeast	1500	917	103	14

Vu l'écart entre les valeurs des différents attributs, nous utilisons des données *centrées réduites*.

5.2 Les métriques d'évaluation

Pour évaluer la performance de notre classifieur multi-label, plusieurs critères existent dans la littérature [9]. Nous rappelons dans ce qui suit deux métriques d'évaluation utilisées : le *coût de Hamming* et la *précision*. Soit l'ensemble test $T = \{(x_i, Y_i), i = 1, ..., N\}$, avec Y_i le vrai ensemble d'étiquettes associé à l'exemple test x_i et soit $\hat{Y}_i$ l'ensemble d'étiquettes prédites par le classifieur multi-label pour x_i.

- **Coût de Hamming** : Ce critère évalue combien d'étiquettes sont mal classées (une étiquette n'appartenant pas à Y_i est prédite ou bien une étiquette appartenant à Y_i n'est pas prédite).

$$C.H. = \frac{1}{N}\sum_{i=1}^{N}\frac{\mid Y_i \Delta \hat{Y}_i \mid}{l}. \qquad (21)$$

Δ étant la différence symétrique entre deux ensembles. Plus petite est la valeur du *Coût de Hamming*, plus grande est la performance.

- **Précision** : Cette métrique mesure le degré de similarité entre Y_i et $\hat{Y}_i$:

$$Prec. = \frac{1}{N}\sum_{i=1}^{N}\frac{\mid Y_i \cap \hat{Y}_i \mid}{\mid Y_i \cup \hat{Y}_i \mid}. \qquad (22)$$

Contrairement au *Coût de Hamming*, plus grande est la valeur de la *Précision*, plus grande est la performance.

5.3 Algorithme d'apprentissage et résultats

Comme on a déjà vu dans les paragraphes 2 et 3, on a besoin de $l(l+1)/2$ classifieurs binaires évidentiels, pour ceci nous transformons les jeux de données multi-label $(paragraphe\ 4.2)$ et nous utilisons comme classifieurs les *k-plus proches voisins (k-PPV) evidentiels de Denoeux* [1] (Programmes matlab téléchargeables à l'adresse www.hds.utc.fr/ tdenoeux). Nous aurons en sortie, pour chaque classifieur M_{ij} utilisés sur les jeux d'apprentissage transformés (voir paragraphe 4.2), les fonctions de masses associées à chaque exemple x : $m_x^{\Theta_{ij}}(\theta_{ij}) = \alpha_{ij}$ et $m_x^{\Theta_{ij}}(\overline{\theta_{ij}}) = \beta_{ij}$. Ensuite, ayant pour chaque instance x l'ensemble des valeurs α_{ij} et β_{ij}, nous cherchons la permutation optimale $\tau^*(x)$. En utilisant l'étiquette λ_0, on peut passer de l'ordonnancement $\tau^*(x)$ à l'ensemble d'étiquettes pertinentes P_x *(paragraphe 4.1)*.
Nous avons fait varier le nombre de plus proches voisins k de 1 à 13, et nous avons reporté les meilleurs résultats dans la dernière ligne des tableaux 4, 5 et 6.

5.4 Discussion

Pour tester la performance de notre méthode, nous comparons les résultats expérimentaux obtenus avec d'autres approches. Plusieurs méthodes ont été proposées dans la littérature pour traiter les problèmes de classification multi-label. Ces méthodes peuvent être divisées en deux catégories selon la façon dont on traite l'ensemble de données d'apprentissage. La première catégorie transforme le problème d'apprentissage multi-label en un ou plusieurs problèmes d'apprentissage à une seule classe, tandis que la seconde catégorie se base sur l'adaptation directe des algorithmes de classification mono-label pour l'apprentissage multi-label. Notre approche faisant partie du premier

groupe, nous comparons nos résultats avec des méthodes appartenant à la première catégorie :

- *Binary Relevance* (BR) : [10] Elle consiste à construire l classifieurs binaires, chaque classifieur est utilisé pour séparer une classe des autres. Pour une nouvelle instance x, la sortie de BR est l'ensemble des étiquettes λ_i prédites par chaque classifieur binaire.
- *Label Powerset* (LP) : [10] Cette approche considère chaque ensemble P_x dans l'ensemble d'apprentissage comme une étiquette pour un nouveau problème d'apprentissage mono-label. Pour une nouvelle instance x, le classifieur monolabel fait sortir une étiquette, qui n'est autre que l'ensemble P_x prédit pour l'instance x.
- *Random k-labelsets* (RA*k*EL) : [10] Elle consiste à décomposer de manière aléatoire l'ensemble $\pounds$ en des sous-ensembles et puis à construire un classifieur LP pour chacun de ces sous-ensembles. Pour chaque étiquette λ_i une décision moyenne est calculée. La décision finale est positive pour une étiquette donnée si la décision moyenne est plus grande qu'une valeur seuil *t*.
- *RAkEL évidentielle* (E-RA*k*EL) : [4] Elle utilise la méthode RA*k*EL conjointement avec la théorie des fonctions de croyance. L'utilisation de la théorie des fonctions de croyance rend possible l'association d'une fonction de masse à chaque classifieur. Ces fonctions sont ensuite combinées par un opérateur adapté dans le but de donner une décision finale sur l'appartenance d'un individu à un ensemble d'étiquettes.
- *Classifier Chains* (CC) : [6] Comme dans le cas de la méthode *Binary Relevance*, on construit l classifieurs binaires, mais afin d'intégrer d'éventuelles dépendances entre étiquettes, le $j^{\text{ième}}$ classifieur utilise dans ses attributs d'entrée les prédictions des $j-1$ classifieurs précédents.

Pour l'approche RA*k*EL évidentielle, nous utilisons les résultats figurant dans [4]. Pour les autres approches, nous utilisons les résultats figurant dans [7] qui sont issus des algorithmes se trouvant dans la librairie *MULAN (http : //mlkd.csd.auth.gr/multilabel.html#Software)*. Nous considérons uniquement les résultats issus de l'utilisation des *k-plus proches voisins* comme classifieurs binaires.

En analysant les valeurs se trouvant dans les tableaux 4, 5 et 6, on voit que les résultats obtenus en transformant un problème de classification multi-label en un problème d'ordonnancement, résolu en utilisant le cadre théorique des fonctions de croyance, sont comparables par rapport à ceux des autres méthodes de classification multi-label. Il serait cependant nécessaire de pousser cette comparaison pour inclure le coût de rang (ranking loss), potentiellement plus favorable à notre méthode basée sur l'identification d'un ordonnancement entre étiquettes.

Tableau 4 – Résultats sur le jeu de données emotions.

Approche	C.H.	Rang	Prec	Rang
BR	0.188	1	0.551	4
LP	0.215	5	0.56	3
RAkEL	0.198	4	0.577	2
E-RAkEL	0.235	6	0.519	6
CC	0.197	3	0.584	1
MC	0.1914	2	0.5404	5

Tableau 5 – Résultats sur le jeu de données scene.

Approche	C.H.	Rang	Prec	Rang
BR	0.094	1	0.643	5
LP	0.097	3	0.713	1
RAkEL	0.095	2	0.694	3
E-RAkEL	0.129	6	0.611	6
CC	0.100	4	0.701	2
MC	0.101	5	0.6469	4

Tableau 6 – Résultats sur le jeu de données yeast.

Approche	C.H.	Rang	Prec	Rang
BR	0.193	1	0.522	2
LP	0.213	4	0.523	1
RAkEL	0.208	3	0.493	5
CC	0.213	4	0.521	3
MC	0.1974	2	0.5157	4

6 Conclusion

Dans cet article, nous avons présenté une nouvelle méthode pour la classification multi-label. L'approche proposée utilise la technique de transformation d'un problème de classification multi-label en un problème d'ordonnancement. Ce dernier est résolu par la méthode de préférence par paires. Les sorties des classifieurs binaires sont représentées par des fonctions de croyance. Nous utilisons la combinaison conjonctive et les commonalités pour aboutir à l'ordonnancement optimal *(méthode de maximum de plausibilité)*. La méthode développée a été testée sur des jeux de données classiques. Les résultats obtenus montrent que la méthode est compétitive par rapport à d'autres méthodes de classification multi-label.

Dans le futur, nous envisageons d'étendre l'approche d'ordonnancement d'étiquettes par fonctions de croyance à la prédiction d'ordres partiels et de l'appliquer en particulier à la classification multi-label. En effet, les fonctions de croyance sont bien adaptées au problème de prédictions partielles, et pourrait être avantageuses dans le cas où de telles prédictions font sens. Dans les autres cas (prédictions complètes), notre approche apparait comme coûteuse par rapport aux gains de précisions obtenus.

Références

[1] T. Denoeux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*, 25(5), 804-813, May 1995.

[2] J. Fürnkranz, E. Hüllermeier. Preference Learning and Ranking by Pairwise Comparison. *Preference Learning*, Springer, 2011.

[3] J. Fürnkranz, E. Hüllermeier, E.L. Mencia, K. Brinker Multilabel Classification via Calibrated Label Ranking. In *Journal Machine Learning*, 73(2) : 133-153, 2008.

[4] S. Kanj, F. Abdallah, T. Denoeux. La méthode RA*k*EL évidentielle pour la classification multi-label.*Rencontres Francophones sur la Logique Floue et ses Applications (LFA)*, 2012.

[5] M.-H. Masson, L. Qiang, T. Denoeux Ordonnancement d'alternatives dans le cadre de la théorie des fonctions de croyance. *Rencontres Francophones sur la Logique Floue et ses Applications (LFA)*, 2010.

[6] J. Read, B. Pfahringer, G. Holmes, and E. Frank. Classifier Chains for multi-label classification, *Machine Learning*, 85(3), 333-359, (2011).

[7] A. M. Santos, A. M. P. Canuto and A. F. Neto. A Comparative Analysis of Classification Methods to Multi-Label Tasks in Different Application Domains, *International Journal of Computer Information Systems and Industrial Management Applications*, ISSN 2150-7988(3), 218-227, (2011).

[8] G. Shafer A mathematical theory of evidence. *Princeton University Press*, Princeton, N.J., 1976.

[9] G. Tsoumakas and I. Katakis. Multi-label classification : An overview. *International Journal of Data Warehousing and Mining*, 3(3), 1-13, July-September 2007.

[10] G. Tsoumakas, I. Katakis, and I. Vlahavas. Random k-Labelsets for Multi-Label Classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7), 1079– 1089, 2011.