

État de l'art (B) : Sélection de variables *Semi-Supervisée – Mono-Label*

LEFEBVRE Julien

P2105454

MERCIER Loris

P1906860

Encadrant : Mr. Khalid BENABDESLEM

Unité d'Enseignement : MIF11 – Ouverture à la recherche

Année : 2022 - 2023

Table des matières

Introduction	1
1 Les méthodes de filtre :	1
1.1 Théorie des graphes spectraux	1
1.2 Score Laplacien	1
1.3 Critère de Fisher	2
1.4 Score de contrainte	2
2 Les méthodes Wrapper (symbioses) :	3
2.1 Un seul classifieur	3
2.2 Ensemble de classifieur	4
3 Les méthodes Embedded (intégrées)	5
3.1 FS-Manifold	5
3.2 Elimination récursive de variables (Backward elimination)	5
Conclusion.....	6
Références	7

Table des illustrations

Figure 1: Algorithme FW-SemiFS [10]	3
Figure 2: Algorithme SSFI [2]	4
Figure 3: Algorithme SEFR [4].....	4
Figure 4: Algorithme S ³ VM-RFE [1].....	5

Introduction

La sélection de variables est une tâche cruciale dans de nombreux domaines de l'analyse de données tel que la finance, la santé, ou la reconnaissance d'image. Elle permet d'identifier les variables les plus pertinentes pour la prédiction d'une variable cible, tout en réduisant la complexité du modèle. Traditionnellement, la sélection de variables est effectuée en utilisant des données étiquetées, c'est-à-dire des données pour lesquelles la variable cible est connue. Vous retrouvez d'ailleurs une présentation complète de cette approche dans notre état de l'art¹ dédié à l'approche supervisée.

Cependant, dans de nombreux cas, les données disponibles ne sont pas entièrement labélisées rendant la sélection de variables supervisée difficilement utilisable. C'est alors qu'intervient la sélection de variables semi-supervisée permettant d'intégrer dans un même processus des données étiquetées et non étiquetées afin de mieux prédire la variable cible.

Tout comme la partie supervisée, il est possible de classer ces méthodes selon trois grandes familles d'approche que sont les filtres, les wrapper (*symbiose*) et les méthodes embedded (*intégrées*). C'est d'ailleurs à partir de cette taxonomie que nous présenterons dans le cadre de cet état de l'art les différentes méthodes de sélection de variables semi-supervisée en tentant de mettre en évidence les avantages et limites de chacune de ces approches.

1 Les méthodes de filtre :

Tout comme l'apprentissage supervisé, les méthodes de filtrage semi-supervisées ont pour objectif de supprimer les variables « non-signifiantes » afin de réduire l'espace de données. Ces méthodes sont indépendantes du classifieur utilisé empêchant ainsi un surajustement des données. Dans le cadre de l'approche semi-supervisée, différentes sous-familles d'approche par filtrage co-existent.

1.1 Théorie des graphes spectraux

Présentée en 2007 par Z.Zhao et H.Liu [13], cette méthode est la première approche semi-supervisée se basant sur l'analyse spectrale permettant de résoudre le problème du « plus petit échantillon étiqueté ». En effet, il n'est pas rare dans l'approche semi-supervisée que la proportion de donnée étiquetée soit bien plus faible que la partie étiquetée. Ainsi, les méthodes de sélection classique utilisées en mode supervisé ne peuvent fonctionner correctement. Pour pallier ce problème, la méthode proposée ici exploite à la fois les données labélisées et non-labelisées à travers une méthode de régularisation des données.

1.2 Score Laplacien

Utilisé initialement dans l'approche non-supervisée, cette méthode est introduite en 2005 par X.He, D.Cai et P.Niyogi [7]. Cette technique consiste à sélectionner les variables pertinentes préservant au mieux la structure locale et produisant de grandes valeurs de variances. Très performante pour les

¹ Voir *État de l'art (A) : Sélection de variables Supervisée – Mono-Label*

données non-étiquetées, plusieurs chercheurs ont depuis 15 ans essayé d'adapter cette méthode à l'approche semi-supervisée.

Nous pouvons par exemple citer le S-Laplacian Score introduit en 2011 par Cheng and al. [6] renforçant encore plus l'aspect de préservation de localité dans le choix des variables ou l'algorithme LSDF [14] fondé sur l'analyse discriminante sensible à la localité.

A noter que l'ensemble de ces algorithmes font partie de la plus grande famille des sélections de variables basées sur des graphes.

1.3 Critère de Fisher

Très largement utilisé dans le cadre de l'approche supervisée, plusieurs chercheurs ont depuis adapté cette technique à l'approche semi-supervisée. Concrètement, l'ensemble de ces techniques utilise la partie étiquetées des données pour les séparer en différentes classes et utilise la partie non-labellisé pour conserver la structure locale. Ces méthodes combinent donc les avantages du critère de Fisher pour la partie supervisées des données avec différents critères plus appropriées aux données non-labellisées.

C'est par exemple le cas de Yang et al. [12] qui combine un critère de variance utilisé dans le cadre du test de fisher et un critère de préservation de la localité proche du score Laplacien présentée auparavant.

1.4 Score de contrainte

Cette technique s'appuie sur la notion de contraintes entre les données non évoquées jusqu'à présent. Aussi bien dans l'approche supervisée que semi-supervisée, les contraintes représentent une connaissance initiale supplémentaires afin de mieux décrire la variable cible. Elle ajoute des conditions, critères entre les données souvent plus facile à obtenir dans les jeux de données que l'étiquetage complet. Plusieurs types de contraintes existent. Néanmoins, la majorité des algorithmes semi-supervisé repose sur les contraintes « par paires ».

Les contraintes par paires précisent pour une paire de donnée spécifique si elles appartiennent, ou non, à la même classe. On nomme alors ces contraintes « must-link » (M) pour des classes identiques et « not-link » (C) pour des classes différentes. C'est partir de ces contraintes qu'est défini un « score de contrainte » propre à chaque algorithme de sélection de variable. Cela se traduit principalement par la définition d'une formule mathématique cherchant à maximiser le nombre de contrainte satisfaite.

Introduite en 2008 par Zhang et al en 2008 [15] dans le cadre de l'approche supervisée, les méthodes reposant sur le score de contrainte peuvent facilement s'étendre au cadre semi-supervisé. C'est par exemple le cas de l'algorithme proposé par Zhao et al [14] qui définit un score utilisant à la fois les contraintes par paires et les plus proches voisins non étiquetés des échantillons. Il a d'ailleurs été montré par des résultats expérimentaux que cette approche obtient de meilleures performances que les approches utilisant le critère de Fisher.

Une amélioration a cet algorithme a ensuite été proposé en 2011 [9] par Kalakech et al. Les chercheurs ont défini leur score de contrainte comme le simple produit entre le score Laplacien (prenant en compte les données non-labellisées) et le score de contrainte défini par Zhang [15] prenant en compte les données étiquetées.

2 Les méthodes Wrapper (symbioses) :

Deuxième approche de la sélection de variable, les méthodes wrapper consistent à utiliser directement le modèle d'apprentissage pour évaluer pas à pas un sous-ensemble de variables afin de sélectionner celui améliorant au mieux les performances du classifieur. Ces méthodes ont l'avantage par rapport aux autres approches de mieux prendre en compte les variables cibles complexes et non-linéaires. Néanmoins, ces méthodes sont plus lourdes et plus lentes à utiliser rendant leur utilisation délicate pour de grands jeux de données.

Dans le cadre de l'approche semi-supervisée, deux catégories de méthodes Wrapper peuvent se retrouver dans la littérature : les méthodes basées sur un seul classifieur et les méthodes basées sur un ensemble de classifieurs.

2.1 Un seul classifieur

Ren et Al [10] ont introduit en 2008 le Forward Semi-Supervised Feature Selection (*FW-SemiFS*). L'algorithme utilise l'algorithme de *sélection de variables séquentiel avant (SFFS)* avec l'approche wrapper. En pratique, cette méthode consiste à sélectionner les n premières variables afin de construire un classificateur qui sera utilisé pour prédire les étiquettes des données non étiquetées. Ensuite, les données non étiquetées sélectionnées aléatoirement avec leurs étiquettes prédites sont combinées avec les données déjà étiquetées pour former un nouvel ensemble d'apprentissage.

```
Input:  $L, U, sizeFS, samplingRate, samplingTimes, maxIterations,$   
        $startfn, fnstep$   
Output:  $resultfs$   
1 Perform feature selection on  $L$  using SFFS, select  $startfn$  features to  
  form the current feature subset  $currentfs$ ;  
2  $ReducedL \leftarrow L * currentfs$ ;  
3  $ReducedU \leftarrow U * currentfs$ ;  
4 for iteration  $\leftarrow 1$  to  $maxIterations$  do  
5    $Predicted \leftarrow \text{classifier}(ReducedL, ReducedU)$ ;  
6   for rand  $\leftarrow 1$  to  $samplingTimes$  do  
7     Randomly select  $samplingRate\%$  of instances from  $Predicted$ ,  
       and add it into  $L$  to form a new dataset  $NewDataset$ ;  
8     Perform feature selection on  $NewDataset$  using SFFS, select  
        $fnstep$  features to form feature subset  $fs[rand]$ ;  
9   end  
10  Count the frequency of every feature in  $fs$ , add the most frequent  
    and not in  $currentfs$  feature into  $currentfs$ ;  
11   $ReducedL \leftarrow L * currentfs$ ;  
12   $ReducedU \leftarrow U * currentfs$ ;  
13  if  $SIZE(currentfs) == sizeFS$  then break;  
14 end  
15  $resultfs \leftarrow currentfs$ ;
```

Figure 1: Algorithme FW-SemiFS [10]

Dans un second temps, le nouvel ensemble de données d'apprentissage est utilisé pour sélectionner les variables en fonction de la SFFS et de l'apprenant. Le processus de sélection aléatoire et de sélection des variables est répété un certain nombre de fois et un certain nombre de groupes de variables sont sélectionnés. La fréquence de chaque caractéristique dans les groupes de variables est calculée, et celle qui a la plus grande fréquence est ajoutée pour former un nouveau sous-ensemble de variables.

Ce processus se répète jusqu'à ce que la taille du sous-ensemble de variables atteigne un nombre prédéfini.

Néanmoins comme l'évoque Han et al [8] dans leur article de 2011, la méthode FW-SemiFS ne tient pas compte de la confiance des données prédites non étiquetées, mais évalue plutôt la pertinence des variables en fonction de leur fréquence. Ces fréquences sont obtenues par une sélection itérative supervisée de variables séquentielles avant (SFFS). Cependant, le temps de calcul important associé à la SFFS itérative est préjudiciable à FW-SemiFS. De plus, cette méthode d'évaluation de la pertinence élimine le principal avantage de la sélection de variables de type wrapper : la possibilité d'évaluer le pouvoir discriminant d'une combinaison de variables [8].

2.2 Ensemble de classifieur

D'autre part, il existe des méthodes entraînant plusieurs classifieurs avant de combiner leurs résultats en sortie.

Algorithm 1 SSFI($L, U, F, K, N, n, \text{maxiter}, \text{BaseLearn}$)

Require:
 set of labeled training examples (L), set of unlabeled training examples (U), input space ($F = \{f_1, \dots, f_p\}$), number of classes (K), committee size (N), sample size (n), maximum number of iterations (maxiter) and base learning algorithm (BaseLearn)

- 1: Get the class prior probabilities, $\{Pr_k\}_{k=1}^K$
- 2: Set the class growth rate, $n_k = n \times Pr_k$ where $k = 1, \dots, K$

Initial committee construction H

- 3: $H = \emptyset$
- 4: **for** $i = 1 : N$ **do**
- 5: RSM^i = randomly draw m features from F
- 6: L_{bag}^i = bootstrap sample from L projected onto RSM^i
- 7: U_{bag}^i = bootstrap sample from U projected onto RSM^i
- 8: $L_{oob}^i = L \setminus L_{bag}^i$, $U_{oob}^i = U \setminus U_{bag}^i$
- 9: $h^i = \text{BaseLearn}(L_{bag}^i)$
- 10: $H = H \cup h^i$
- 11: **end for**

Committee refinement using SSL ensemble method

- 12: $t = 1$
- 13: **repeat**
- 14: **for** each $h^i \in H$ **do**
- 15: $\pi^i = \text{SelectConfidentExamples}(i, H, U_{bag}^i, \{n_k\}_{k=1}^K)$
- 16: $L_{bag}^i = L_{bag}^i \cup \pi^i$, $U_{bag}^i = U_{bag}^i \setminus \pi^i$
- 17: $h^i = \text{BaseLearn}(L_{bag}^i)$
- 18: **end for**
- 19: $t = t + 1$
- 20: **until** ($t > \text{maxiter}$ OR no committee member changes)

Feature relevance estimate

- 21: $\text{imp} = 0$
- 22: **for** each $h^i \in H$ **do**
- 23: $[O_{data}^i, O_{label}^i, O_{conf}^i] = \text{BuildOOBMatrix}(i, H, L_{oob}^i, U_{oob}^i, K)$
- 24: **for** each $f \in RSM^i$ **do**
- 25: randomly permute the values of f over the O_{data}^i examples to form O_{perm}^i
- 26: **for** each $x \in O_{perm}^i$ **do**
- 27: **if** ($h^i(x) \neq O_{label}^i(x)$) **then**
- 28: $\text{imp}(f) = \text{imp}(f) + O_{conf}^i(x)$
- 29: **end if**
- 30: **end for**
- 31: **end for**
- 32: **end for**
- 33: rank the features f according to $\text{imp}(f)$
- 34: **return** F and imp

Figure 2: Algorithme SSFI [2]

En effet dans l'article de Barkia et al [2], les chercheurs proposent une méthode d'évaluation de l'importance des variables semi-supervisées, appelée SSFI. Cette méthode combine le co-training et le random forest [5] avec une nouvelle mesure de l'importance des variables basées sur la permutation tout en utilisant les données étiquetées et non étiquetées. SSFI combine à la fois des stratégies de ré-échantillonnage des données (bagging) et de sous-espace aléatoire pour générer un apprenant d'ensemble à l'aide d'un algorithme de type co-training. La combinaison de ces deux stratégies pour produire un ensemble de classificateurs conduit à l'exploration de points de vue distincts sur les relations inter-modèles. Une fois que chaque membre de l'ensemble est obtenu, une extension de la mesure d'importance de permutation RF [5],

utilisant les données étiquetées et non étiquetées ensemble, est proposée pour mesurer la pertinence de la variable. Un classement de toutes les variables est finalement obtenu par rapport à leurs pertinences dans tous les classifieurs semi-supervisés obtenus.

Data: $\mathcal{D}, L, U, F = \{f_1, \dots, f_p\}, \text{nbags}, \text{BaseLearn}, \phi$
Result: F, imp
 $\text{imp} = 0$;
for $i = 1 : \text{nbags}$ **do**
 L_{bag}^i = bootstrap sample from L ;
 U_{bag}^i = bootstrap sample from U ;
 $L_{oob}^i = L \setminus L_{bag}^i$; $U_{oob}^i = U \setminus U_{bag}^i$;
 randomly draw $\sqrt{|F|}$ features from F to form F' ;
 /* Labeling by self-training */
while $U_{bag}^i \neq \emptyset$ **do**
 $U' = \text{selectMostConfident}(\mathcal{D}(L_{bag}^i \cup U_{bag}^i, F'),$
 $L_{bag}^i, U_{bag}^i, \phi)$;
if $U' = \emptyset$ **then**
 break;
end
 $L_{bag}^i = L_{bag}^i \cup U'$;
 $U_{bag}^i = U_{bag}^i \setminus U'$;
end
 /* Feature importance measures */
 apply ϕ to $\mathcal{D}(L_{oob}^i \cup U_{oob}^i, F')$;
 select the well classified samples in L_{oob}^i to form L' ;
 $U' = \text{selectMostConfident}(\mathcal{D}(L_{oob}^i \cup U_{oob}^i, F'), L_{oob}^i, U_{oob}^i, \phi)$;
 define y as the predicted labels of $L' \cup U'$;
for each $f \in F$ **do**
 randomly permute f in \mathcal{D} to form $\text{perm}\mathcal{D}$;
 apply ϕ to $\text{perm}\mathcal{D}(L' \cup U', F)$;
 define y_p as the predicted labels of $L' \cup U'$;
 Increase $\text{imp}(f)$ by the number of mismatches
 between y and y_p ;
end
end
 rank the features f according to $\text{imp}(f)$ and return both F and imp ;

De plus, l'article [4] présente une méthode similaire à SSFI appelée méthode de classement des variables guidée par l'apprentissage d'ensemble semi-supervisé (SEFR), l'algorithme classe les variables à travers un ensemble de modèles, dans lequel la pertinence d'une variable est évaluée par sa précision prédictive en utilisant des données étiquetées et non étiquetées.

Figure 3: Algorithme SEFR [4]

3 Les méthodes Embedded (intégrées)

Sur le même principe que les méthodes filtres ou de symbioses, les méthodes intégrées semi-supervisées s'appuient sur une combinaison d'algorithmes reconnues en mode supervisé et/ou non supervisé. Les avantages sont donc très semblables, à savoir l'intégration du processus de sélection de variables directement dans la phase d'apprentissage du classifieur optimisant au mieux cette sélection. Ces méthodes sont plus rapides que les approches wrapper mais moins que l'approche filtrage. L'inconvénient majeurs de cette technique est en revanche la spécificité de chaque sous-ensemble de variables sélectionné. En effet, les variables choisies étant spécifique à un classifieur, il faut pour chaque modification de classifieur réexécuter toute la phase de sélection.

Parmi les nombreux algorithmes intégrés existant, nous évoquerons principalement les algorithmes s'appuyant sur les classifieurs linéaires tel que SVM. (Support-Vector Machine). Ces algorithmes utilisent pour cela différentes méthodes de sélection comme la régularisation multiple (*Manifold Regularization*), les normes de régularisation l1 et/ou l2 ou l'élimination récursive de variable.

3.1 FS-Manifold

Dans leur article de 2009, Xu et al [11] ont d'ailleurs proposé une nouvelle méthode de sélection semi-supervisée discriminative s'appuyant sur l'idée de la régularisation multiple introduit par Belkin et al [3]. Ce nouvel algorithme sélectionne les variables en maximisant la marge de classification entre les différentes classes et en exploitant parallèlement la structure locales des données labélisées et non-labélisées. D'après les chercheurs, cette méthode permet de trouver des variables plus discriminées. Leurs résultats expérimentaux ont aussi montré des performances supérieures aux algorithmes classiques de filtrage tel que le critère de Fisher. Cela est principalement lié à la grande optimisation de leur méthode qui peut se formaliser à travers un problème d'optimisation concave-convex à résoudre.

3.2 Elimination récursive de variables (Backward elimination)

Autres approches de sélection semi-supervisées, les algorithmes *S3VM-RFE* et *S3VM-FS* [1] (Ang et al) s'appuient sur la méthode supervisée SVM-RFE. Cette méthode est très ressemblante au principe de backward elimination détaillée précédemment.

Algorithm 1. S^3VM -RFE (X_0, y):
Input
Training examples, $X_0 = [x_1, x_2, \dots, x_t, x_{(t+1)}, \dots, x_u]^T$
Class labels, $y = [y_1, y_2, \dots, y_t, y_{(t+1)}, \dots, y_u]^T$, where $[y_1, \dots, y_t] \in \{-1, +1\}$ and $[y_{(t+1)}, \dots, y_u] = 0$
Initialize
Subset of surviving features, $s = [1, 2, \dots, n]$
Feature ranked list, $r = []$
Repeat until $s = []$
Start
1. Restrict training examples to good feature indices $labelx = X_0(:, s);$ $unlabeledx = X_0(:, s);$
2. Train the classifier $\alpha_{label} = SVM\text{-train}(X, y);$ %Supervised SVM $\alpha_{unlabel} = SVM\text{-train}(X);$ %Unsupervised one-class SVM
3. Compute the weight vector of dimension length(s), for all i $w_i^l = \sum_k \alpha_{(label_k)} y_k x_k$ $w_i^u = \sum_k \alpha_{(unlabel_k)} y_k x_k$
4. Determine the sum of w_i^l and w_i^u , $W_i = w_i^l + w_i^u$;
5. Compute the ranking criteria, $c_i = (W_i)^2$;
6. Find the feature with smallest ranking criterion, $f = \text{argmin}(c)$;
7. Update feature ranked list, $r = [s(f), r]$;
8. Eliminate the feature with smallest ranking criterion $s = s(1 : f - 1, f + 1 : \text{length}(s))$;
9. Repeat until all the features are ranked.
Output
Ranked feature set r .

Figure 4: Algorithme S^3VM -RFE [1]

Elle consiste à classer les variables dans l'ordre décroissant en fonction de leur poids puis à supprimer celles les moins bien classer. Les variables restantes s'entraînent ensuite récursivement avec le classifieur SVM jusqu'à ce que l'ensemble des variables soient classées.

S3VM-RFE est une première évolution semi-supervisée de l'algorithme supervisée SVM-RFE. Il prend en compte à la fois les données étiquetées et celles non-étiquetées dans son processus d'apprentissage. Les données labellisées sont formées avec un SVM supervisé alors qu'un algorithme SVM non-supervisé est appliqué aux données non-labellisées. Le vecteur de pondération des variables, faisant office de fonction de score, est ensuite conçu en

combinant les deux SVM utilisés. La variable avec le plus petit score est éliminée puis l'algorithme continue son apprentissage jusqu'à ce que l'ensemble des variables soient classées.

Une autre variante de cet algorithme est la méthode *S3VM-FS* utilisant un nombre de variable à conserver prédéfini.

Conclusion

En conclusion, nous avons vu à travers cet état de l'art que la majeure partie des méthodes de sélection de variables semi-supervisées reposent sur des adaptations algorithmes préexistants dans d'autres modes d'apprentissage. En effet, l'approche semi-supervisée étant un compromis entre les approches supervisées et non-supervisées plus ancienne, il n'est pas étonnant de repartir de ces méthodes ayant fait leurs preuves dans leurs domaines respectifs.

Ainsi, nous retrouvons une classification commune des méthodes de sélection de variable, à savoir les méthodes de filtre s'appuyant sur le classement puis la sélection de variables en fonction de leur résultat à un test statistique, les méthodes wrapper utilisant directement le modèle d'apprentissage afin d'évaluer la performance de plusieurs combinaisons de variables et les méthodes embedded, compromis entre les deux précédentes méthodes, consistant à sélectionner les variables tout en entraînant le modèle.

Tout comme pour l'approche supervisée, il est difficile de définir laquelle des méthodes présentées est la plus performante. En réalité, il n'existe tout simplement pas de « meilleur » algorithme dans l'absolu. A chaque problème se voit attribuer une solution qui variera en fonction des différents paramètres. Ainsi, il sera par exemple préférable d'utiliser des approches par filtrage lorsqu'un jeu de données présente une grande dimensionnalité. Cependant, dans un cas plus réduit avec des variables très complexes et non linéaires, une approche wrapper pourrait être plus performante. Chaque méthode possède donc ses avantages et inconvénients et il est important de comprendre ces différences afin de choisir la solution la plus appropriée à chaque situation.

Références

- [1] Ang, J. C., Haron, H., & Hamed, H. N. A. (2015, May). Semi-supervised SVM-based feature selection for cancer classification using microarray gene expression data. In *Current Approaches in Applied Artificial Intelligence: 28th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2015, Seoul, South Korea, June 10-12, 2015, Proceedings* (pp. 468-477). Cham: Springer International Publishing.
- [2] Barkia, H., Elghazel, H., & Aussem, A. (2011, December). Semi-supervised feature importance evaluation with ensemble learning. In *2011 IEEE 11th International Conference on Data Mining* (pp. 31-40). IEEE.
- [3] Belkin, M., Niyogi, P., & Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(11).
- [4] Bellal, F., Elghazel, H., & Aussem, A. (2012). A semi-supervised feature ranking method with ensemble learning. *Pattern Recognition Letters*, 33(10), 1426-1433.
- [5] Breiman, L. (2001). Random forests. *Machine learning*, 45, 5-32.
- [6] Cheng, H., Deng, W., Fu, C., Wang, Y., & Qin, Z. (2011). Graph-based semi-supervised feature selection with application to automatic spam image identification. In *Computer Science for Environmental Engineering and EcoInformatics: International Workshop, CSEEE 2011, Kunming, China, July 29-31, 2011, Proceedings, Part II* (pp. 259-264). Springer Berlin Heidelberg.
- [7] He, X., Cai, D., & Niyogi, P. (2005). Laplacian score for feature selection. *Advances in neural information processing systems*, 18.
- [8] Han, Y., Park, K., & Lee, Y. K. (2011, August). Confident wrapper-type semi-supervised feature selection using an ensemble classifier. In *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)* (pp. 4581-4586). IEEE.
- [9] Kalakech, M., Biela, P., Macaire, L., & Hamad, D. (2011). Constraint scores for semi-supervised feature selection: A comparative study. *Pattern Recognition Letters*, 32(5), 656-665.
- [10] Ren, J., Qiu, Z., Fan, W., Cheng, H., & Yu, P. S. (2008). Forward semi-supervised feature selection. In *Advances in Knowledge Discovery and Data Mining: 12th Pacific-Asia Conference, PAKDD 2008 Osaka, Japan, May 20-23, 2008 Proceedings 12* (pp. 970-976). Springer Berlin Heidelberg.
- [11] Xu, Z., King, I., Lyu, M. R. T., & Jin, R. (2010). Discriminative semi-supervised feature selection via manifold regularization. *IEEE Transactions on Neural networks*, 21(7), 1033-1047.
- [12] Yang, M., Chen, Y. J., & Ji, G. L. (2010, July). Semi_Fisher Score: A semi-supervised method for feature selection. In *2010 International Conference on Machine Learning and Cybernetics* (Vol. 1, pp. 527-532). IEEE.
- [13] Zhao, Z., & Liu, H. (2007, April). Semi-supervised feature selection via spectral analysis. In *Proceedings of the 2007 SIAM international conference on data mining* (pp. 641-646). Society for Industrial and Applied Mathematics.
- [14] Zhao, J., Lu, K., & He, X. (2008). Locality sensitive semi-supervised feature selection. *Neurocomputing*, 71(10-12), 1842-1849.
- [15] Zhang, D., Chen, S., & Zhou, Z. H. (2008). Constraint score: A new filter method for feature selection with pairwise constraints. *Pattern Recognition*, 41(5), 1440-1451.