Expériences et résultats

Sommaire

Résumé

Ce chapitre rapporte les expériences menées pour évaluer les processus d'annotation stochastique à base de DBN utilisés sur le corpus MEDIA. *Les trois systèmes proposés dans le chapitre 8 sont appliqués à un ensemble de test, comprenant 3005 tours de parole utilisateur. Les résultats obtenus par chaque modèle sont détaillés selon la nature manuelle ou automatique des données de test considérées.*

9.1 Introduction

Pour évaluer les performances des système de composition des frames utilisant les DBN, un ensemble de données de test est préparé comme indiqué en 6.4. Les 3005 tours de parole utilisateur annotés en frames et FE par un expert forment l'ensemble de référence REF. Le système d'annotation en deux étapes à base de règles (décrit en CHAP.6) est utilisé pour produire une annotation en frames et FE sur le corpus MEDIA (transcription et annotation conceptuelle manuelles), les données de test étant exclues.

La qualité de cette annotation a été évaluée sur les données de test : l'obtention d'une F-mesure toujours supérieure à 0,9 pour l'identification des frames, FE et liens confirme la fiabilité du système et la consistance des données d'apprentissage.

Les expériences visant à évaluer les systèmes de compréhension stochastique à base de DBN proposés dans ce travail sont décrites dans la section 9.2. Les résultats obtenus sont donnés en 9.3.

9.2 Expériences

Les expériences sont menées sur l'ensemble de test dans trois conditions différentes, fonctions de la nature des données initiales :

- MAN : les tours de parole du locuteur sont manuellement transcrits et annotés en concepts ;
- SLU : les concepts de base sont décodés à partir des transcriptions manuelles des tours de parole locuteur, en utilisant le module SLU à base de DBN décrit dans (Lefèvre, 2006) ;
- ASR+SLU : les concepts sont décodés par le modèle de compréhension en utilisant la meilleure hypothèse (1-best) de séquence de mots générée par un système ASR, conforme à (Barrault et al., 2008).

Les données SLU et ASR+SLU comportent des erreurs de transcription et d'annotation conceptuelle liées à l'imperfection des systèmes qui les produisent. Les taux d'erreurs observés sur les 3005 tours de parole de test sont rappelés dans le tableau 9.1.

Type de données	SLU	ASR + SLU
Taux d'erreurs mots (%)	0,0	27
Taux d'erreurs concepts (%)	10,6	24,3

TABLE 9.1 – *Taux d'erreurs en mot et en concept observés sur les données SLU et ASR+SLU de l'ensemble de test* MEDIA.

9.3 Résultats

Toutes les expériences présentées ici ont été réalisées en utilisant GMTK (Bilmes et Zweig, 2002), outil logiciel de calcul et de manipulation des modèles graphiques et SRILM (Stolcke, 2002), outil logiciel pour les modèles de langage.

Les implémentations des trois modèles DBN proposés sont fournies dans l'Annexe C au format standard utilisé par GMTK.

Pour indiquer le dimensionnement des modèles DBN, le nombres de mots, concepts et fragments de frames-FE, frames et FE distincts utilisés pour leur entraînement est donné dans le tableau 9.2.

Modèle DBN	Mots	Concepts	Frag. frames-FE	Frag. frames	Frag. FE
compact	2201	78	636	x	x
factorisé	2201	78	x	234	339
2-niveaux	2201	78	x	234	339

TABLE 9.2 – *Cardinalités des variables de mots, concepts et des classes de fragments de frames-FE, frames et FE distincts utilisées dans les 3 types de modèles DBN (compact, factorisé et 2-niveaux).*

Le nombre total de frames, FE et liens présents sur les tours de parole de l'ensemble REF ansi que sur l'ensemble de test MEDIA est donné en 6.4 (tableau 6.2).

Les résultats des systèmes DBN sont donnés en termes de précision, rappel et F-mesure. La précision est le nombre de frames, FE ou liens corrects proposés par le système divisé par le nombre total de frame, FE ou liens proposés par le système. Le rappel est le nombre de frames, FE ou liens corrects proposés par le système divisé par le nombre total de frames, FE ou liens contenus dans l'annotation de référence. La F-mesure est la moyenne harmonique standard de la précision et du rappel.

Les fragments de frames et FE produits par les systèmes DBN sont évalués à trois niveaux :

- **Frames** : les hypothèses de frames sont considérées séparément et comparées aux frames présentes dans la référence ;
- **FE** : les hypothèses de FE sont considérées séparément et comparées aux FE présents dans la référence ;
- **Frames-FE** : les hypothèses de frames et de FE sont considérées conjointement.

Dans tous les cas, l'ordre d'apparition n'est pas pris en compte.

Pour chacun de ces niveaux, la précision p, le rappel r et la F-mesure $F\text{-}m$ sont calculés globalement sur l'ensemble des N tours de parole (ici $N = 3005$). On a donc :

$$p = \frac{\text{nb d'hypothèses correctes dans les N tours}}{\text{nb total d'hypothèses présentes dans les N tours}}$$

$$r = \frac{\text{nb d'hypothèses correctes présentes dans les N tours}}{\text{nb total d'objets sémantiques présents dans les N tours de référence}}$$

Type de données		**MAN**		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.89	*0.86*	**0.85**
	r	0.81	*0.77*	**0.76**
	$F\text{-}m$	**0.85**	***0.82***	**0.80**
	$\bar{p}$	0.94	*0.93*	**0.92**
	$\bar{r}$	0.89	*0.90*	**0.87**
	$\overline{F\text{-}m}$	**0.92**	***0.92***	**0.89**
F et FE (factorisé)	p	0.85	*0.78*	**0.77**
	r	0.83	*0.72*	**0.72**
	$F\text{-}m$	**0.83**	***0.74***	**0.73**
	$\bar{p}$	0.92	*0.89*	**0.88**
	$\bar{r}$	0.89	*0.88*	**0.86**
	$\overline{F\text{-}m}$	**0.91**	***0.88***	**0.87**
F puis FE (2-niveaux)	p	0.84	*0.76*	**0.75**
	r	0.82	*0.69*	**0.71**
	$F\text{-}m$	**0.83**	***0.73***	**0.73**
	$\bar{p}$	0.91	*0.88*	**0.85**
	$\bar{r}$	0.90	*0.86*	**0.84**
	$\overline{F\text{-}m}$	**0.91**	***0.87***	**0.85**

TABLE 9.3 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version MAN pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

$$F\text{-}m = \frac{p+r}{2}$$

Sont également évalués la précision moyenne $\bar{p}$, le rappel moyen $\bar{r}$ et la F-mesure moyenne $\bar{F\text{-}m}$ pour un tour de parole par les calculs suivants :

$$\bar{p} = \frac{\sum_{i=1}^{N} p_i}{N} \text{ où } p_i \text{ est la précision obtenue au tour } i$$

$$\bar{r} = \frac{\sum_{i=1}^{N} r_i}{N} \text{ où } r_i \text{ est le rappel obtenu au tour } i$$

$$\overline{F\text{-}m} = \frac{\sum_{i=1}^{N} F\text{-}m_i}{N} \text{ où } F\text{-}m_i \text{ est la F-mesure obtenue au tour } i$$

L'intervalle de confiance à 10% des valeurs estimées est d'amplitude 0.02.

Les systèmes apparaissent robustes à la dégradation des données d'entrées : une dégradation de plus de 20% sur les variables observées (mots et concepts) entraine une baisse des performances obtenues sur la génération des fragments sémantiques de moins de 10%.

	Type de données	SLU		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.88	*0.85*	**0.84**
	r	0.78	*0.69*	**0.71**
	$F\text{-}m$	**0.83**	***0.77***	**0.78**
	$\bar{p}$	0.93	*0.92*	**0.91**
	$\bar{r}$	0.87	*0.83*	**0.84**
	$\overline{F\text{-}m}$	**0.90**	***0.88***	**0.87**
F et FE (factorisé)	p	0.84	*0.78*	**0.77**
	r	0.81	*0.64*	**0.68**
	$F\text{-}m$	**0.83**	***0.71***	**0.74**
	$\bar{p}$	0.92	*0.89*	**0.86**
	$\bar{r}$	0.88	*0.81*	**0.82**
	$\overline{F\text{-}m}$	**0.89**	***0.85***	**0.84**
F puis FE (2-niveaux)	p	0.84	*0.75*	**0.75**
	r	0.80	*0.62*	**0.67**
	$F\text{-}m$	**0.82**	***0.69***	**0.71**
	$\bar{p}$	0.91	*0.88*	**0.85**
	$\bar{r}$	0.89	*0.80*	**0.82**
	$\overline{F\text{-}m}$	**0.90**	***0.84***	**0.83**

TABLE 9.4 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version SLU pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

On remarque également que sur les données SLU, le taux d'erreur sur les fragments est voisin du taux d'erreurs concepts observé. Le taux d'erreur concepts est majoré de 13,4% sur les données ASR+SLU (taux d'erreur mots de 27%) alors que les résultats sur les fragments ne sont dégradés que de 6%.

Les résultats des tableaux 9.3, 9.4 et 9.5 montrent que les performances du modèle compact sont supérieures à celles des deux autres modèles. Le domaine de connaissance MEDIA est défini de telle façon qu'un FE ne peut prendre qu'un nombre très limité de frames pour valeur. Ainsi, dans ce contexte, l'utilisation par le modèle compacte de liens déterministes entre frame et FE favorise la production de fragments sémantiques consistants et disposant de statistiques fiables. La simplicité du modèle compact est également un atout dans le cadre de l'intégration de ce modèle à un système de dialogue complet.

Les performances du modèle factorisé et du modèle à deux niveaux permettent de considérer que ces deux modèles sont également adaptés à la tâche de décodage de fragments sémantiques. Nous espérons pouvoir les évaluer rapidement sur la base de connaissances LUNA évoquée en 5.5. Son dimensionnement induit potentiellement un niveau d'incertitude plus élevé dans le choix des frames valeurs de FE. La liberté de combinaison des frames et FE dans les fragments offerte par le modèle factorisé et le

Type de données		ASR + SLU		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.83	*0.78*	**0.78**
	r	0.72	*0.65*	**0.67**
	$F\text{-}m$	**0.77**	***0.71***	**0.72**
	$\bar{p}$	0.87	*0.88*	**0.85**
	$\bar{r}$	0.80	*0.80*	**0.77**
	$\overline{F\text{-}m}$	**0.84**	***0.84***	**0.81**
F et FE (factorisé)	p	0.78	*0.73*	**0.72**
	r	0.75	*0.60*	**0.63**
	$F\text{-}m$	**0.76**	***0.67***	**0.69**
	$\bar{p}$	0.85	*0.84*	**0.80**
	$\bar{r}$	0.82	*0.78*	**0.76**
	$\overline{F\text{-}m}$	**0.83**	***0.82***	**0.78**
F puis FE (2-niveaux)	p	0.79	*0.71*	**0.70**
	r	0.74	*0.58*	**0.62**
	$F\text{-}m$	**0.77**	***0.65***	**0.66**
	$\bar{p}$	0.86	*0.84*	**0.80**
	$\bar{r}$	0.82	*0.77*	**0.75**
	$\overline{F\text{-}m}$	**0.84**	***0.81***	**0.77**

TABLE 9.5 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version ASR+SLU pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

modèle à deux niveaux pourra être un atout dans ce contexte.

9.4 Conclusion

Les résultats obtenus par les systèmes évalués confirment que les modèles à base de DBN peuvent être utilisés pour générer des sous-structures sémantiques hiérarchiques consistantes. De plus, ces modèles ayant la capacité de produire des hypothèses avec leurs scores de confiance, ils peuvent être utilisés dans des contextes où les hypothèses sont multiples (réseaux de confusion, n-meilleures hypothèses) ou encore dans des protocoles d'évaluation en classant les hypothèses proposées par d'autres systèmes.

Les modèles factorisé et à deux niveaux sont aptes à produire des fragments sémantiques consistants sur les dialogues de test MEDIA. Leurs performances restent cependant inférieures à celles du modèle compact, certainement avantagé par la structure et le dimensionnement de notre base de connaissances.

Les fragments sémantiques sont générés par les DBN dans le cadre d'un processus séquentiel qui ne prend pas en compte les dépendances "longue-distance" aux obser-

vations. Ces fragments forment les constituants structurés de la représentation sémantique complète du message de l'utilisateur. Celle-ci est obtenue grâce à une étape de recomposition complémentaire présentée dans la dernière partie de ce document.

Composition des fragments sémantiques

Chapitre 10

Composition d'arbres : modèles et stratégies

Sommaire

Résumé

Ce chapitre propose une présentation de la notion d'arbre employée pour représenter les relations sémantiques dans notre contexte de travail. Il rappelle ensuite dans la section 10.3 les fondements théoriques des modèles de classification basés sur les séparateurs à vaste marge utilisés dans l'une des stratégies de composition des fragments sémantiques.

10.1 Introduction

Les fragments sémantiques générés par les DBN sont représentés par des arbres que nous composons selon deux stratégies, dont une à base de séparateurs à vaste marge (SVM). La définition des arbres et les bases théoriques des SVM sont rappelées dans ce chapitre.

Les systèmes génératifs à base de DBN présentés au chapitre précédent ont la capacité de produire des fragments d'arbres sémantiques que l'on doit ensuite recomposer.

La première partie 10.2 de ce chapitre rappelle tout d'abord quelques définitions et propriétés associées à la structure d'arbre qui est utilisée dans nos travaux pour supporter la représentation sémantique des messages utilisateur. Quelques approches classiques de composition des structures d'arbres sont ensuite présentées.

Une des stratégie de composition évaluée dans ce travail s'appuie sur un algorithme de décison à base de séparateurs à vaste marge (SVM). La seconde section de ce chapitre 10.3 s'attache à exposer les points théoriques qui sous-tendent cet algorithme de décision et définit les SVM dans leur contexte mathématique.

10.2 Notion d'arbre

La notion d'*arbre* est définie dans le cadre de la théorie des graphes. Les graphes permettent de modéliser toute situation mettant en jeu un nombre fini d'éléments en interaction. Les éléments considérés sont les *sommets* ou *nœuds* du graphe. Les interactions entre ces sommets sont matérialisées par les *arêtes* du graphe.

Un graphe G est donc bien défini par la donnée du couple (V, E) tel que V est l'ensemble des sommets de G et $E \subset V \times V$ est l'ensemble des arêtes de G. L'arête $e \in E$ ayant pour extrémités les sommets u et v de V est souvent notée $e = uv$. On se limite ici à définir les graphes simples (une seule arête relie deux sommets) et non orientés (les arêtes de G ne sont pas dirigées). On peut remarquera cependant qu'un graphe non orienté peut être considéré comme un graphe orienté tel que pour toute arête de u vers v, l'arête de v vers u appartient à E.

L'exemple du graphe G_5 est présenté 10.1 pour illustrer les définitions données ci-après :

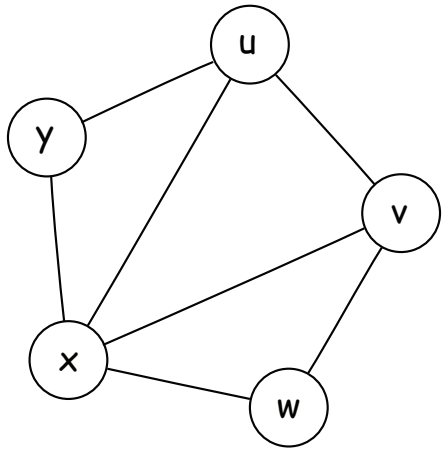

FIGURE *10.1* – *Le graphe G_5*

Un graphe a n sommets est dit d'*ordre* n tandis que sa *taille* est le nombre de ses arêtes. G_5 est d'ordre 5 et de taille 7.

Deux sommets reliés par une arête sont dits *adjacents* ou *voisins*. Le *degré* d'un sommet est le nombre de ses voisins. Les sommets u et v de G_5 sont voisins. v est de degré 3 et w est de degré 2.

Une *chaine* est une suite de sommets reliés par des arêtes et un *cycle* est une chaine dont les extrémités coïncident. On dit qu'un graphe est *connexe* si et seulement si il existe une chaine reliant toute paire de sommets. (u, v, w) est une chaine de G_5, (x, y, u) est un cycle et G_5 est connexe.

Les graphes utilisés dans ce travail sont non orientés, connexes et sans cycles. Ces graphes particuliers sont des *arbres* non orientés.

Le graphe A_5 présenté 10.2 est un arbre d'ordre 5.

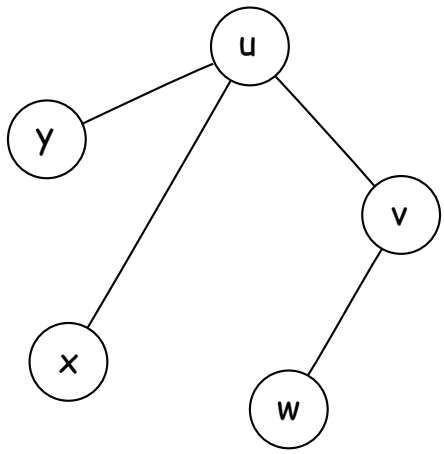

FIGURE *10.2* – *L'arbre A_5*

Les sommets de degré 1 sont les feuilles de l'arbre. A_5 possède 3 feuilles w, x et y.

La manipulation des arbres est rendue plus aisée par le choix d'une *racine*. Il s'agit d'un nœud de l'arbre qui sert de repère dans l'exploration des *branches*, chaines ayant pour extrémités la racine et une feuille. Ce choix est arbitraire dans le cas des arbres non orientés tandis que dans les arbres orientés, la racine est l'unique nœuds sans prédécesseur de l'arbre. Si l'on choisit u pour racine de l'arbre A_5, cet arbre possède alors trois branches (u, v, w), (u, x) et (u, y).

Un arbre est *étiqueté* si à chacun de ses sommets est attribuée une étiquette issue d'un ensemble fini de symboles. A_5 est étiqueté par l'ensemble $\{u, v, w, x, y\}$.

Pour être à même de comparer et de transformer des arbres étiquetés il est nécessaire de définir les opérations réalisables sur les nœuds de ces arbres. Ces opérations sont généralement de trois types : suppression, insertion et renommage. La donnée d'un ensemble d'arbre étiquetés et de ces trois opérations permet de définir plusieurs distances entre les arbres (alignement, édition) ainsi que de considérer les problèmes d'inclusion (Bille, 2005).

L'emploi d'arbres étiquetés pour représenter les connaissances syntaxyco-sémantiques associées à une proposition a été présenté dans le chapitre 1. L'usage de ces structures est également privilégié dans la manipulation des fichiers de données au format XML (*eXtensible Markup Language*). Ce langage permet de décrire des données sous forme arborescente à partir d'une structure préalablement définie. La grande variété de structures des arbres XML renouvelle l'intérêt pour la maîtrise des transformations d'arbres étiquetés. En effet, ces transformations conditionnent la communication entre les applications utilisant des données XML . Elles sont indispensables à l'utilisation des données du Web.

Les opérations de transformation sont centrales dans les taches de classification d'arbres et de découverte de motifs fréquents. Les travaux de (Candillier, 2006) adaptent différentes techniques de clustering à la classification de documents XML . Des approches algorithmiques voisines de celles que nous avons développées (voir chapitre 11) sont proposées par (Candillier et al., 2007) dans le contexte de la fouille de documents XML .

Les transformations peuvent être réalisées par des programmes dédiés à chaque application en utilisant par exemple le langage de transformation XSLT (*eXtensible Stylesheet Language Transformations*) ou des langages généralistes tels Perl ou Python. Cette approche est coûteuse et produit des solution spécifiques à chaque application, non évolutives et non génériques. Une alternative intéressante est proposée par (Jousse, 2007) en utilisant des techniques d'apprentissage supervisé : les opérations de transformation sont apprises à l'aide de modèles probabilistes à partir d'exemple d'arbres XML originaux et transformés.

Au cours des transformations, les décisions à prendre lors de la décomposition d'un arbre ou de la recomposition de branches peuvent donc se baser sur les opérations observées dans un corpus d'apprentissage. Dans ce travail, l'approche adoptée pour prendre en compte ces observations fait intervenir des techniques de classification automa-

tique supervisée. Des classifieurs à base de machines à vecteurs supports ou séparateurs à vaste marge (SVM) sont utilisés. Les notions de base permettant d'appréhender leur fonctionnement sont présentées dans la partie suivante 10.3.

10.3 Séparateurs à vaste marge

La classification a pour but de regrouper des objets de même nature en fonction de certaines de leurs caractéristiques. Chaque groupe d'objets forme une *classe*. Dans le contexte de la classification automatique, un *classifieur* désigne un algorithme permettant d'attribuer une classe à un objet à partir de l'observation de ses caractéristiques.

Les méthodes de classification non supervisée cherchent à partitionner l'ensemble des objets en groupes d'objets similaires sans qu'aucune partition *apriori* ne soit fournie. Ces méthodes, utilisées parfois en classification sémantique comme l'analyse latente sémantique (Bellegarda, 2007), ne permettent toutefois pas une extraction fine des composants. Elles sont adaptées à des applications de routage d'appels ou de classification de phrases.

Dans les méthodes de classification supervisée, l'ensemble des classes est fixé. L'application de ces méthodes à l'analyse sémantique est proposée par (Pradhan et al., 2004). Deux classes au moins sont définies et la répartition des données d'apprentissage au sein de ces classes est connue, ce qui justifie l'appellation "*supervisée*" de cette classification. Une donnée dont la classe d'appartenance est connue est souvent qualifiée de donnée *étiquetée*.

On dispose d'un ensemble X de n données étiquetées et d'un ensemble fini U de k classes. Chaque donnée $x_{i\in[\![1,n]\!]}$ est caractérisée par p caractéristiques et par sa classe $u_i \in U$. Le problème de classification consiste à prédire la classe de toute nouvelle données x en s'appuyant sur la connaissance des données de X. Procédant par induction puisqu'ils prédisent une connaissance plus générale à partir d'un ensemble de cas particuliers, les classifieurs produits dans ce contexte ont donc de bonnes capacités de généralisation.

Les données sont décrites vectoriellement dans un espace de Hilbert [1] de dimension p. Dans ce travail, la classification opérée est binaire : $U = \{-1, 1\}$. Les données sont étiquetées *positives* si elles sont de classe 1 et *négatives* si elles sont de classe -1.

Quand le problème de classification n'est pas linéairement séparable dans l'espace originel, il peut le devenir en réalisant un *déplacement* des données dans un espace de dimension plus élevée (Cover, 1965).

Dans l'exemple donné en 10.3, l'espace initial de représentation des données est $\mathbb{R}^2$ dans lequel les données d'entraînement ne sont pas linéairement séparables. Après déplacement dans $\mathbb{R}^3$, la séparation linéaire des données est réalisée par un hyperplan de $\mathbb{R}^3$.

1. Un espace de Hilbert est un espace vectoriel muni d'un produit scalaire, complet pour la norme associée. $\mathbb{R}^p$ muni du produit scalaire est un espace de Hilbert.

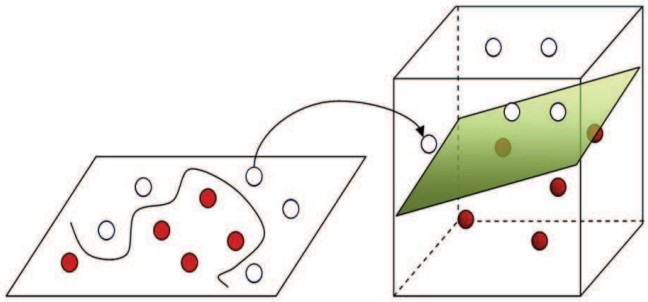

FIGURE 10.3 – *Déplacement de l'espace de représentation vers un espace de dimension supérieure*

Si le problème de classification est linéairement séparable, il existe une famille infinie de formes linéaires discriminantes qui peut lui être associée.

Toute forme de cette famille s'écrit $C(x) = \vec{w}.\vec{x} + w_0$ et $\forall i \in [\![1, n]\!]$ on a :

$$C(x_i) > 0 \Rightarrow u_i = 1$$
$$\text{et}$$
$$C(x_i) < 0 \Rightarrow u_i = -1$$

$$\text{soit} : \forall i \in [\![1, n]\!], \quad u_i C(x_i) > 0.$$

Il existe ainsi une infinité d'hyperplans capables de séparer les données positives des données négatives, chacun de ces hyperplans étant le noyau d'un forme linéaire discriminante associée au problème de classification.

Soit H un hyperplan séparateur d'équation $y = \vec{w}.\vec{x} + w_0$ ($\vec{w}$ normal à H). Soient H^+ et H^- les hyperplans parallèles à H contenant respectivement les éléments positifs et négatifs de X les plus proches de H.

La figure 10.4 illustre cette situation en dimension 2. Les données positives sont représentées en noir tandis que les données négatives sont en clair. H sépare les données positives des négatives, H^+ et H^- sont tous deux parallèles à H. A et B appartiennent respectivement à H^+ et H^- tels que la distance AB est minimale.

Un classifieur à base de séparateurs à vaste marge (SVM) linéaires détermine H tel que la distance entre H^+ et H^-, appelée la *marge*, est maximale. Les SVM font partie des méthodes à noyaux, inspirées de la théorie mathématique de l'apprentissage développée depuis les années 1960 par Vapnik et Chervonenkis (théorie VC) (Vapnik, 1995, 1998).

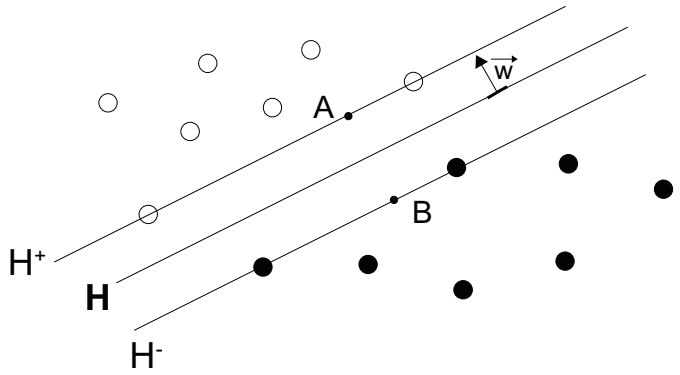

FIGURE *10.4* – *Schéma représentant la séparation de données par un hyperplan H*

Le problème étant linéairement séparable, on peut choisir :
$H^+ : \quad \vec{w}.\vec{x} + w_0 = 1$
$H^- : \quad \vec{w}.\vec{x} + w_0 = -1$ sous les contraintes $\forall i \in [\![1, n]\!] \qquad u_i C(x_i) \geq 1.$

La distance de tout point $x^- \in H^-$ à H est alors $d(x^-, H) = \dfrac{|\vec{w}.\vec{x^-} + w_0|}{\|\vec{w}\|}$ et de même la distance de tout point $x^+ \in H^+$ à H est alors $d(x^+, H) = \dfrac{|\vec{w}.\vec{x^+} + w_0|}{\|\vec{w}\|}$.

La marge λ s'écrit alors : $\lambda = \dfrac{2}{\|\vec{w}\|}$.

Maximiser λ sous les contraintes $\forall i \in [\![1, n]\!] \qquad u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$ revient donc à minimiser $\|\vec{w}\|$ ou encore $\dfrac{1}{2}\|\vec{w}\|^2$ sous les mêmes contraintes.

La recherche de l'hyperplan optimal se ramène à résoudre le problème d'optimisation sur $\vec{w}$ et w_0 :

Minimiser	$\frac{1}{2}\|\vec{w}\|^2$	
sous les contraintes	$\forall i \in [\![1, n]\!]$	$u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$

Le problème ainsi énoncé en forme primale impose une résolution en dimension $p + 1$, les données étant décrites dans $\mathbb{R}^p$, ce qui est d'autant plus complexe que p est grand. Cela compromet l'obtention de solution dans l'espace de grande dimension dans lequel les données ont été projetées pour obtenir un problème de classification linéairement séparable.

L'expression du problème d'optimisation dans sa forme duale permet de contourner cet écueil. Les contraintes du problème étant toutes linéaires, on peut appliquer la méthode des multiplicateurs de Lagrange pour transformer le problème d'optimisation sous contraintes en un problème d'optimisation sans contrainte ayant la même solution. L'application de cette méthode est détaillée dans l'Annexe D.

Ainsi, l'utilisation de la méthode de Lagrange permet de démontrer que **seules les données correspondant aux vecteurs supports sont utiles à l'apprentissage**.

Parmi les évolutions récentes des méthodes à base de SVM, il est intéressant de signaler l'introduction des modèles *soft margin* pour lequels la contrainte de marge est assouplie.

Cette contrainte $u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$, utilisée dans le modèle classique précédement présenté, devient $u_i(\vec{w}.\vec{x_i} + w_0) \geq 1 - \xi_i$ avec ξ_i proche de 0 variables d'erreurs.

L'introduction de ces variables permet de séparer linéairement les données au mieux tout en ignorant quelques exemples mal classés.

Dans le cas de problèmes non linéairement séparables dans l'espace initial, il est intéressant de remarquer que l'heuristique de déplacement vers un espace de grande dimension est indépendante du choix de l'algorithme de classification. Cependant, les classifieurs à base de SVM sont particulièrement bien adaptés à cette approche.

En effet, la classification dans l'espace de dimension supérieure nécessite seulement la connaissance :
- de la fonction de déplacement Φ (non linéaire) ;
- du mode de calcul des produits scalaires dans l'espace d'arrivée en fonction des vecteurs de l'espace initial ($\Phi(\vec{x}).\Phi(\vec{y})$).

Si l'on suppose l'existence d'une fonction **noyau** K telle que $\Phi(\vec{x}).\Phi(\vec{y}) = K(\vec{x}, \vec{y})$, il n'est plus nécessaire de connaitre la fonction de déplacement Φ. L'*astuce du noyau* (*Kernel Trick*) est attractive puisqu'elle permet d'utiliser des noyaux variés [2].

En conclusion, les méthodes à base de SVM permettent de traiter des problèmes de grande dimension. Essentiellement dépendantes des vecteurs supports, elles produisent des résultats pertinents même si les données d'apprentissage sont peu nombreuses. Elles offrent ainsi un bon compromis entre capacité de généralisation et complexité.

De nombreuses librairies libres implémentent les méthodes à base de SVM [3].

10.4 Conclusion

La notion d'arbre définie dans ce chapitre permet de modéliser et de manipuler les fragments sémantiques. Le chapitre suivant expose comment les opérations de composition des fragments produits par les DBN sont effectuées sur des structures d'arbres.

Pour appliquer les opérations de composition, nous proposons une approche heuristique et un algorithme de décision. Notre algorithme de décision repose sur l'utilisation des classifieurs SVM. Le modèle théorique des SVM, décrit dans ce chapitre, met en évidence leur capacité discrimination. En effet, la dépendance des paramètres de ces modèles aux seuls vecteurs supports a pour avantage qu'un ensemble de données

2. Tout noyau respectant la condition de Mercer est admissible(Vapnik, 1998). Cette condition s'écrit $\forall g$ tq $\int g(x)^2 dx$ est finie, $\iint K(x,y)g(x)g(y)dxdy \geq 0$ et elle garantit l'existence d'une solution au problème quadratique duale

3. Une liste non exhaustive de ces librairies peut être consultée à l'addresse `http://www.support-vector-machines.org/SVM_soft.html`.

d'apprentissage de taille restreinte ne compromet pas le niveau de performance de ces méthodes.

Approche heuristique et algorithme de décision à base de SVM sont présentés dans le chapitre suivant 11. L'utilisation des SVM dans notre contexte applicatif y est également détaillée.