Analyse formelle de concepts incertains pour l'analyse d'un questionnaire d'évaluation des enseignements

G. Petiot

Analyse formelle de concepts incertains pour l'analyse d'un questionnaire d'évaluation des enseignements

G. Petiot[1]

[1] UR CERES, Institut Catholique de Toulouse, 31 rue de la Fonderie, 31068, Toulouse

guillaume.petiot@ict-toulouse.fr

Résumé

L'Analyse Formelle de Concepts est une méthode d'analyse des données. Plusieurs travaux ont démontré qu'il est possible de compléter l'opérateur de Galois de l'AFC par de nouveaux opérateurs issus de la théorie des possibilités. Il est aussi possible de modéliser les incertitudes avant de les propager lors du calcul des concepts formels. Nous allons nous focaliser sur ce dernier point dans le cadre de cette étude. Nous proposons une expérimentation de l'AFC pour l'extraction de connaissances concernant un questionnaire d'évaluation des enseignements d'un cours de professionnalisation en licence. Certaines questions sont des questions ouvertes dont les réponses sont formulées librement. Afin de pouvoir les exploiter dans l'analyse du questionnaire nous réalisons une classification des réponses. Lors de la classification, des incertitudes peuvent apparaître. La méthode que nous proposons permet de prendre en compte ces incertitudes et de calculer une mesure de nécessité pour chaque concept formel. Ensuite, nous utilisons des requêtes pour extraire les concepts formels incertains et nous visualisons le résultat en utilisant un outil de visualisation.

Mots Clef

Analyse formelle de concepts, théorie des possibilités, traitement du langage naturel, réseaux de neurones, fouille de données.

Abstract

Formal Concept Analysis is an approach of data analysis. Several studies have demonstrated that it is possible to complete the Galois operator of the FCA by new operators from possibility theory. Thus, it is possible to model uncertainties before propagating them during the calculation of formal concepts. In this study, we will focus our interest on this last point. We propose experimentation of FCA for the knowledge extraction concerning a satisfaction questionnaire for a course of professionalisation in bachelor. Some of the questions are open questions, which require free answers. In order to be able to use them in the analysis of the questionnaire, we carry out a classification of the answers. During the classification, uncertainties may appear. The proposed method allows us to take into account these uncertainties and to compute a necessity measure for each formal concept. Then, we will use queries to extract uncertain formal concepts and we visualize the result by using a visualization tool.

Keywords

Formal concept analysis, possibility theory, natural language processing, neural networks, data mining.

1 Introduction

La réalisation d'un questionnaire d'évaluation des enseignements à la fin d'un cours à l'université permet de mieux connaître les étudiants et leurs attentes. Elle permet aussi d'évaluer la qualité de l'approche pédagogique, des ressources, des évaluations,... De nombreuses universités en proposent déjà en utilisant par exemple des outils comme Moodle. L'évaluation des enseignements contribue à l'amélioration continue des formations. Il est assez simple de réaliser un questionnaire, par contre son exploitation peut demander des traitements complexes bien connus en fouille de données. Il est aussi possible de proposer des réponses libres afin que les étudiants puissent s'exprimer librement. On peut par la suite réaliser un traitement du langage naturel ou une fouille de texte pour extraire une catégorisation des réponses. Il est ensuite nécessaire d'utiliser une approche permettant d'extraire une synthèse des réponses données pour en faciliter l'exploitation. Les statistiques descriptives et les tests statistiques sont souvent utilisés. Cependant, les tests statistiques peuvent concerner des hypothèses parfois mal connues. De plus l'interprétation que l'on peut faire des statistiques peut être difficile si l'on n'est pas expert dans ce domaine. L'AFC est une alternative aux approches statistiques [3]. Cette méthode connue en analyse des données a pour objectif l'extraction d'un ensemble ordonné de concepts formels. Un concept formel étant composé d'un ensemble d'objets et d'un ensemble d'attributs partagés par ces objets. L'analyse des concepts formels est beaucoup plus facile et intuitive que des statistiques et ne présente pas de perte d'informations. Plusieurs travaux existent concernant des applications de l'AFC au

domaine de l'éducation [1, 11, 15] ou l'analyse des réseaux sociaux [21]. Cependant, on retrouve rarement une prise en compte de l'incertitude dans les traitements. Pourtant, cela est possible, comme le montrent certains travaux de recherche qui proposent une généralisation de l'AFC en utilisant la théorie des possibilités [7, 8, 9, 6, 19, 17]. Nous proposons dans cet article une expérimentation de l'AFC qui prend en compte les incertitudes. Notre application vise à analyser les réponses d'un questionnaire de satisfaction proposé à des étudiants de licence pour un cours de PPP (Projet Personnel Professionnel). Les questions fermées ne présentent pas d'incertitude car les réponses possibles sont connues par avance. Nous avons aussi des questions ouvertes où l'étudiant peut librement s'exprimer sur ce qu'il a apprécié dans le cours. Pour analyser les réponses de ces questions un traitement du langage naturel est nécessaire. Nous proposons de réaliser une classification des réponses. Les méthodes de classification les plus utilisées sont décrites dans [13], on retrouve la méthode SVM, les classifieurs bayésiens naïfs, les k plus proches voisins, les arbres de décision, les réseaux de neurones,... Dans notre application, nous avons choisi d'utiliser un réseau de neurones. Le processus de classification génère des incertitudes sur l'appartenance d'une réponse à une classe. Ces incertitudes doivent être prises en compte dans l'AFC. L'approche que nous proposons peut se généraliser à tous les questionnaires qui comportent des réponses incertaines. Cela permet de prendre en compte l'ensemble des informations, des plus certaines au moins certaines, pour le calcul des concepts formels. On peut ensuite calculer une mesure de certitude pour chaque concept formel. Cet article est organisé de la façon suivante. Nous allons dans une première partie présenter la théorie des possibilités, ensuite nous allons présenter l'AFC et rappeler comment généraliser l'AFC en utilisant la théorie des possibilités. Dans la partie suivante nous allons décrire notre application et comment nous avons réalisé le traitement des questions ouvertes. Enfin, nous proposerons quelques résultats illustrant notre approche.

2 La théorie des possibilités

La théorie des possibilités a été développé par L. A. Zadeh [25] dans le prolongement de la théorie des ensembles flous. Cette théorie permet de modéliser à la fois les imprécisions associées aux connaissances mais aussi leur incertitude. Elle propose aussi une représentation de l'ignorance. Dans [10] on définit une distribution de possibilité π comme une représentation d'un état de connaissance. Par exemple si Ω est l'univers et π_x une distribution de possibilité d'une variable x définie de Ω dans $[0, 1]$, alors si $\pi_x(u) = 0$ alors $x = u$ est impossible. Si $\pi_x(u) = 1$ alors $x = u$ est possible. Par la suite, on appellera Π la mesure de possibilité, *N* la mesure de nécessité, Δ la mesure de possibilité garantie, et ∇ la mesure de nécessité potentielle. Ces mesures sont définies sur l'ensemble des parties de Ω noté $P(\Omega)$ dans $[0, 1]$:

$$\forall A \in P(\Omega), \Pi(A) = \sup_{x \in A} \pi(x). \quad (1)$$

$$\forall A \in P(\Omega), N(A) = 1 - \Pi(\neg A) = \inf_{x \notin A} 1 - \pi(x). \quad (2)$$

$$\forall A \in P(\Omega), \Delta(A) = \inf_{x \in A} \pi(x). \quad (3)$$

$$\forall A \in P(\Omega), \nabla(A) = 1 - \inf_{x \notin A} \pi(x). \quad (4)$$

La théorie des possibilités n'est pas additive mais maxitive :

$$\forall A, B \in P(\Omega), \Pi(A \cup B) = \max(\Pi(A), \Pi(B)). \quad (5)$$

Ces différentes mesures permettent de proposer d'autres opérateurs dans le cadre de l'AFC nécessaire pour l'analyse complète d'une situation. La mesure de possibilité garantie correspond à l'opérateur de Galois utilisé dans l'AFC.

3 L'analyse formelle de concepts

L'analyse formelle de concepts est une méthode d'analyse des données proposée par R. Wille [23] qui consiste à étudier les concepts en utilisant les treillis. L'intention et l'extension permettent de définir un concept. On retrouve ces notions aussi en philosophie. L'intention est tout simplement la définition du concept et l'extension l'ensemble des éléments sur lequel il s'applique. Mathématiquement un contexte est un triplet $(O, P, \Re)$ où $O = \{o_1, ..., o_n\}$ est l'ensemble des objets, $P = \{p_1, ..., p_m\}$ l'ensemble des propriétés possibles et $\Re$ une relation telle que $\Re \subseteq O \times P$. En fait si $(o, p) \in \Re$ alors l'objet o a la propriété p. Généralement on représente cela en utilisant une table dont les lignes sont les objets et les colonnes les propriétés, la relation quant à elle est représentée soit par un 0 si $(o, p) \notin \Re$ ou par un 1 si $(o, p) \in \Re$ et correspond à la valeur de la table. On peut définir une valuation $\vartheta(o, p)$ qui retourne la valeur de la table pour un objet o et une propriété p. Un concept formel de $(O, P, \Re)$ est un couple (X, Y) tel que $X \in O$ et $Y \in P$ tel que Y n'est en fait que l'ensemble des propriétés partagées par l'ensemble des objets de X. On peut le noter $X^{\uparrow} = Y$ ou $Y^{\downarrow} = X$. Par exemple pour le contexte formel ci-dessous, $(\{o_2, o_3, o_5\}, \{p_2, p_3\})$ et $(\{o_4, o_5\}, \{p_1, p_2\})$ sont deux concepts formels.

Objet	p_1	p_2	p_3
o_1	0	1	0
o_2	0	1	1
o_3	0	1	1
o_4	1	1	0
o_5	1	1	1

TABLE 1 – Exemple de contexte formel.

L'ensemble de tous les concepts formels de $(O, P, \Re)$ est noté $\beta(U, V, \Re) = \{(X, Y) | X^{\uparrow} = Y, Y^{\downarrow} = X\}$. Soit un ordre partiel $\leq$ tel que pour $(X_1, Y_1), (X_2, Y_2) \in$

$\beta(U, V, \Re)$ alors $(X_1, Y_1) \leq (X_2, Y_2)$ si $X_1 \subseteq X_2$ ou $Y_2 \subseteq Y_1$. On peut alors construire le treillis de concepts. Un treillis de concepts peut se visualiser en utilisant un diagramme de Hasse. Nous pouvons visualiser le treillis de concepts formels en utilisant l'outil ConExp (*http ://sourceforge.net/projects/conexp*) :

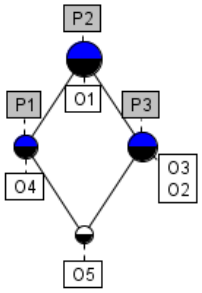

FIGURE 1 – Treillis de concepts formels.

On peut remarquer que la notation est réduite car on utilise l'héritage des propriétés et des objets. Lorsque des propriétés sont multivaluées, il est nécessaire de réaliser une transformation du contexte pour se ramener à un contexte formel binaire. Prenons l'exemple suivant :

Objet	Note	Qualité
o_1	5	faible
o_2	15	bonne
o_3	15	moyenne
o_4	12	faible
o_5	18	moyenne

TABLE 2 – Exemple de contexte multivalué.

Nous pouvons voir que la note est dans l'intervalle $[0, 20]$. On peut de ce fait proposer par exemple une catégorisation des valeurs selon trois classes. La première correspond à la classe faible pour des valeurs dans $[0, 7]$, la seconde est moyenne pour des valeurs dans $[8, 14]$, enfin la dernière est forte pour des valeurs dans $[15, 20]$. On retrouve ci-dessous un exemple de transformation en contexte formel binaire :

Objet	N_{faible}	$N_{moyenne}$	N_{forte}	Q_{faible}	$Q_{moyenne}$	Q_{forte}
o_1	1	0	0	1	0	0
o_2	0	0	1	0	0	1
o_3	0	0	1	0	1	0
o_4	0	1	0	1	0	0
o_5	0	0	1	0	1	0

TABLE 3 – Transformation du contexte multivalué en contexte binaire.

Jusqu'à présent les valeurs des propriétés étaient certaines cependant si ce n'est pas le cas on peut proposer d'utiliser la théorie des possibilités [25] pour prendre en compte les incertitudes comme le proposent les auteurs dans [6]. Ainsi, on peut avoir la distribution de possibilité $\pi_{o_p}(u)$ définie pour $u \in \Omega$ qui est la possibilité que la propriété p de l'objet o soit u. Il faut bien sûr que cette distribution de possibilité soit normalisée. Les auteurs proposent aussi d'étendre l'AFC en définissant quatre opérateurs qui s'inspirent de la théorie des possibilités que l'on peut rappeler brièvement. Si $\Re$ est une relation (ou un contexte) et si $R(o) = \{p \in P | (o, p) \in \Re\}$ et $R^t(p) = \{o \in O | (o, p) \in \Re\}$ alors pour S, un sous-ensemble de O, on a les opérateurs suivants :

— $(S)^{\Pi} = \{p \in P | R^t(p) \cap S \neq \oslash\}$
— $(S)^{N} = \{p \in P | R^t(p) \subseteq S\}$
— $(S)^{\Delta} = \{p \in P | R^t(p) \supseteq S\}$
— $(S)^{\nabla} = \{p \in P | R^t(p) \cup S \neq O\}$

Ci-dessous nous proposons un exemple d'application de ces opérateurs pour différents ensembles d'objets de la Table 1 :

S	$(S)^{\Pi}$	$(S)^{N}$	$(S)^{\Delta}$	$(S)^{\nabla}$
(o_1, o_2, o_3)	(p_2, p_3)	$(\oslash)$	(p_2)	$(\oslash)$
(o_2)	(p_2, p_3)	$(\oslash)$	(p_2, p_3)	(p_1, p_3)
(o_4, o_5)	(p_1, p_2, p_3)	(p_1)	(p_1, p_2)	(p_1, p_3)
(o_1, o_2, o_3, o_4)	(p_1, p_2, p_3)	$(\oslash)$	(p_2)	$(\oslash)$

TABLE 4 – Exemple d'application des opérateurs.

$(S)^{\Pi}$ est l'ensemble des propriétés possédées par au moins un objet de S. $(S)^{N}$ est l'ensemble des propriétés possédées seulement par les objets de S. $(S)^{\Delta}$ est l'ensemble des propriétés partagées par tous les objets de S. $(S)^{\nabla}$ est l'ensemble des propriétés qui ne sont pas satisfaites pour au moins un objet de $\bar{S}$. Le comportement de l'opérateur $(.)^{\Delta}$ est celui que l'on retrouve en général dans l'analyse formelle de concepts. Par la suite, nous allons utiliser cet opérateur. Dans l'article [6] les auteurs rappellent que ces opérateurs ont déjà été proposés sans toutefois faire référence à la théorie des possibilités. Ces opérateurs peuvent être représentés sous la forme d'un cube des oppositions pour l'AFC (*AffIrmo nEgO*) :

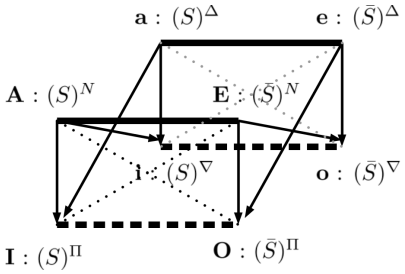

FIGURE 2 – Cube des oppositions.

Ces opérateurs sont définis pour un ensemble d'objets S. On peut facilement proposer des opérateurs équivalents pour des ensembles de propriétés notés $(.)^{-1\Pi}$, $(.)^{-1N}$, $(.)^{-1\Delta}$, $(.)^{-1\nabla}$ comme dans [6]. La prise en compte de l'imprécision pour les propriétés peut se faire en utilisant les ensembles flous comme dans [2]. Il y a par contre moins de travaux concernant la prise en compte de l'incertitude et donc de l'ignorance qui peut être partielle ou totale. La certitude est associée à la mesure de nécessité dans la théorie des possibilités. Cette mesure peut être intégrée à la table et

être propagée lors du calcul des concepts formels. Ainsi, si l'on dispose comme dans [9] d'une paire de mesure de nécessité $(\alpha(o,p), \beta(o,p))$ avec $\alpha(o,p) = N((o,p) \in \Re)$ et $\beta(o,p) = N((o,p) \notin \Re)$, on peut en déduire pour un contexte donné un ensemble de concepts formels. Il faut satisfaire la propriété $min(\alpha(o,p), \beta(o,p)) = 0$ de la théorie des possibilités. Les paires $(1,0)$ et $(0,1)$ représentent le fait qu'un objet a une propriété ou pas. Si $1 > max(\alpha(o,p), \beta(o,p)) > 0$, l'ignorance est partielle, et si l'on a $(0,0)$, l'ignorance est totale. On peut définir un contexte formel incertain comme ci-dessous :

$$\Re' = \{(\alpha(o,p), \beta(o,p)) | o \in O, p \in P\} \quad (6)$$

Pour cette première expérimentation nous allons transformer le contexte incertain en remplaçant les valeurs $(\alpha(o,p), 0)$ par 1 et $(0, \beta(o,p))$ par 0. On remplace les valeurs incertaines par des valeurs certaines. Par exemple, $(0.2, 0)$ peut être transformé en 1 et $(0, 0.6)$ en 0. On obtient ainsi un nouveau contexte formel pour lequel on peut facilement extraire les concepts formels. Une fois les concepts formels extraits, il est possible de calculer la mesure de nécessité d'un concept formel $C = (X, Y)$ donc sa certitude en utilisant la formule suivante :

$$N(C) = \min_{o \in X, p \in Y} N((o,p) \in \Re) \quad (7)$$

Nous présentons ci-dessous un exemple de contexte formel incertain pour illustrer ce calcul de certitude :

Objets	p_1	p_2	p_3
o_1	(0,1)	(1,0)	(0.2,0)
o_2	(0,0.5)	(1,0)	(1,0)
o_3	(0.5,0)	(1,0)	(0,0.9)
o_4	(1,0)	(1,0)	(0.8,0)
o_5	(1,0)	(1,0)	(1,0)

TABLE 5 – Exemple de contexte formel incertain.

En transformant ce contexte comme décrit précédemment nous obtenons :

Objets	p_1	p_2	p_3
o_1	0	1	1
o_2	0	1	1
o_3	1	1	0
o_4	1	1	1
o_5	1	1	1

TABLE 6 – Transformation du contexte formel incertain en contexte formel binaire.

Dans cet exemple on peut voir que $(\{o_1, o_2, o_4, o_5\}, \{p_2, p_3\})$, $(\{o_3, o_4, o_5\}, \{p_1, p_2\})$, $(\{o_4, o_5\}, \{p_1, p_2, p_3\})$ et $(\{o_1, o_2, o_3, o_4, o_5\}, \{p_2\})$ sont les concepts formels de ce contexte formel. On obtient pour ces concepts formels les résultats suivants pour le calcul de la certitude :

Concept formel	Certitude
$(\{o_1, o_2, o_4, o_5\}, \{p_2, p_3\})$	0.2
$(\{o_3, o_4, o_5\}, \{p_1, p_2\})$	0.5
$(\{o_4, o_5\}, \{p_1, p_2, p_3\})$	0.8
$(\{o_1, o_2, o_3, o_4, o_5\}, \{p_2\})$	1

TABLE 7 – Calcul de la certitude des concepts formels.

De nombreux algorithmes existent concernant le calcul des concepts formels [16]. Nous avons choisi pour notre expérimentation d'implémenter l'algorithme de Ganter *Next Closure* [12] qui fait partie des algorithmes les plus connus. Cet algorithme permet de trouver toutes les intentions (il peut aussi permettre de trouver toutes les extensions). Nous l'avons adapté au calcul des concepts formels incertains. Pour cela, nous proposons de modifier la fonction R pour un seuil de certitude μ qui devient $R_\mu(o) = \{p \in P | N((o,p) \in \Re) > \mu\}$ et $R^t_\mu(p) = \{o \in O | N((o,p) \in \Re) > \mu\}$. Ce qui donne deux nouveaux opérateurs que nous allons utiliser $(.)^{\Delta_\mu}$ et $(.)^{-1\Delta_\mu}$:

$$(S)^{\Delta_\mu} = \{p \in P | R^t_\mu(p) \supseteq S\} \quad (8)$$

$$(S)^{-1\Delta_\mu} = \{o \in O | R_\mu(o) \supseteq S\} \quad (9)$$

On définit aussi l'opérateur $\oplus_\mu$ de la fermeture tel que :

$$X \oplus_\mu i = ((X \cap \{p_1, ..., p_{i-1}\}) \cup \{p_i\})^{-1\Delta_\mu \Delta_\mu} \quad (10)$$

Enfin, nous définissons aussi l'opérateur de comparaison $<_i$ (ordre lexicographique). Si $X \in P$ et $Y \in P$ alors $X <_i Y$ si :

$$\begin{cases} p_i \in Y - X \text{ et} \\ X \cap \{p_1, ..., p_{i-1}\} = Y \cap \{p_1, ..., p_{i-1}\} \end{cases} \quad (11)$$

De plus on a $X < Y$ s'il existe un i pour lequel la relation $X <_i Y$ est vérifiée. L'algorithme de calcul des concepts formels incertains est le suivant :

Algorithme 1 : NextClosure incertain

Entrée : Un contexte incertain R; Le seuil de certitude μ
Sortie : L'ensemble des intentions noté I

1 **début**
2 $V = \oslash^{-1\Delta_\mu \Delta_\mu}$
$Sauver(V)$
tant que $V \neq P$ **faire**
3 **pour** $i \leftarrow |P|$ **à** 1 **faire**
4 $V^+ = V \oplus_\mu i$
si $V <_i V^+$ on sort de la boucle.
5 $Sauver(V^+)$
$V \longleftarrow V^+$

L'ensemble des concepts formels est noté $\beta(U, V, \Re) = \{(X^{-1\Delta}, X)|X \in I\}$. Si on utilise un opérateur de comparaison $\leq$ entre les concepts formels alors on a un treillis de concept.

4 Traitement du langage naturel

Dans le questionnaire de notre expérimentation nous allons traiter une question ouverte dont les réponses peuvent être données librement contrairement aux réponses des questions fermées qui sont limitées à une liste de propositions. Nous avons toutefois une idée des réponses possibles pour cette question ouverte car elle porte sur les parties du cours que les étudiants préfèrent. De ce fait, nous pouvons réaliser une classification des réponses à partir d'une description textuelle de la classe et d'échantillons issus des réponses. Nous avons choisi de réaliser la classification en utilisant un réseau de neurones. La première étape consiste à utiliser une méthode de fouille de texte pour extraire les corpus des descriptions des classes, des échantillons et des autres réponses. Ensuite, nous réalisons un traitement sur ces corpus pour changer la casse, éliminer les caractères indésirables, la ponctuation, les chiffres, et les mots non voulus car ne donnant aucune information sur la réponse. Ci-dessous, nous présentons le processus de traitement des corpus :

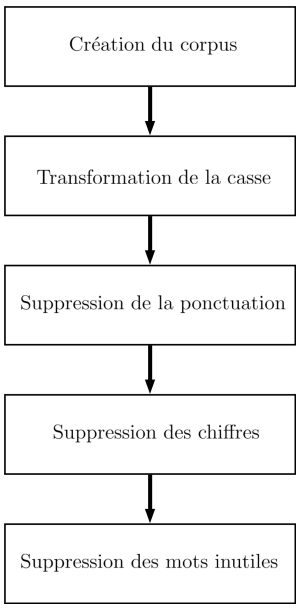

FIGURE 3 – Processus de traitement des corpus.

Ensuite, nous calculons des Matrices Documents-Termes (DTM en anglais pour *Document-Term Matrix*) pour les réponses et les descriptions des classes. La classification utilisera pour l'apprentissage la matrice MDT de la description des classes et des échantillons. La classification se fera en utilisant le dictionnaire des mots utilisés pour l'apprentissage. Un problème important dans le traitement du langage naturel est que lorsque l'on traite des réponses libres dans un sondage il y a très souvent des fautes de syntaxes, et d'orthographe. L'idée que nous avons eue est d'utiliser une mesure de similarité entre les mots lors du calcul de la matrice MDT et lors de la classification. Plusieurs mesures existent pour comparer deux mots [4, 14], les plus connues sont des distances. On peut citer par exemple la distance de Levenshtein, la distance de Jaccard, la distance de Damerau-Levenshtein, la distance de Hamming, la distance de la plus longue sous chaîne commune, la distance de Smith-Waterman ou la distance de Jaro-Winkler [24]. Pour que ces distances deviennent des mesures de similarité normalisées il faut réaliser une transformation afin qu'elles vérifient la définition suivante :
Une mesure de similarité normalisée entre deux mots m_1 et m_2 est une fonction notée λ de $\Omega \times \Omega \rightarrow [0, 1]$ où Ω est l'ensemble des mots possibles. Cette fonction doit respecter les propriétés suivantes :

— $\lambda(m_1, m_2) \in [0, 1]$
— $\lambda(m_1, m_2) = \lambda(m_2, m_1)$
— $\forall m_2, \lambda(m_1, m_1) \geq \lambda(m_1, m_2)$
— $\lambda(m_1, m_1) = 1$

La distance de Jaro-Winkler vérifie cette définition, de plus elle donne de bons résultats comme le décrit l'auteur de [4] et ne nécessite pas de transformation car elle renvoie des résultats dans $[0, 1]$. Cette distance se calcule en utilisant la distance de Jaro entre les mots m_1 et m_2 :

$$d_J(m_1, m_2) = \frac{1}{3}\left(\frac{\chi}{|m_1|} + \frac{\chi}{|m_2|} + \frac{\chi - \tau}{\chi}\right) \qquad (12)$$

Avec $|m_i|$ la longueur du mot i, χ le nombre de caractères correspondant (que l'on retrouve dans les deux mots à une distance inférieure à $\left\lfloor \frac{\max(|m_1|,|m_2|)}{2} \right\rfloor - 1$), τ le nombre de transposition (les caractères inversés). La distance de Jaro-Winkler est la suivante :

$$d_{JW}(m_1, m_2) = d_J(m_1, m_2) + \alpha\beta(1 - d_J(m_1, m_2)) \quad (13)$$

Avec α la longueur du préfixe commun aux deux mots avec un maximum de 4 caractères et β un coefficient que l'on prend en général égal à 0.1.
Prenons l'exemple suivant qui concerne l'écriture du mot *intelligence* et comparons la distance de Jaro-Winkler pour différents mots :

Mot	Mesure de Jaro-Winkler
Intelligence	1.0
Inteligence	0.95
Inelligence	0.89
Intelijence	0.92
Hasard	0

TABLE 8 – Comparaison de mots proches de *intelligence* à l'aide de la mesure de Jaro-Winkler.

On peut remarquer que lorsque les mots ressemblent au mot *intelligence* alors la mesure renvoie une valeur proche de 1. Plus le mot est différent, plus la mesure de ressemblance sera proche de 0. Si la mesure n'est pas assez proche

de 1 alors on peut considérer que les mots sont différents. Donc si d_{JW}(mot n°1, mot n°2)$< \eta$ alors mot n°1 $\neq$ mot n°2. Dans les études existantes [20, 5, 18] η est souvent pris entre 0.7 et 0.9. Nous choisissons de prendre $\eta = 0.85$. Avec cette valeur de η le mot *hasard* présent dans le tableau ci-dessus ne serait pas considéré comme proche du mot *intelligence*. Ensuite, nous construisons la matrice MDT. Par exemple, pour notre application nous avons obtenu la matrice MDT suivante :

Etudiants \ Mots	intelligences	gardner	questionnaire	proust	cv	europass	blason	projet	professionnel	personnalise	...
Etudiant n°1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
Etudiant n°2	1.0	0.96	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
Etudiant n°3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.93	...
Etudiant n°4	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
Etudiant n°5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	...
...	...	...	...	...	...	...	...	...	...	...	...
Etudiant n°144	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.92	...

TABLE 9 – Exemple de matrice MDT pour les réponses des étudiants.

L'étape suivante est la classification des réponses par un réseau de neurones. Ce réseau prend en entrée les poids de la matrice MDT des réponses en ne considérant que les mots présents dans la matrice MDT des échantillons. Ensuite on propage ces valeurs dans le réseau de neurones. En sortie nous avons un degré d'appartenance pour chaque classe. Ci-dessous nous présentons le réseau de neurones :

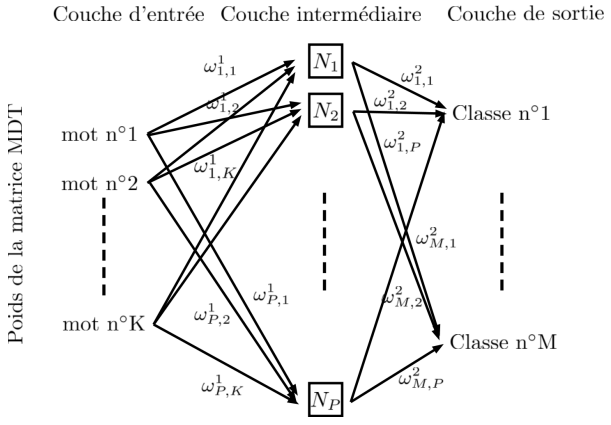

FIGURE 4 – Le réseau de neurones.

L'apprentissage du réseau de neurones est fait en utilisant une rétro-propagation du gradient. Pour cela, nous construisons des groupes d'échantillons qui comprennent des échantillons de chaque classe. Le résultat de la classification peut être vu comme une mesure de possibilité après renormalisation pour chaque classe. Nous pouvons aussi calculer une mesure de nécessité.

5 Expérimentation

L'expérimentation que nous proposons concerne l'analyse d'un questionnaire d'évaluation d'un cours de professionnalisation en licence. 144 étudiants ont répondu au questionnaire. Dans ce questionnaire, il y avait des questions fermées et des questions ouvertes. Pour simplifier le traitement nous prenons en compte uniquement une question ouverte. Celle qui nous semble la plus importante et qui porte sur les parties abordées en cours que les étudiants ont le plus appréciés. Notre objectif est de réaliser une classification des réponses en utilisant le réseau de neurone décrit dans la partie précédente. Nous avons décidé d'associer une classe par partie et de décrire chaque partie du cours avec une phrase. Au final, nous avons 8 classes. Nous avons aussi sélectionné 3 échantillons pour chacune des 8 classes en plus de la description de la classe pour constituer des groupes d'échantillons. Ensuite, nous avons calculé les matrices MDT pour les groupes d'échantillons. Nous en déduisons le nombre de mots en entrée du réseau de neurone $N = 45$, le nombre de classe $M = 8$, et nous choisissons le nombre de neurones pour la couche intermédiaire $P = 34$ (75% de 45 comme décrit dans [22]).
Ensuite, nous avons réalisé l'apprentissage des poids du réseau de neurones avant de faire la classification. Afin de vérifier notre approche nous avons calculé la matrice de confusion en considérant l'ensemble des réponses. Pour cela, nous avons associé pour chaque réponse une classe. La matrice de confusion représente suivant les lignes, les classes des réponses, et suivant les colonnes les classes prédites.

Réelle \ Prédite	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
C_1	**35**	0	0	0	0	0	0	0
C_2	0	**29**	0	0	0	0	0	0
C_3	0	0	**25**	0	0	0	0	0
C_4	0	0	0	**20**	2	0	0	0
C_5	0	0	0	0	**11**	0	0	0
C_6	0	0	0	0	1	**7**	0	0
C_7	0	0	0	0	0	0	**4**	0
C_8	0	0	0	0	2	0	0	**8**

TABLE 10 – Matrice de confusion.

Nous présentons ci-dessous le calcul pour chaque classe de la précision, du rappel et du score F1 :

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	**Moyenne**
Précision	1	1	1	1	0.68	1	1	1	**0.96**
Rappel	1	1	1	0.90	1	0.87	1	0.8	**0.95**
Score F1	1	1	1	0.95	0.81	0.93	1	0.88	**0.95**

TABLE 11 – Calcul de la précision, du rappel et du score F1.

Les erreurs de classification proviennent pour la plupart d'une confusion avec la classe C_5 bien que l'ensemble des réponses identifiées comme appartenant à cette classe soient bien classées. Cette confusion est principalement

due à des formulations ambiguës lors des réponses qui utilisent des mots proches de ceux rencontrés dans les échantillons de la classe C_5.
A partir du résultat de la classification nous calculons pour toutes les réponses une mesure de possibilité et de nécessité pour chaque classe. Parmi les résultats, nous avons observé des problèmes de quasi équipossibilité (4 réponses sur 144) dus à des réponses qui portent sur plusieurs classes. Deux solutions sont possibles pour ces réponses. Soit nous ne les prenons pas en compte dans l'analyse formelle de concepts. Soit nous tentons de les prendre en compte en partant de l'hypothèse que si la mesure de possibilité de plusieurs classes est proches de 1 pour une réponse alors la réponse peut être associée à plusieurs classes. Dans ce cas nous acceptons une erreur si l'équipossibilité représente un cas d'ignorance entre le choix de deux classes. Compte tenu du faible nombre de cas nous choisissons de ne pas prendre en compte ces réponses dans l'analyse formelle de concept (première solution). Cela revient à prendre μ différent de 0 dans notre algorithme de calcul des concepts formels. En prenant $\mu = 0.11$ nous ignorons les réponses anormales dans notre expérimentation.
Il y a en plus 32 autres questions fermées dont les réponses sont multivaluées. Nous avons transformé les données initiales qui sont multivaluées et réalisé un traitement des réponses libres en utilisant l'approche proposée dans la partie précédente. Le résultat est un contexte formel incertain dont les colonnes sont les réponses possibles (propriétés de l'AFC) et les lignes correspondent aux étudiants (objets de l'AFC). L'intégration du traitement de la question ouverte consiste à insérer autant de colonnes que de classes possibles. Les valeurs de la table pour ces colonnes sont une paire de mesures de nécessité. Ce qui donne au final une table de 160 colonnes et 144 lignes. Le contexte formel incertain obtenu est le suivant :

Etudiants	Classe n°1	Classe n°2	Classe n°3	Classe n°4	Classe n°5	...
Etudiant n°1	(0,1)	(0,1)	(0,1)	(0,1)	(0.5,0)	...
Etudiant n°2	(0.99,0)	(0,1)	(0,1)	(0,1)	(0,1)	...
Etudiant n°3	(0,1)	(0,1)	(0,1)	(0,1)	(0.99,0)	...
Etudiant n°4	(0.1,0)	(0,1)	(0,1)	(0,1)	(0,1)	...
Etudiant n°5	(0,1)	(0,1)	(0,1)	(0,1)	(0.91,0)	...
...	...	...	...	...	...	...
Etudiant n°144	(0,1)	(0,1)	(0,1)	(0,1)	(0.99, 0)	...

TABLE 12 – Contexte formel incertain obtenu.

Si nous recherchons les concepts formels qui représentent le plus les réponses des étudiants, nous pouvons définir un score. Si $\beta(U, V, \Re)$ est l'ensemble des concepts formels et (X, Y) un concept formel avec X son extension et Y son intention, alors le score peut-être le suivant :

$$S = \frac{|X| + |Y|}{\max\limits_{(u,v)\in\beta(U,V,\Re)} |u| + |v|} \tag{14}$$

Compte tenu du nombre important de concepts formels, il est nécessaire de pouvoir filtrer les concepts formels pour ne visualiser que ceux qui comportent des informations importantes pour l'utilisateur. Pour cela nous avons réalisé deux traitements sur les concepts formels. Le premier est l'utilisation d'une requête pour extraire les concepts formels qui correspondent à ce que l'on recherche. Le second est la visualisation du résultat. Le résultat de la requête sera représenté par un graphe directionnel. L'orientation des arêtes étant définie par la relation d'ordre partiel entre les concepts formels. La taille de chaque noeud sera proportionnelle au score du concept formel. L'incertitude des concepts sera représentée par un dégradé de couleur (Rouge, Jaune, Vert) allant du rouge pour les moins certains au vert pour les certains. Plusieurs outils de visualisation des données existent. Nous avons choisi d'utiliser l'outil Gephi qui est gratuit et qui propose un certain nombre de fonctionnalités. Les résultats des requêtes sont générés dans des fichiers Excel pour faciliter l'import dans Gephi. Nous proposons pour illustrer ces traitements un exemple de requête :

Q=**SELECT** c **FROM** $\beta(U, V, \Re)$
WHERE
($c.C_1$ **OR** $c.C_2$ **OR** $c.C_3$ **OR** $c.C_4$ **OR** $c.C_5$ **OR** $c.C_6$ **OR** $c.C_7$ **OR** $c.C_8$)
AND
$Score(c) \geq 0.1$
AND
$Card(c.X) \geq 15$

Avec $c = (X, Y)$ un concept formel de $\beta(U, V, \Re)$, $Card$ le nombre de propriétés ou d'objets du concept. Enfin, $c.C_i$ est vrai si la propriété C_i qui représente la classe n°i est présente dans Y sinon la valeur retournée est fausse. Le circuit logique qui permet l'évaluation de la requête Q pour chaque concept formel est le suivant :

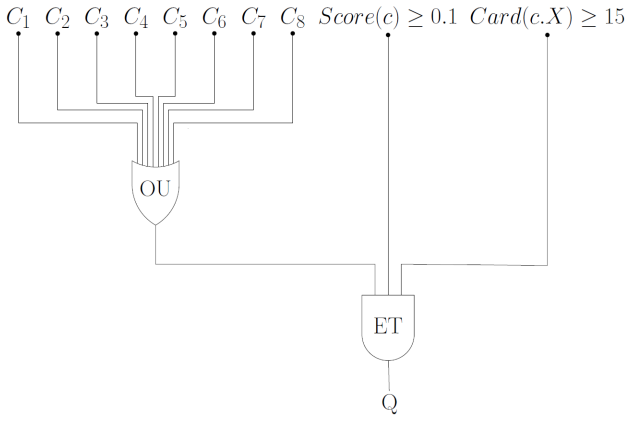

FIGURE 5 – Le circuit logique associé à la requête Q.

Le résultat de la requête est donc une évaluation de ce circuit logique pour chaque concept formel. Compte tenu des opérateurs booléens utilisés, le résultat de chaque évaluation est soit vrai si le concept formel est compatible avec la

requête soit faux dans le cas contraire. Par exemple, pour notre expérimentation nous obtenons le résultat suivant :

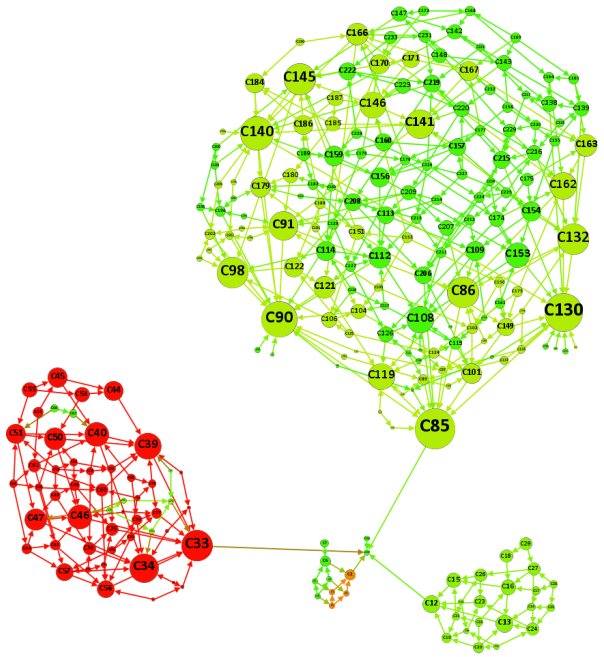

FIGURE 6 – Exemple de résultat d'une requête.

Dans cette figure, on peut voir l'incertitude des concepts formels mais aussi les concepts formels qui sont associés à de nombreuses réponses d'étudiants. Prenons l'exemple du concept formel C85 qui contient dans son intention la propriété associée à la classe n°1 qui représente la partie du cours concernant la théorie des intelligences multiples de H. Gardner et dans son extension 33 étudiants. Ce concept formel semble plus important que les autres car le noeud est d'un rayon supérieur. On peut voir qu'il présente une bonne certitude. La connaissance que nous pouvons extraire de ce concept formel est que de nombreux étudiants ont apprécié la partie du cours portant sur la théorie des intelligences multiples de H. Gardner. Nous pouvons généraliser cette approche d'évaluation des requêtes par des circuits logiques en utilisant des réseaux possibilistes en particulier des portes logiques incertaines (ET, OU, ...). Les réseaux possibilistes permettent de prendre en compte l'imprécision des connaissances que l'on souhaite prendre en compte dans les requêtes, ils permettent aussi de propager l'incertitude et d'obtenir un score de pertinence pour chaque concept formel qui serait la possibilité que le concept formel corresponde à ce que l'on recherche.

6 Conclusion

L'analyse formelle de concepts permet une analyse des données pour en extraire des connaissances. Cependant, il peut s'avérer que certaines données soient incertaines. La théorie des possibilités permet dans ce cas de prendre en compte les incertitudes et de les propager lors du calcul des concepts formels. Nous avons proposé d'utiliser des requêtes pour extraire les concepts formels. Ces requêtes peuvent porter sur les objets ou les propriétés mais aussi sur d'autres informations comme la cardinalité ou un score. Nous avons proposé d'utiliser un outil de visualisation pour mettre en évidence les concepts les plus intéressants en faisant ressortir leur incertitude et le score de chaque concept formel. En termes de perspective, nous souhaitons évaluer l'intérêt pour l'extraction de connaissances des différents opérateurs issus de la théorie des possibilités qui complètent l'opérateur de Galois de l'analyse formelle de concepts. Pour l'instant nous avons pris en compte que des critères booléens dans les requêtes qui peuvent être représentées par un circuit logique. Nous comptons expérimenter l'utilisation des portes logiques incertaines afin d'élaborer des requêtes traduisant des connaissances imprécises et incertaines. Nous comptons proposer un score de pertinence pour l'indexation des concepts formels. Il sera ainsi possible de classer les concepts formels pour faciliter la lecture des résultats.

Références

[1] M. A. Bedek, S. Kopeinik, B. Prünster, D. Albert. Applying the Formal Concept Analysis to Introduce Guidance in an Inquiry-Based Learning Environment. *IEEE 15th International Conference on Advanced Learning Technologies*, pp. 285-289, july, 2015.

[2] R. Bělohlávek. Concept lattices and order in fuzzy logic. *Annals of Pure and Applied Logic*, Vol. 128, Issues 1-3, pp. 277-298, 2004.

[3] R. Bělohlávek, V. Sklenar, J. Zacpal, E. Sigmund. Evaluation of questionnaires by means of formal concept analysis. *In : J. Diatta, P. Eklund, M. Liquiere (Eds.) : CLA 2007, Int. Conference on Concept Lattices and Their Applications*, Montpellier, France, pp. 100-111, October 24-26, 2007.

[4] P. Christen. A Comparison of Personal Name Matching : Techniques and Practical Issues. *Proceedings of the Sixth IEEE International Conference on Data Mining - Workshops, ICDMW '06*, IEEE Computer Society, Washington, DC, USA, pp. 290-294, 2006.

[5] W. W. Cohen, P. Ravikumar, S. E. Fienberg. A comparison of string distance metrics for name-matching tasks. *Proceedings of the 2003 International Conference on Information Integration on the Web*, pp. 73-78, 2003.

[6] D. Dubois, F. Dupin de Saint-Cyr, H. Prade. A Possibility-Theoretic View of Formal Concept Analysis. *Fundamenta Informaticae*, IOS Press, Vol. 75, Amsterdam, The Netherlands, pp. 195-213, 2007.

[7] D. Dubois, H. Prade. Possibility theory and formal concept analysis in information systems. *IFSA-EUSFLAT*, pp. 1021-1026, 2009.

[8] D. Dubois, H. Prade. Possibility theory and formal concept analysis : Characterizing independent sub-

contexts. *Fuzzy Sets and Systems*, Vol. 196, pp. 4-16, 2012.

[9] D. Dubois, H. Prade. Formal Concept Analysis from the Standpoint of Possibility Theory. *In : J. Baixeries, C. Sacarea, M. Ojeda-Aciego (eds) Formal Concept Analysis. ICFCA 2015*. Lecture Notes in Computer Science, Vol 9113. Springer, Cham, pp. 21-38, 2015.

[10] D. Dubois, H. Prade. *Possibility theory : An Approach to Computerized Processing of Uncertainty*, New York, Plenum Press, 1988.

[11] B. Fernandez-Manjon, A. Fernandez-Valmayor. Building Educational Tools Based on Formal Concept Analysis. *Journal of Education and Information Technologies*, Vol. 3, no. 3, pp. 187-201, 1 décembre, 1998.

[12] B. Ganter. Algorithmen zur Formalen Begriffsanalyse. *In B. Ganter, R. Wille, K. Wolf, (eds.), Beitrage zur Begriffsanalyse, Wissenschaftsverlag, Mannheim*, pp. 241-255, 1987.

[13] A. Hotho, A. Nürnberger, G. Paass. (2005). A Brief Survey of Text Mining. *LDV Forum - GLDV Journal for Computational Linguistics and Language Technology*. Vol. 20, pp 19-62, 2005.

[14] M. A. Jaro. Advances in record linking methodology as applied to the 1985 census of Tampa Florida. *Journal of the American Statistical Society*, Vol. 84, no. 406, pp. 414-420, 1989.

[15] M. D. Kickmeier-Rust, M. Bedek, D. Albert. Theory-based Learning Analytics : Using Formal Concept Analysis for Intelligent Student Modelling. *In H. R. Arabnia, D. de la Fuente, R. Dziegiel & E. B. Kozerenko (Eds.), Proceedings of the 2016 International Conference on Artificial Intelligence*, Las Vegas, Nevada, USA, pp. 97-100, 25-28 July 2016.

[16] S. O. Kuznetsov, S. A. Obiedkov. Comparing performance of algorithms for generating concept lattices. *J. Experimental & Theoretical Artificial Intelligence*, Vol. 14, pp. 189-216, 2003.

[17] L. Miclet, H. Prade, D. Guennec. Looking for Analogical Proportions in a Formal Concept Analysis Setting. *Conference on Concept Lattices and Their Applications*, Nancy, France, pp. 295-307, octobre, 2011.

[18] F. Mastjik, C. Varol, A. Varol. Comparison of Pattern Matching Techniques on Identification of Same Family Malware.*International Journal of Information Security Science (IJISS)*, Vol. 4, Issue 3, pp. 104-111, September, 2015.

[19] E. Navarro, H. Prade, B. Gaume. Clustering Sets of Objects Using Concepts-Objects Bipartite Graphs. *In : E. Hüllermeier, S. Link, T. Fober, B. Seeger (eds) Scalable Uncertainty Management. SUM 2012*. Lecture Notes in Computer Science, vol 7520. Springer, Berlin, Heidelberg, pp. 420-432, 2012.

[20] M. del Pilar Angeles, A. Espino-Gamez. Comparison of methods Hamming Distance, Jaro, and Monge-Elkan. *DBKDA 2015, The Seventh International Conference on Advances in Databases, Knowledge, and Data Applications*, pp 63-69, 2015.

[21] Václav Snášel, Zdenek Horák, Ajith Abraham. Understanding Social Networks Using Formal Concept Analysis. *WI-IAT '08 Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 03, pp. 390-393, December 09-12, 2008.

[22] V. Venugopal, W. Baets. Neural Networks and Statistical Techniques in Marketing Research : A Conceptual Comparison. *Marketing Intelligence & Planning*, Vol.12, pp. 30-38, 1994.

[23] R. Wille. Restructuring lattice theory : an approach based on hierarchies of concepts. *I. Rival, (ed.) Ordered Sets. Reidel, Dordrecht-Boston*, pp. 445-470, 1982.

[24] W. E. Winkler. The state of record linkage and current research problems. *Statistical Research Division, U.S. Bureau of the Census*, 1999.

[25] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, Vol. 1, pp. 3-28, 1978.