Ministère de l'enseignement supérieur et de la recherche scientifique

UNIVERSITE SAAD DAHLAB DE BLIDA 1

Faculté des Sciences Département d'informatique

Master Ingénierie Logiciel

Le machine Learning pour l'Evaluation Automatique des Réponses Courtes : Application à la Langue Arabe.

Présenté par :

- Hadjersi Mohamed
- Benguergoura Oussama

Proposé et Encadré par:

- Mme. Leila Ouahrani

Composition de jury :

- M. FERFERA SOFIANE Président
- M. BENAISSI SALAMI Examinateur

Promotion

2019/2020

Abstract

Automatic grading has become an imperative in the educational field, thanks to the privileges it offers over the manual method, which costs time, effort and precision. Automatic grading offers some privileges such as online tests, instant scoring and eliminates the need of a supervisor teacher. The objective of this work is to develop an automatic evaluation system for short answers, written in natural language, to open questions. The model of the system is based on supervised machine learning techniques. The conception of this system highlights an engineering of extracting the essential characteristics used in the scoring process by combining information related to a specific domain such as Semantic space with that related to the general domain of learning such as word embeddings, the similarity mesures are also used as characteristics. Even though our work is initially destined for the Arabic language, we conducted our experiments on the two languages Arabic and English by training the models offered on two datasets, one in Arabic and the other one in English. The results confirm the impact of the use of machine learning and specific supervised approaches on improving accuracy compared to unsupervised models based on computational statistics developed in a previous work. The results obtained with the English dataset are comparable to certain works in the literature and exceed others and confirm the genericity of our approach and its independence from the language.

Keywords: Automatic short answer grading, Word Embedding, Semantic space, Similarity measures, Machine learning.

ملخص

أصبح التنقيط التلقائي امرا ضروريًا في مجال التعليم بفضل الامتيازات التي يوفرها مقارنة بالطريقة اليدوية التي تعد مكلفة للغاية من حيث الوقت والجهد وحتى الدقة. يمنحنا التنقيط التلقائي العديد من المزايا مثل الاختبارات عبر الإنترنت ، التنقيط الفوري ، ويريح المعلم من المهمة الشاقة المتمثلة في التنقيط اليدوي. الهدف من عملنا هو تطوير نظام تقييم آلي للإجابات القصيرة ، المكتوبة بلغة طبيعية ، الخاصة بالأسئلة المفتوحة. يعتمد نموذج النظام على تقنيات التعلم الآلي الخاضعة للإشراف. يسلط تصميم هذا النظام الضوء على هندسة استخراج الخصائص الأساسية المستخدمة في عملية التنقيط من خلال الجمع بين المعلومات المتعلقة بالمجال المحدد بالاعتماد على الفضاء الدلالي مع تلك المتعلقة بالمجال العام للتعلم بالاعتماد على تضمين الكلمات ، ناهيك عن استخدام مقاييس التشابه النصي كخصائص . على الرغم من أن عملنا موجه في البداية للغة العربية ، فقد أجرينا تجاربنا على اللغتين العربية والإنجليزية من خلال تدريب النماذج المقترحة على مجموعتي بيانات ، واحدة باللغة العربية والأخرى باللغة الإنجليزية. تؤكد النتائج التي تم الحصول عليها تأثير استخدام التعلم الآلي والنهج الخاضع للإشراف بشكل خاص على تحسين الدقة مقارنة بالنماذج غير الخاضعة للإشراف بناءً على الإحصائيات الحسابية التي تم تطويرها في عمل سابق. النتائج التي تم الحصول عليها من خلال مجموعة البيانات الإنجليزية قابلة للمقارنة مع بعض الأعمال في الأدب وتتفوق على غيرها وتؤكد شمولية نهجنا واستقلاله عن اللغة.

الكلمات المفتاحية: التقييم الالي للإجابات القصيرة ، تضمين الكلمات ، الفضاء الدلالي ، مقاييس التشابه ، التعلم الآلي.

Résumé

La notation automatique est devenue un impératif dans le domaine de l'éducation grâce aux privilèges qu'elle offre devant la méthode manuelle très coûteuse en temps, en effort et même en précision. La notation automatique nous offre de nombreux avantages tels que les tests en ligne, la notation instantanée et décharge l'enseignant de la tâche fastidieuse de notation manuelle. L'objectif de notre travail est de développer un système d'évaluation automatique des réponses courtes, écrites en langage naturel, aux questions ouvertes. Le modèle du système est basé sur des techniques supervisées d'apprentissage automatique (Machine Learning). La conception de ce système met en évidence une ingénierie d'extraction des caractéristiques essentielles utilisées dans le processus de notation en combinant les informations liées au domaine spécifique en se basant sur l'espace sémantique à celles liées au domaine général en se basant sur l'incorporation des mots, et l'utilisation des similarités textuels comme des caractéristiques. Bien que notre travail soit destiné initialement, pour la langue arabe, nous avons conduit nos expérimentations sur les deux langues l'arabe et l'anglais en entraînant les modèles proposés sur deux Datasets l'un en arabe et l'autre en anglais. Les résultats obtenus confirment bien l'impact de l'utilisation de l'apprentissage automatique et particulièrement les approches supervisées sur l'amélioration de la précision par rapport aux modèles non supervisés basées sur les statistiques computationnelles développées dans un travail précédent. Les résultats obtenus avec le Dataset anglais sont comparables à certains travaux de la littérature et dépassent d'autres et confirment bien la généricité de notre approche et son indépendance à la langue.

Mots clés : Evaluation automatique des réponses courtes, Incorporation des mots, espace sémantique, Mesures de similarité, Apprentissage Automatique.

Remerciements

Nous commençons notre parole avec celui qui a commencé notre création, Dieu qui nous a conduits et nous a facilité à atteindre ce point, qui nous a aidés avec patience, force et discernement, merci Dieu.

Ceux qui remercient Dieu, remercient les personnes, les personnes qui ont contribué à notre succès, que ce soit financièrement ou moralement. Nous présentons des remerciements et une appréciation à notre professeur superviseur **Mme Ouahrani Leila** pour tous ses efforts, encadrements, ses conseils, et sa coopération, merci Madame.

Nous remercions également les membres du jury pour leur présence, pour leur lecture attentive de notre mémoire ainsi que pour les remarques qu'ils nous adresseront lors de cette soutenance afin d'améliorer notre travail.

Chaque succès à un ingénieur, et l'architecte de notre succès est attribué à nos parents, qui ont été un soutien pour nous et notre esprit absolu. Merci et que Dieu vous garde pour nous.

Nos remerciements s'adressent pour tous les frères et sœurs qui étaient avec nous tout au long de ce parcours et qui nous ont soutenu dans les moments de faiblesse et d'effondrement, Nous adressons nos remerciements à tous les membres de la famille **Hadjersi** et **Benguergoura** les frères, les sœurs et les cousins. Nous remercions également tous les collègues que nous avons eu à l'université et partagés avec eux les plus beaux jours, un merci spécial à **Ouzeri Roufaida**, **Rahma Amira, Boumenir Hamza** et le groupe **UGK14**, pour ses supports.

Liste des Tables

Liste des Figures

Glossaire

Abréviation	Signification
ASAG	Notation automatique des réponses courtes (Automatic Short Answer Grading)
RM	Réponse Modèle
RE	Réponse Etudiant
Q	Question
WE	Incorporation des mots (Word Embedding)
ML	Apprentissage automatique (Machine learning)
ES	Espace Sémantique
CBOW	Sac continu de mots (Continuous Bag Of Words)
RMSE	Erreur quadratique moyenne (Root Mean Squared Error)
CP	Calcul de Pearson
SVM	Machine à vecteur de support (Support Vector Machine)
RNN	Les réseaux de neurones récurrents (Recurrent Neural Networks)
LR	Régression Linéaire (Linear Regression)
LogR	Régression logistique (Logistic Regression)
RF	Forêt aléatoire (Random Forest)
DT	Arbre de décision (Decision tree)
LSTM	Long Short-Term Memory (longue mémoire à court terme)
KNN	K-plus proche voisin(k-nearest neighbors)
ANN	Réseau neurone artificiel(Artificial neural network)
SVR	Prise en charge de la régression vectorielle(support vector regression)
RNN2	les réseaux de neurones récursifs (recursive neural networks)

DBN	Réseau de croyances profondes(Deep Belief Network)
NB	Naïve Bayes.
FCNN	Réseau de neurones entièrement connecté (FulyConectedNN)
TE	Textual entailmet (Implication de texte)
MM	Modèle mathématique
TM	Note de l'enseignant (teacher mark)

Table des matières

Chapitre 1 : Introduction générale

1. Introduction :

Vu le développement de la technologie ces dernières années, plusieurs domaines ont adopté l'automatisation, parmi eux le domaine éducatif qui a bénéficié de cette technologie, car la méthode manuelle de correction prend un temps considérable, et c'est une opération couteuse en même temps. Sachant que la notation des réponses en texte libre de l'élève est souvent longue et difficile, donc plusieurs techniques pour la notation automatique des réponses courtes ont été réalisées par l'apport de l'efficacité dans le domaine informatique.

Nous nous intéressons dans cette étude à un système basé sur l'apprentissage automatique (MACHINE LEARNING) pour prédire les notes des étudiants sur des questions ouvertes à réponses courtes (quelques mots à quelques phrases construites en **langage naturel**)

Notre travail est orienté vers la **langue arabe** qui, bien que largement utilisée aujourd'hui, les travaux faits en langue arabe sont très limités, ils n'ont pas une maturité des résultats dans le domaine de l'évaluation automatique.

Notre système final sera testé avec la langue anglaise afin d'obtenir un système complet, fiable et standard.

2. Problématique :

L'idée de base de l'évaluation automatique consiste à comparer une réponse d'un ou de plusieurs étudiant avec une réponse **modèle** (de référence) et d'accorder une note ou bien un score. Cette procédure est souvent difficile vu la variété linguistiques (une réponse donnée pourrait être articulée de différentes façons), nature subjective de l'évaluation (multiples réponses possibles).

La situation actuelle des outils d'évaluation automatique des réponses courtes ne semble pas avoir atteint un haut degré de maturité et impose aux examinés (étudiants ou apprenants) des compétences et des contraintes importantes.

A l'Ere du Machine Learning nous voulons explorer ces modèles pour l'évaluation automatique en langue arabe et analyser l'impact sur les outils déjà développés dans le contexte de travaux précédents.

3. Objectifs :

L'objectif de notre travail est de concevoir une approche basée sur l'apprentissage automatique pour l'évaluation automatique des réponses courtes en Langue Arabe.

La réalisation de cet objectif passe par plusieurs sous-objectifs nous citons :

- Exploration des différentes approches de machine Learning dans l'évaluation automatique.
- Elaboration des caractéristiques **(features)** à considérer pour pouvoir réaliser un apprentissage automatique en considérant les aspects liés à l'évaluation automatique.
- Concevoir le modèle d'apprentissage **(Régression, Classification)** pour la prise en compte des caractéristiques d'évaluation automatique.
- Entrainement et test du modèle **(Acquisition du Dataset et métriques d'évaluation).**
- Evaluer la nouvelle approche par rapport aux approches déjà développées dans des travaux précédents rentrant dans le même contexte.
- Mener une étude parallèle pour l'anglais afin de comparer nos résultats à ceux de la littérature et vérifier leur indépendance à la langue.

4. Implications du travail:

Ce travail vise à atteindre certains objectifs ayant un impact dans la société qui sont :

- Faciliter la tâche d'évaluation pour les enseignants et à réduire le temps de correction.
- L'encouragement de l'intégration des systèmes de tests spécifiques liés aux réponses ouvertes dans les plateformes Elearning.
- L'Adaptation avec la culture des examens en ligne, ce qui est un nouvel avantage dans le domaine de l'éducation.

5. Organisation du mémoire :

La structure de notre travail est composée de 4 chapitres principaux :

- **Chapitre 1 :** Introduction générale qui définit le contexte de notre étude, cite la problématique posée et les objectifs souhaités par cette étude.
- **Chapitre 2 :** Etat de l'art qui englobe les concepts fondamentaux liés à la notation automatique, les concepts fondamentaux du machine learning ainsi qu'une synthèse des travaux déjà faits dans ce contexte.
- **Chapitre3 :** Conception du système de l'Evaluation Automatique des Réponses Courtes et la méthodologie suivie pour la mise en œuvre du système.
- **Chapitre 4 :** Présente les différentes techniques utilisées pour la phase d'implémentation et les résultats obtenus.

Nous terminons notre étude par une conclusion générale et des perspectives

Chapitre 2 : Etat de l’art

Le domaine de la notation automatique comprend de nombreux concepts qui sont considérés comme un pilier pour la construction d'un système performant. Dans cette partie nous abordons les concepts généraux de la notation automatique et la manipulation des textes. Nous présentons également les concepts fondamentaux dans l'apprentissage automatique ainsi que les concepts liés à la similarité des textes courts qui est la base du processus d'évaluation.

Dans la section 4, nous présentons une revue de la littérature de travaux les plus récents utilisant le Machine Learning. Cette analyse nous permet de dresser une synthèse sur la base de critères que nous avons considérés suite à nos lectures. Nous profitons pour présenter succinctement les travaux déjà réalisés dans le cadre du même projet. Ces travaux nous ont permis d'extraire les propriétés nécessaires à l'entrainement de nos modèles.

1. Concepts fondamentaux liés à l'évaluation automatique:

Dans ce qui suit, nous examinons le domaine de l'évaluation automatique qui est la base de cette étude et du système que nous proposons, en présentant sa définition générale, nous parlons sur son historique, son origine et les systèmes qui ont été développés dans ce domaine ainsi que leurs classifications.

1.1. Un aperçu sur la notation automatique :

La notation automatique est une technique d'évaluation des réponses en langage naturel aux questions objectives, qui note de manière autonome les réponses données par l'étudiant. Les questions qui se base sur une réponse modèle, s'appelle des questions fermées et les questions de compréhension, appelées questions semi-ouvertes (Zhang et al. 2019). La recherche dans le domaine de classement automatiques des réponses et la notation a vu sa naissance les années 1966, le premier travail dans ce contexte était celle de Ellis Batten « The imminence of grading essays by computers » (Ellis Batten 1966). Les techniques de la notation automatiques sont diversifiées en fonction du type de question, comme les réponses courtes et les essais, cela a mené que la notation automatique des réponses en langage naturel devienne un vaste domaine. Cette étude est consacrée pour le type de notation automatique des réponses courtes. La question de ce type doit avoir une réponse qui repose sur les connaissances externes au lieu d'une réponse reconnue à l'intérieur de la question. Les réponses données doivent être formées en quelques mots à quelques phrases construites en langage naturel. L'évaluation de ces réponses doit se reposer sur le contenu plutôt que sur le style d'écriture. (Steven Burrows1 et al. ,2015)

1.2. Les approches des systèmes de notation automatiques :

Selon la revue de littérature de (Steven Burrows1 et al.,2015) plusieurs approches sont utilisées pour le développements des systèmes de notation automatique. Ces approches se basent sur le mappage de concepts, L'extraction d'informations, le corpus, et le Machine Learning(ML). Nous présentons dans la suite de manière succincte les premières approches. Nous insistons plus sur l'approche lié à notre travail à savoir le ML. La figure **« Figure 1 »** présente les quatre approches des systèmes de notation automatiques :

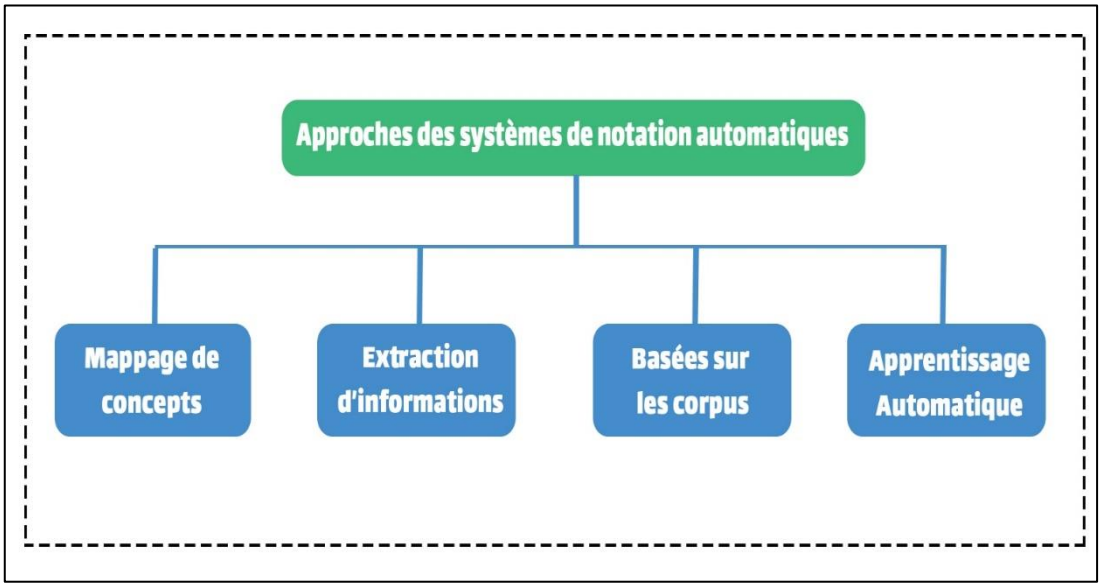


Figure 1: Les approches des systèmes de notation automatiques

a) Systèmes basés sur le mappage de concepts :

La cartographie conceptuelle est basée sur la présence des concepts spécifique dans les réponses des étudiants pour une question donnée. Le système (Burstein et al.,1996) est pour le but d'être capable d'analyser de manière fiable le contenu des réponses, les créateurs ont construits un lexique basé sur un concept spécifique au domaine et une grammaire conceptuelle pour représenter l'ensemble de réponses puis les réponses sont automatiquement notées en se voyant attribuer des classifications appropriées, un autre système c'est Concept Rater **(c-rater)** (Leacock et al., 2003) été développé pour mesurer la compréhension par un étudiant d'un contenu spécifique sans tenir compte de la compétences en écriture de l'élève, il test si la réponse de l'élève contient des informations linguistiques spécifiques requises comme preuve que le concept a été appris, L'appariement est basé sur un ensemble de règles et une représentation canonique des textes en utilisant la variation syntaxique, l'anaphore, la variation morphologique, les synonymes et la correction orthographique.

Le prototype de marqueur de texte automatisé **(ATM)** (Callear et al., 2001) décompose automatiquement une réponse de modèle écrite par un expert, à une question fermée, en la plus petite unité de concepts viable avec leurs dépendances prises en compte en étiquetant automatiquement les concepts résultants et leurs dépendances avec des nombres. Le même processus est appliqué à la réponse de chaque élève et les concepts résultants et leurs dépendances sont ensuite mis en correspondance avec ceux de la réponse de l'examinateur de modèle.

b) Systèmes basés sur l'Extraction d'informations (information extraction) :

L'extraction d'informations (Cowie et al., 2000) est tout processus qui structure et combine de manière sélective des données trouvées dans un document ou plusieurs, Dans notre étude l'extraction d'informations (Steven Burrows1 et al.,2015) concerne la recherche de faits dans les réponses des élèves.

Étant donné que les réponses courtes sont généralement censées inclure des idées spécifiques, **AutoMark** (Mitchell et al., 2002) un système qui utilise les techniques d'extraction d'informations pour fournir un marquage informatisé des réponses courtes en texte libre, **AutoMark** recherche un contenu spécifique dans les réponses en texte libre, le contenu étant spécifié sous la forme d'un certain nombre de modèles de schéma de marque, puis faire l'appariement avec les modèles en extrayant des information depuis la réponse.

WebLAS (Web-based Language Assessment System) (Bachman et al., 2002) il essai de décomposer la réponse modèle a des parties de texte et trouver des éléments important qui nous aide pour la notation, l'instructeur doit confirmer les résultats, après la confirmation le système va chercher les alternatives possibles de ces éléments et de leurs mots individuels et doivent être aussi confirmées par l'instructeur, puis il va générer une expression régulière qui sera chargée d'évaluer la réponse.

c) Systèmes basés sur les corpus (Corpus-based) :

Les méthodes basées sur les corpus exploitent les propriétés statistiques des grands corpus de documents (Steven Burrows1 et al.,2015), ces méthodes facilitent l'interprétation de texte et aident pour le calcul de similarité entre la réponse modèle et la réponse d'étudiant.

Pour l'étude de (Mohler et al., ,2009) ont fait un certain nombre de mesures de similitude de texte basées sur les connaissances et sur le corpus, pour le corpus les méthodes utilisées sont Explicit Semantic Analysis**(ESA)** (Gabrilovich et al., 2006) and LSA (Landauer et al., 1997).

d) Systèmes basés sur d'Apprentissage Automatique (Machine Learning) :

Le problème de notation automatique peut être considéré comme un problème de classification ou de prédiction. Le processus d'application des algorithmes d'apprentissage automatique pour la notation peut être divisé en deux étapes principales: l'ingénierie des fonctionnalités et le choix de modèle de classification. D'après un ensemble de données étiqueté des réponses d'étudiant, nous essayons d'extraire des bonnes caractéristiques pour les utiliser dans notre modèle de classification.

Nous avons le système (Madnani et al., 2013) qui utilise huit caractéristiques de texte comme fonctionnalités pour l'algorithme de régression logistique dans le but de faire la classification des réponses de compréhension des étudiants.(Hou et al., 2010) est un système qui utilise Tags POS(Part Of Speech), fréquence des termes, tf-idf et entropie comme fonctionnalités pour le classificateur SVM, qui est utilisé comme un indicateur pour aider les enseignants de suivre le progrès des étudiants en classifiant les réponses dans des classes d'échelles de 0 à 10. E-Examiner (Gütl, 2007) prédit des scores en utilisant les métriques ROUGE (Lin, 2004) comme fonctionnalités d'apprentissage automatique.

Nous allons reprendre une synthèse plus approfondie de travaux ayant utilisé le ML dans la section 4.

Les systèmes de notation automatiques reposent fortement sur la mesure de la similitude entre la réponse d'étudiant et la réponse modèle. Il existe de nombreuses techniques pour ce faire, ces dernières se diffèrent d'une à une autre selon le type de similitude à calculer (l'analyse de données textuelles, la recherche documentaire ou l'extraction de connaissances à partir de données textuelles).

Notre système se base sur le concept de similarité entre la réponse modèle et la réponse étudiant où nous expliquons les différentes techniques de similarité des textes courts dans ce qui suit.

2. Concepts fondamentaux liés à la similarité des textes courts :

Les mesures de similarités varient en plusieurs sections différentes, chacune ayant un ensemble de techniques qui contribuent grandement à l'existence d'un rapprochement entre les mots ou entre les phrases. Il y a celles qui se basent sur une chaîne, sur un corpus ainsi que des similitudes basées sur la connaissance. Dans la partie suivante, nous parlons de chacun de ces types, en donnant à chacun d'eux un ensemble de mesures.

2.1.Similarité des chaines :

Les mesures de similitude des chaînes dépendent sur des mots correspondants, de la séquence des phrases correspondantes ou de la composition des lettres, sans considération de sens du mot. Par conséquent, pour deux mots ayant pratiquement le même sens peuvent être considérés comme des mots différent en raison de leur composition différente, comme «نصب " et "احتال", Or que «احتال » et «احتيال » peuvent être considérées comme très proches.

Les techniques de mesure de similarité des chaines peuvent être basé-terme ou basé-caractère, comme indique la figure **« figure 2 »** :

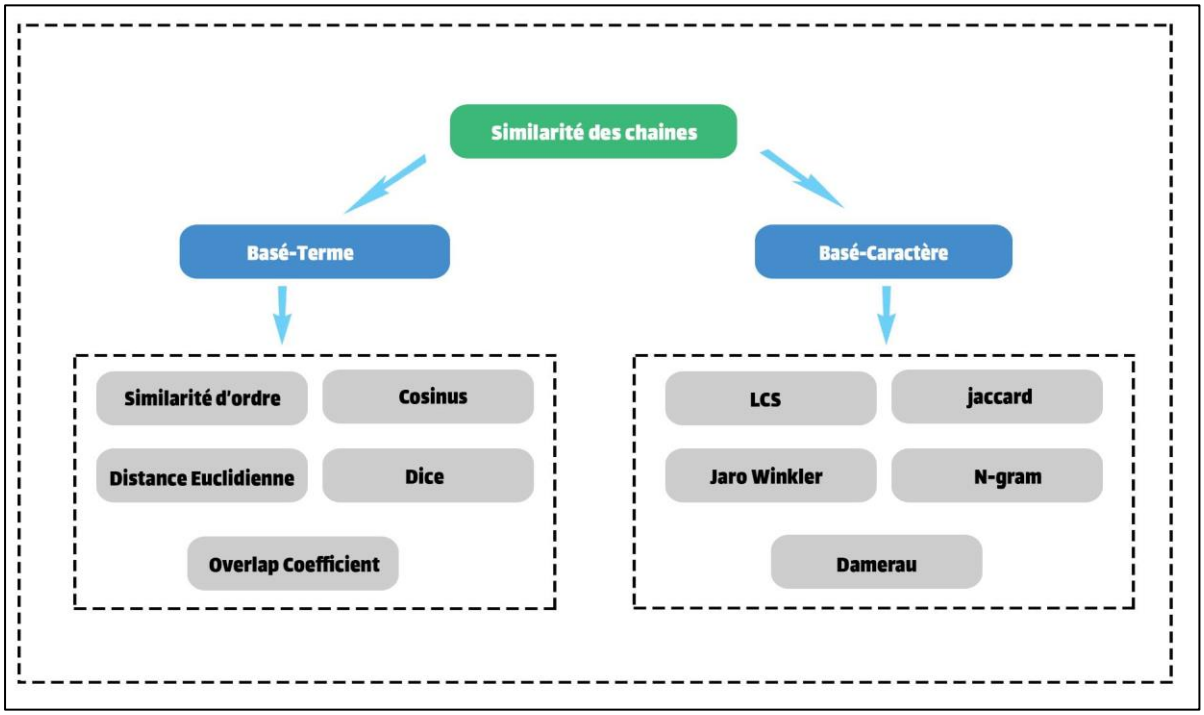


Figure 2: Les méthodes de calcul de similarité des chaines

a) Basée-terme :

- Similarité Cosinus : La similarité cosinus est de calculer le cosinus de l'angle entre les représentations vectorielles des textes à comparer. La similarité obtenue $Sim_{Cosinus}$(d1, d2) ∈ [0, 1].

$$sim_{cosinus}(d_1, d_2) = \frac{\vec{d_1}.\vec{d_2}}{\|\vec{d_1}\|\|\vec{d_2}\|}$$

- Dice's coefficient (Dice, 1945) : L'indice de Dice, mesure la similarité entre deux textes d1 et d2 en se basant sur le nombre de termes communs à d1 et d2.

$$sim_{dice}(d_1, d_2) = \frac{2N_c}{N_1+N_2}$$

D'où Nc est le nombre de termes communs à d1 et d2, et N1 (resp. N2) est le nombre de termes de d1 (resp. d2).

- Similarité d'ordre : Son principe est de calculer la similarité entre deux textes selon les mots communs contenants dans les deux textes en considérant l'ordre d'apparition de ces termes d'un texte par rapport à l'autre.

b) Basée-caractère :

- Similarité de Jaccard (Jaccard, 1901) : Coefficient de Jaccard : est le rapport entre la cardinalité de l'intersection des caractères de deux mot et la cardinalité de l'union des caractères. Il permet d'évaluer la similarité entre les mots. Les mots d1 et d2 sont donc représentés non pas comme des vecteurs, mais comme des ensembles de caractères. La similarité obtenue (d1, d2) ∈ [0, 1].

$$sim_{jaccard}(d_1, d_2) = \frac{\|d_1 \cap d_2\|}{\|d_1 \cup d_2\|}$$

- Distance de **Levenshtein** (Hall et al., 1980)(Peterson, 1980) **:** La distance DLv est calculer selon le nombre minimum d'opérations qui sont appliquer pour la transformation d'une chaine a une autre comme l'insertion, la suppression ou la substitution d'un seul caractère, soit la transposition de deux caractères adjacents

- Distance de Jaro (Jaro et al., 1989) La distance de **Jaro** et une technique basé caractère d'où l'algorithme cherche les caractères correspondants dans les deux chaines en considérant le facteur d'éloignement maximal qui est calculé comme suit :

$$\left\lfloor \frac{\max(|s_1|, |s_2|)}{2} \right\rfloor - 1$$

Donc deux caractères qui sont identiques dans les deux chaines mais la différence de leurs ordres est inférieure au facteur d'éloignement ne vont pas être considéré comme correspondants.

La distance de **Jaro** entre les chaînes **s1** et **s2** est définie par **dj** :

$$d_j = \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right)$$

D'où :

|si| : est la longueur de la chaîne de caractères **si**

m : est le nombre de caractères correspondants

t :est le nombre de transpositions (Le nombre de transpositions est obtenu en comparant le i-ème caractère correspondant de **s1** avec le i-ème caractère correspondant de **s2**. Le nombre de fois où ces caractères sont différents, divisé par deux)

- N-gram similarité (Barrón-Cedeno et al., 2010).

- La plus long sous-chaine commune (Longest common substring (Gusfield, 1997)**)** est un algorithme qui considère la chaine commune la plus longue. La plus longue sous-séquence commune à deux suites, ou deux chaînes de caractère, est une séquence étant sous-suite des deux suites, et étant de taille maximum.

$$sim_{LCS}(P,R) = \frac{len(LCS)}{Max(len(P), len(R))}$$

- Needleman-Wunsch (Needleman, et al.,1970) **:** Il effectue un alignement global pour trouver le meilleur alignement sur l'ensemble des deux séquences. Il est approprié lorsque les deux séquences sont de longueur similaire.

2.2. Approches basées sur un corpus :

Les approches basées sur le corpus diffèrent du reste des critères de similitude qui se rapportent aux similitudes basées sur la chaîne ou les connaissances. Ces approches sont liées aux informations obtenues à partir d'une grande collection de documents (corpus) d'un domaine spécifique. La figure **« Figure 3 »** illustre les différentes techniques basées sur le corpus:

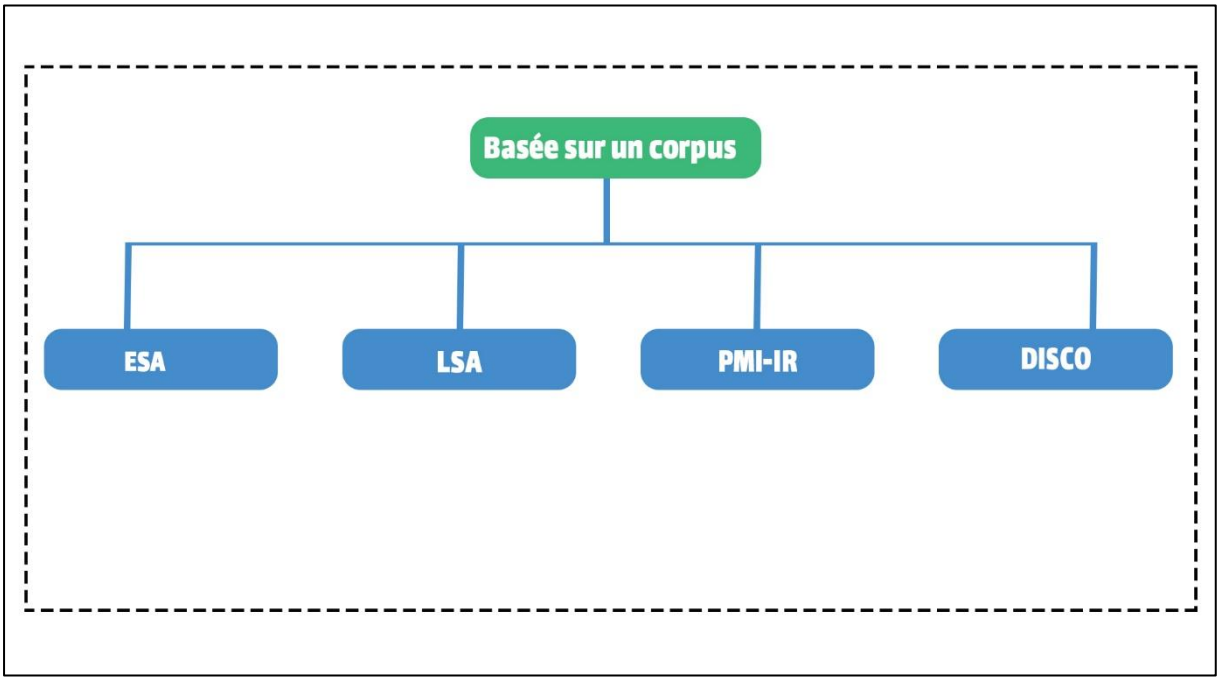


Figure 3: Représentation des différentes techniques basées sur un corpus

Dans ce qui suit, nous expliquons chacune des techniques ci-dessous :

Analyse sémantique latente (LSA) (Landauer et al., 1997) est une méthode mathématique pour la modélisation informatique et la simulation du sens des mots et des passages par l'analyse de corpus représentatifs de texte naturel. **Analyse sémantique explicite (ESA)** (Gabrilovich et al., 2007) est une représentation vectorielle de texte (mots individuels ou documents entiers) qui utilise un corpus de documents comme base de connaissances où un mot est représenté comme un vecteur colonne dans la matrice **tf – idf** du corpus de texte et un document (chaîne de mots) est représenté comme le centre de gravité des vecteurs représentant ses mots.

Information mutuelle point par point - recherche d'informations (PMI-IR) (Turney, 2001) est un algorithme qui utilise les informations mutuelles ponctuelles (PMI) et la récupération d'informations (IR) pour mesurer la similitude des paires de mots. Il est basé sur

des données statistiques acquises en interrogeant un moteur de recherche Web. **Extraction de mots de distribution similaire à l'aide de co-occurrences, Disco** (Kolb, 2008) cette technique calcule la similitude distributionnelle entre les mots tout en considérant que les mots qui ont un sens similaire se produisent dans un contexte similaire. La similitude est basée sur l'analyse statistique de très grandes collections de textes.

Nous parlons aussi sur la méthode **COALS** (Occurrence corrélée analogue à la sémantique lexicale) qui s'avère beaucoup plus évolutifs et faciles à mettre en œuvre dans des situations nécessitant l'utilisation de grands corpus de texte. **COALS** est une technique de normalisation des données numériques transformée à partir des données textuelles (il s'agit d'une matrice de co-occurrence), elle utilise la corrélation à la fois pour la normalisation et pour mesurer la similitude vectorielle (D. L. T. Rohde et al.,2004).

2.3.Approches basées sur les connaissances

La similitude fondée sur les connaissances est une mesure de similitude sémantique qui détermine le degré de similitude entre les mots à l'aide d'informations extraits de ressources sémantiques externes (Gomaa et al., 2014). La similarité sémantique est un concept selon lequel un ensemble de documents ou de termes se voient attribuer une métrique basée sur la ressemblance de leur signification (contenu sémantique) (E. Negre,2013).

La figure **« Figure 4 »** est une illustration de la classification des différentes techniques de similarité basée sur les connaissances :

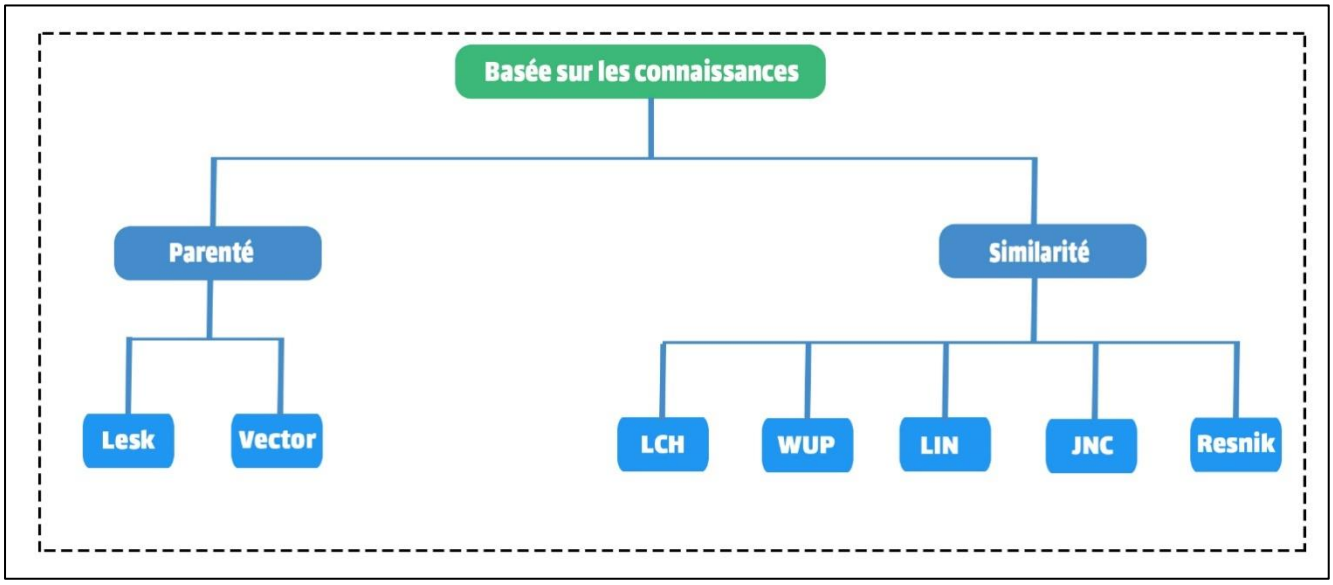


Figure 4: Représentation des différentes techniques basées sur les connaissances

Word Net (Miller, 1995) & (Mihalcea et al., 2006) sont les ressources sémantiques les plus populaire dans le domaine de la mesure de la similitude basée sur la connaissance entre les mots. Les techniques existantes sont classées en techniques basé sur :

- **La similarité** comme : **LCH** (Leacock et al. , 1998) c'est la longueur du chemin le plus court entre deux concepts utilisant le comptage de nœuds sur la profondeur maximale de la taxonomie. **WUP** (Wu et al., 1994) mesure la profondeur de deux concepts donnés dans la taxonomie Word-Net et la profondeur du sous-consommateur le moins commun (LCS), et combine ces chiffres en un score de similitude. **LIN** (Lin 1998), **JNC** (Jiang et al., 1997), **Resnik** (Resnik 1995) cette technique renvoie le contenu informationnel (IC) du LCS de deux concepts en calculant la probabilité de rencontrer une instance du concept c dans un grand corpus.
- **Les relations (parenté)** comme **Lesk**(Lesk ,1986)**, Vector** (Patwardhan, 2003), où la similitude de **Lesk** de deux concepts est définie en fonction du chevauchement entre les définitions correspondantes, en se basant sur un dictionnaire.

Nous abordons dans la section suivante l'aspect de l'apprentissage supervisé afin de l'exploité dans la création de notre système

3. Concepts fondamentaux liés à l'apprentissage automatique :

Étant donné que l'apprentissage automatique pour l'évaluation des étudiants est l'un des piliers de cette étude, nous fournissons une présentation dans lequel nous expliquons les concepts fondamentaux, les métriques utilisées pour mesurer la performance et l'efficacité de l'algorithme choisi et les différentes caractéristiques du machine Learning.

3.1. Définitions :

Le domaine de l'apprentissage automatique est parmi les domaines les plus connus dans le domaine de l'intelligence artificielle.

En 1959, Arthur Samuel a défini l'apprentissage automatique comme un « domaine d'étude qui donne aux ordinateurs la capacité d'apprendre sans être explicitement programmés. »(Swamynathan, 2017). Or que les chercheurs Borovcnik, Bentz, et Kapadia ont définies l'apprentissage automatique comme un ensemble de méthodes capables de détecter automatiquement des modèles dans les données, puis d'utiliser les modèles découverts pour

prédire les données futures ou pour effectuer d'autres types de prise de décision dans l'incertitude. (Borovcnik, al. 1991)

3.2.Types de l'apprentissage automatique :

Dans le domaine de l'apprentissage automatique, il existe deux principaux types de tâches: supervisées et non supervisées.

La figure **« Figure 5 »** présente les deux différents types de l'apprentissage automatique les algorithmes liés à chaque type.

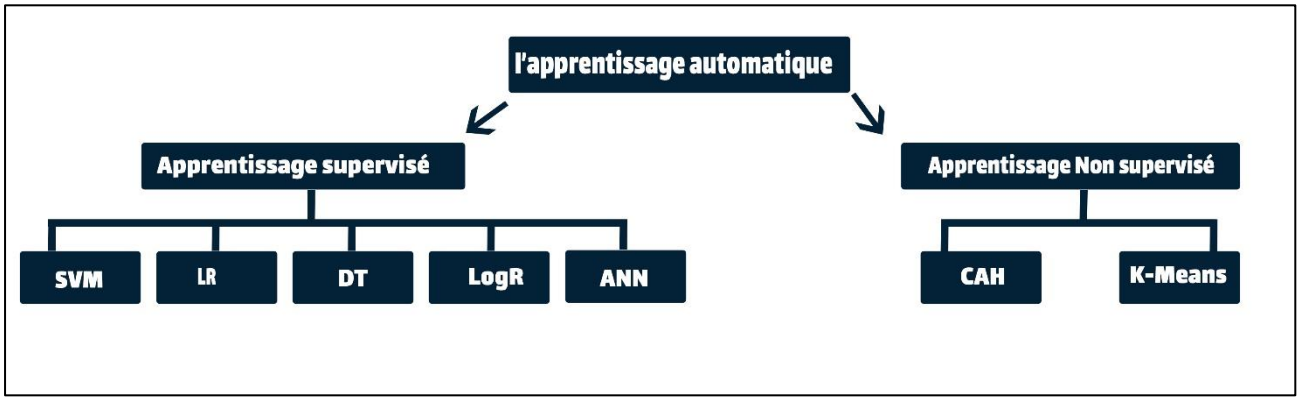


Figure 5: les types d'apprentissage automatique avec leurs algorithmes

- **Apprentissage automatique non supervisé :**

L'apprentissage non supervisé fait référence au processus de regroupement des données en grappes à l'aide de méthodes automatisées ou d'algorithmes sur des données qui n'ont pas été classées ou catégorisées. Dans cette situation, les algorithmes doivent « apprendre » les relations ou caractéristiques sous-jacentes à partir des données disponibles et regrouper les observations avec des caractéristiques similaires. (Berry et al., 2019)

Les problèmes d'apprentissage non supervisés peuvent être regroupés en problèmes de **regroupement** et **d'association.** Le **Regroupement (Clustering)** consiste à regrouper des points de données par similarité ou par distance, sachant que **l'Association** nous permet de découvrir des règles qui décrivent de grandes parties de nos données.

Il existe de nombreux algorithmes liée à l'apprentissage automatique non supervisé, parmi ces algorithmes nous avons :

- **K-moyen(k-means):**

L'algorithme K-Moyen est l'un des algorithmes d'apprentissage non supervisé les plus simples qui résolvent le problème de regroupement. Il est utilisé lorsque les données étiquetées ne sont pas disponibles.

- **Clustering ascendant hiérarchique (CAH) :**

selon (Chaitanya,2018) l'algorithme non supervisé CAH est une technique du Regroupement(clustering).

On peut le résumer dans cette expression "*Chaque point ou cluster est progressivement "absorbé" par le cluster le plus proche* ".(Atif, 2016)

Maintenant nous allons présenter la partie la plus intéressante dans notre projet car notre contexte de travail est basé sur l'apprentissage automatique supervisé.

- **Apprentissage Automatique supervisé :**

L'apprentissage supervisé se concentre sur les modèles d'apprentissage en reliant la relation entre les variables et les résultats connus et en travaillant avec des ensembles de données étiquetés. L'apprentissage supervisé fonctionne en alimentant les données d'échantillon de la machine avec diverses fonctionnalités (représentées par «X ») et la sortie de valeur correcte des données (représentée par « y ») (Theobald 2017)

La figure suivante **« Figure 6 »** nous montre la phase d'apprentissage **:**

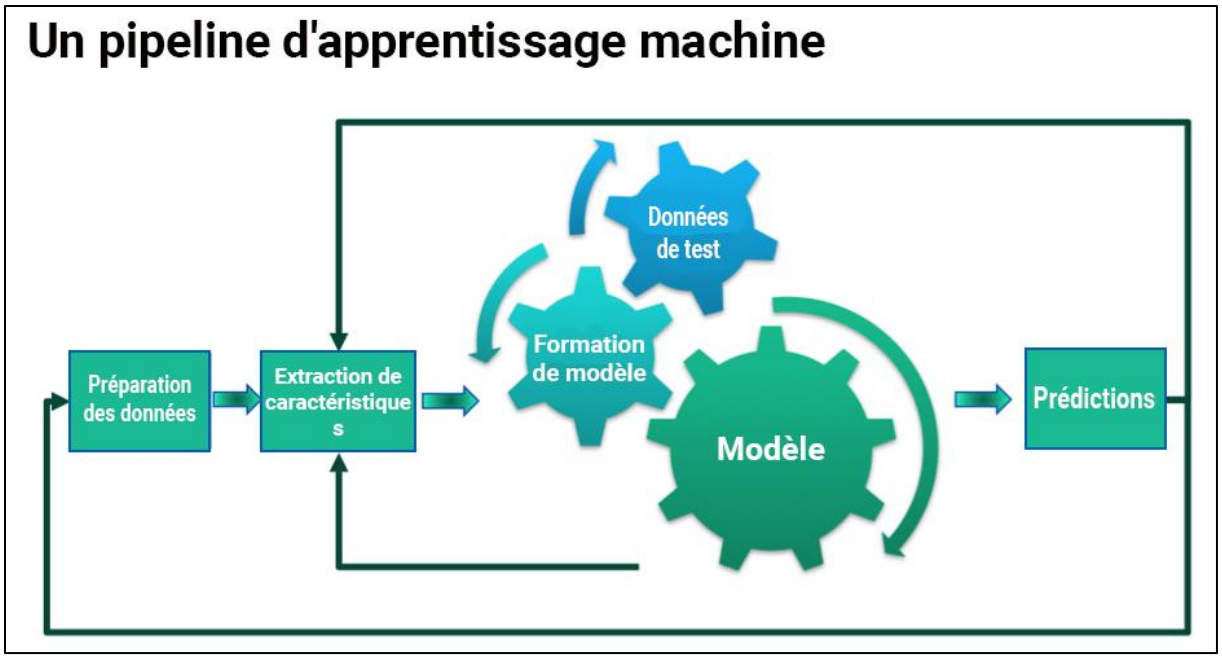


Figure 6: Phase d'apprentissage du modèle d'apprentissage supervisé

Pour l'apprentissage automatique supervisé nous distinguons deux types de modèles, les modèles de régression et les modèles de classification, comme il est illustré dans la figure **« Figure 7 »**

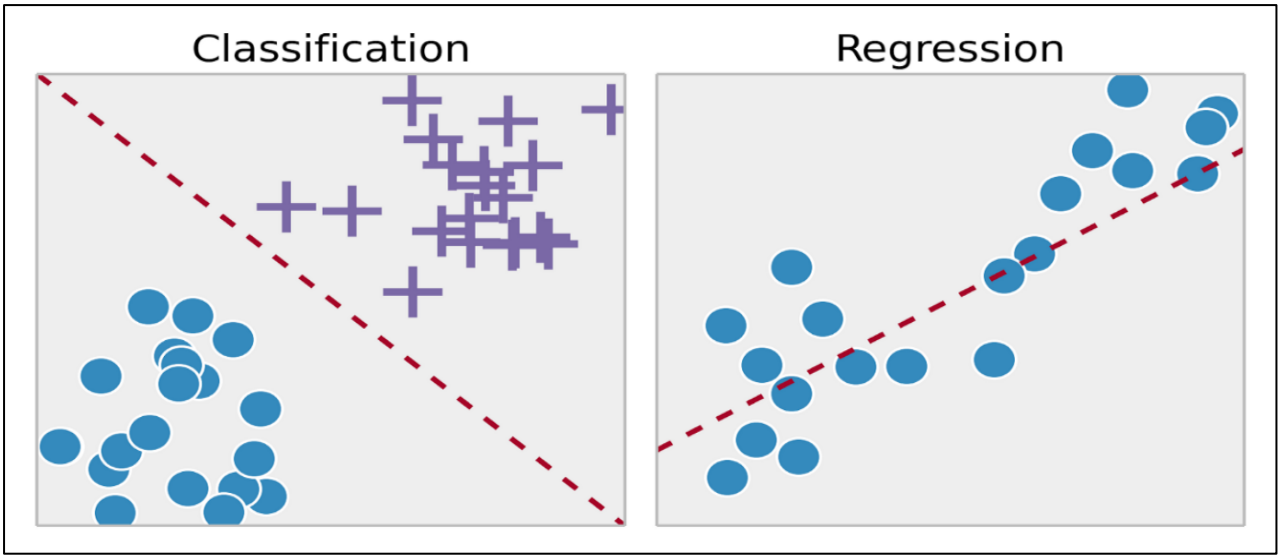


Figure 7: Les différents problèmes de l'apprentissage automatique supervisé

a) Classification: Quand la variable à prédire prend une valeur discrète, on parle d'un problème de classification. Un algorithme de classification peut prédire une valeur continue, mais la valeur continue se présente sous la forme d'une probabilité pour une étiquette de classe.

b) Régression: Un problème de régression survient lorsque la variable de sortie est une valeur réelle. Un algorithme de régression peut prédire une valeur discrète, mais la valeur discrète sous la forme d'une quantité entière. Plusieurs algorithmes d'apprentissage automatique supervisé sont utilisés pour résoudre des problèmes de classification ou de régression parmi ces algorithmes nous avons:

- **Arbre de décision (Decision tree) :**(Avinash Navlani, 2018),l'arbre de décision est un type d'algorithme de traitement automatique supervisé ,c'est une structure arborescente semblable à un organigramme où un nœud interne représente une caractéristique (ou un attribut), la branche représente une règle de décision et chaque nœud feuille représente le résultat.

- **Forêt aléatoire (Random forest) :**(Tony Yiu, 2019)**,** La forêt aléatoire est un algorithme de classification composé de nombreux arbres de décisions qui fonctionnent comme un ensemble. Il utilise l'ensachage et le caractère aléatoire lors de la construction de chaque arbre individuel pour essayer de créer une forêt d'arbres non corrélée dont la prédiction par le comité est plus précise que celle de n'importe quel arbre individuel.

- **Linéaire régression (Linear regression) :** La régression est principalement utilisée pour découvrir la relation entre les variables et les prévisions, ou bien c'est une équation permettant d'estimer une valeur attendue, au moyen de valeurs d'autres variables (x).

- **Machine à vecteurs de support (Support Vector Machine) :** C'est un classificateur discriminant défini formellement par un hyperplan de séparation. Autrement dit c'est un algorithme d'apprentissage automatique supervisé qui peut être utilisé pour des défis de classification. Dans cet algorithme, nous traçons chaque élément de données comme un point dans un espace à n dimensions (où n est le nombre d'entités qu'on a), la valeur de chaque entité étant la valeur d'une coordonnée particulière. Ensuite, nous effectuons la classification en trouvant l'hyperplan qui différencie très bien les 2 classes.

- **Régression logistique (Logistic Regression):**Selon (Ayush Pant, 2019),la régression logistique est un algorithme d'apprentissage automatique supervisé qui est utilisé pour les problèmes de classification, c'est un algorithme d'analyse prédictive et basé sur le concept de probabilité. Il est utilisé pour l'attribution des observations à un ensemble discret de classe.

- **Réseaux de neurones artificiels (Artificial Neural Networks-ANN):** Les réseaux de neurones artificiels ou ANN sont un paradigme de traitement de l'information qui s'inspire de la façon dont le système nerveux biologique, comme le cerveau, traite l'information. Il est composé d'un grand nombre d'éléments de traitement (neurones) hautement interconnectés travaillant à l'unisson pour résoudre un problème spécifique. (Nagesh Singh Chauhan, 2019).

  La figure suivante **« Figure 8 »** représente un modèle général d'un réseau de neurone artificiel qui est inspiré d'un neurone biologique.

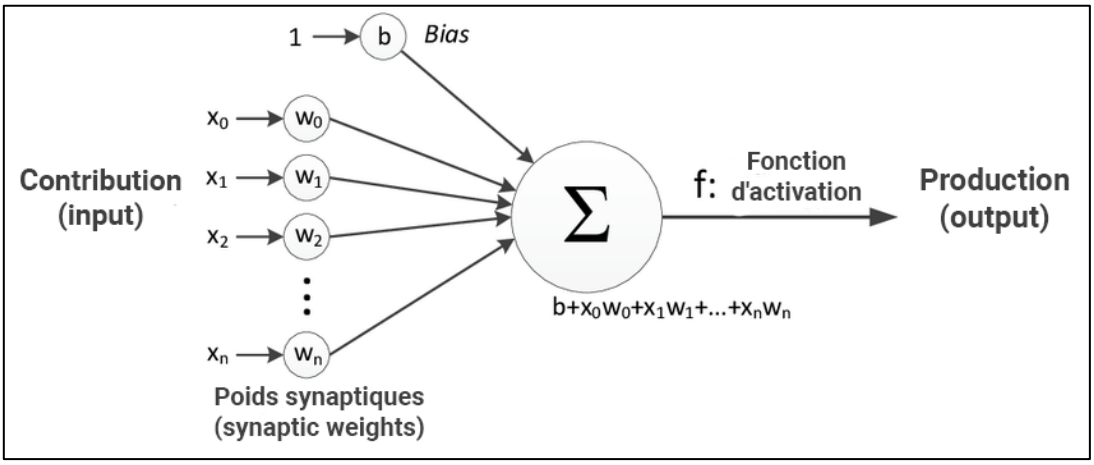


Figure 8: Modèle général d'un réseau de neurone artificiel

3.3.Métriques d'évaluation des modèles :

Le choix et la construction d'un modèle d'apprentissage automatique (Machine Learning) doit passer par une étape d'évaluation pour mesurer sa performance et la validité des résultats. Nous pouvons distinguer qu'il y en' a des mesures utiliser pour l'évaluation des modèles d'apprentissage automatiques pour les problèmes de régression et d'autres pour les problèmes de classification:

a) Métriques d'évaluation de modèles pour les problèmes de régression :

- **Erreur absolue moyenne (Mean Absolute Error)** :(Willmott, et al . 2005) : L'erreur absolue moyenne est la moyenne de la différence entre les valeurs d'origine et les valeurs prédites. Il nous donne la mesure de la distance entre les prévisions et la sortie réelle. Cependant, ils ne nous donnent aucune idée de la direction de l'erreur, c'est-à-dire si nous sommes sous-prédits ou sur-prédits. Mathématiquement, il est représenté comme:

$$MeanAbsoluteError = \frac{1}{N}\sum_{j=1}^{N}|y_j - \hat{y}_j|$$

- **Erreur quadratique moyenne (Mean Squared Error)** : L'erreur quadratique moyenne (MSE) est assez similaire à l'erreur absolue moyenne, la seule différence étant que MSE prend la moyenne du carré de la différence entre les valeurs d'origine et les valeurs prédites. L'avantage de MSE est qu'il est plus facile de calculer le gradient, tandis que l'erreur absolue moyenne nécessite des outils de programmation linéaire compliqués pour calculer le gradient. Comme nous prenons le carré de l'erreur, l'effet des erreurs plus importantes devient plus prononcé que l'erreur plus petite, donc le modèle peut désormais se concentrer davantage sur les erreurs plus importantes. **(Mishra, 2018)**

$$MeanSquaredError = \frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2$$

- **Erreur quadratique moyenne racine (Root Mean Squared Error)** :Selon (Neill el al., 2018) L'erreur quadratique moyenne racine (Root Mean Squared Error) est la racine carrée de la moyenne du carré de toutes les erreurs. L'utilisation de RMSE est très courante et elle est considérée comme une excellente métrique d'erreur à usage général pour les prédictions numériques. C'est une bonne mesure de la précision pour la comparaison entre les erreurs de prédiction de différents modèles.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(S_i - O_i)^2}$$

Où **Oi** sont les observations, **Si** les valeurs prédites d'une variable et **n** le nombre d'observations disponibles pour l'analyse.

- **Le coefficient de corrélation de Pearson :** (karl Pearson, 1895)(Wu et al.,2019) a défini le coefficient de corrélation comme suit : Le coefficient de corrélation de Pearson est généralement représenté par la lettre r, si nous avons un ensemble de données {x1, ..., xn} contenant n valeurs et la prédiction de l'ensemble de données {y1, ..., yn} contenant n valeurs , alors cette formule pour r est :

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Où n est la taille de l'échantillon, xi est l'échantillon indexé avec i, yi est la prédiction du système correspondant, et $\bar{x}$, $\bar{y}$ sont les moyennes de xi et yi, respectivement.

- **Le kappa pondéré quadratique :** Le Kappa pondéré mesure l'accord entre deux notes, cette métrique varie généralement de 0 à 1. Dans le cas où il y a moins d'accord entre les évaluateurs que prévu par hasard, la métrique peut descendre en dessous de 0.(Arora aman, 2019) a présenté la méthode de calcul du kappa quadratique pondéré comme suit : Tout d'abord, une matrice d'histogramme N x N O est construite, de sorte que Oi, j correspond au nombre d'enregistrements d'adoption qui ont une note i (réelle) et ont reçu une note prédite j. Une matrice de pondérations N X N, w, est calculée en fonction de la différence entre les scores de notation réels et prévus.

$$\kappa = 1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}}$$

b) Métriques pour les problèmes de classification :

- **Précision de classification (Classification Accuracy) :** Il s'agit du rapport entre le nombre de prédictions correctes et le nombre total d'échantillons d'entrée. Cela ne fonctionne bien que s'il y a un nombre égal d'échantillons appartenant à chaque classe**.** (Mishra, 2018)

$$Accuracy = \frac{Number\ of\ Correct\ predictions}{Total\ number\ of\ predictions\ made}$$

- **Matrice de confusion (Confusion Matrix)** : (Kohavi et al., 1998) Matrice de confusion comme son nom l'indique nous donne une matrice en sortie et décrit les performances complètes du modèle (Mishra, 2018) . Supposons que nous ayons un problème de classification binaire. Nous avons quelques échantillons appartenant à deux classes: OUI ou NON. De plus, nous avons notre propre classificateur qui prédit une classe pour un échantillon d'entrée donné. En testant notre modèle sur 165 échantillons, nous obtenons le résultat suivant.

n=165	Predicted: NO	Predicted: YES
Actual: NO	50	10
Actual: YES	5	100

Il y a 4 termes importants:

- Vrais positifs: Les cas dans lesquels nous avons prédit OUI et la sortie réelle étaient également OUI.
- Vrais négatifs: les cas dans lesquels nous avons prédit NON et la sortie réelle était NON.
- Faux positifs: les cas dans lesquels nous avons prédit OUI et la sortie réelle était NON.
- Faux négatifs: les cas dans lesquels nous avons prédit NON et la sortie réelle était OUI.

La précision de la matrice peut être calculée en prenant la moyenne des valeurs situées sur la « diagonale principale », c'est-à-dire

$$Accuracy = \frac{TruePositives + FalseNegatives}{TotalNumberofSamples}$$

Confusion Matrix constitue la base des autres types de mesures.

- **Rappel (Recall):**(Kulhare, 2017), TP désigne la taille de l'ensemble positif vrai, FP désigne la taille de l'ensemble faux positif et F N désigne la taille de l'ensemble faux négatif, la précision et le rappel sont définis par :

$$\text{recall} = \frac{TP}{TP + FN}.$$

- **Précision :**

$$\text{precision} = \frac{TP}{TP + FP}$$

- **F1 Score** : Le score F1 peut être interprété comme une moyenne pondérée de la précision et du rappel, où un score F1 atteint sa meilleure valeur à 1 et son pire score à 0**.** La contribution relative de la précision et du rappel au score F1 est égale :

$$F1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}},$$

Afin d'avoir une vue générale sur les travaux déjà développé dans le même contexte d'étude qui est la notation automatique des réponses courte nous fournissons une section qui regroupe tous ces travaux

4. Revue de la littérature sur l'apprentissage automatique :

Dans cette section, nous abordons un ensemble de travaux (les plus récents) dans le domaine de notation automatique, ces travaux se concentrent sur les systèmes qui utilisent l'apprentissage automatique, les systèmes qui essaient de résoudre les problèmes de notation pour la langue arabe, et aussi les travaux liés au même projet entamés ces dernières années, afin de faire une synthèse sur les différents systèmes qui utilisent l'apprentissage automatique comme notre système sera basé.

Le tableau **« Tableau 1 »** représente tous les articles des systèmes de notation automatique des réponses courtes pour lesquelles nous avons fait une synthèse :

Numéro d'article	Titre	Citation
1	Auto Grader for Short Answer Questions	(Patil et al., 2018)
2	EVALUATING SEMANTIC ANALYSIS METHODS FOR SHORT ANSWER GRADING USING LINEAR REGRESSION	(Nau et al., 2017)
3	An automatic short-answer grading model for semi-open-ended questions	(Zhang et al ,2019)
4	Deep Learning + Student Modeling + Clustering: a Recipe for Effective Automatic Short Answer Grading	(Zhang et al., 2016)
5	Automated Essay Scoring System By Using Support Vector Machine	(Rocy Martinez, et al. 2013)
6	Automatic Essay Scoring	(Kenton et al.,2010)
7	A Short Answer Grading System in Chinese by Support Vector Approach	(Wu et al.,2019)
8	Fast and Easy Short Answer Grading with High Accuracy	(Sultan et al., 2016)
9	AUTOMATIC SHORT ANSWER GRADING AND FEEDBACK USING TEXT MINING METHODS	(SUZEN, et al.,2018)
10	Sentence Level or Token Level Features for Automatic Short Answer Grading?: Use Both	(Saha et al., 2018)

Tableau 1: les systèmes de notation automatique utilisant l'apprentissage automatique

4.1.Systèmes qui utilisent le Machine Learning :

Plusieurs travaux ont adopté les techniques d'apprentissage automatique (Machine Learning) pour la résolution des problèmes de correction automatique des réponses courtes (Automatic Short Answer Grading).

[1] Le système (Patil et al., 2018) est un système basé sur l'apprentissage automatique qui classe automatiquement les réponses en fonction des réponses de référence données. le modèle se compose de deux sous-modèles : la modélisation des phrases et la mesure de

similarité. Dans la modélisation des phrases, il utilise une architecture siamoise (Mueller et al.,2016) composée de quatre sous-réseaux, un réseau pour la réponse d'étudiant, la réponse référence et deux réponse corrects des étudiants obtenus par l'application de KNN, ces réseaux sont pour obtenir des représentations de phrases. Chaque sous-réseau comprend également 3 couches: couche d'intégration de mots, une couche Bi-LSTM pour coder la signification sous-jacente exprimée dans une phrase, et une couche d'attention qui s'intéresse à calculer un poids pour chaque annotation de mot obtenus par Bi-LSTM en fonction de l'importance, afin de combiner la sortie de chaque réseau dans un réseau entièrement connecté pour mesurer la similarité entre la réponse d'étudiant et la réponse référence le résultat sera utilisé dans une couche de régression logistique pour calculer l'exactitude de la réponse de l'élève. L'ensemble de données utilisé pour cette étude et celle de Student Response Analysis (SRA) accessible au public, la partie SciEntsBank de l'ensemble de données est exploité. Cet ensemble de données comprend 135 questions provenant de divers domaines des sciences physiques. Il a une réponse courte de référence et 36 réponses des étudiants par question. Ce modèle hybride siamois a atteint une précision de 76%.

[2] L'étude présenté par (Nau et al., 2017) se concentre sur la notation automatique des réponses courtes écrites en langue portugaise, ils combinent l'analyse sémantique latente (LSA)et une méthode de similarité entre les concepts sur le chemin Word Net (Fellbaum, 1998) pour les utiliser dans un algorithme de régression linéaire pour prédire les scores de 76 réponses courtes à trois questions écrites par des élèves du secondaire, en les comparant aux scores humains. Ils calculent l'indice de similitude entre les concepts avec la bibliothèque WordNet Similarity (Pedersen et al.,2004) en considérant un vecteur ayant la taille de la réponse de référence prétraitée, Chaque mot de la réponse de référence a été comparé à tous les mots de la réponse de l'élève prétraité, et la similitude la plus élevée trouvée occupait une position du vecteur. Au final, la moyenne arithmétique des valeurs de ce vecteur sera calculée. Les résultats obtenus avec la combinaison de LSA avec WORDNET sont beaucoup mieux que celles de chaque méthode séparément avec une précision= 0,811.

[3] (Zhang et al., 2019) est un modèle pour la notation des questions semi-ouverte (les questions qui n'ont pas une réponse correct précis , les questions qui basent sur la compréhension) , il se concentre sur les connaissances , tel que les connaissances générale de domaine (domain-general knowledge) extraits depuis Wikipédia avec les connaissances spécifique de domaine(domain-specific knowledge) qui sont les réponses des étudiants , afin de générer un dictionnaire de mot pour ces connaissance en appliquant l'algorithme CBOW,

d’après ce dictionnaire pour chaque mots de la réponse d’étudiant le vecteur de contexte sera extrait pour qu’il sera utilisé dans un LSTM afin de conserver la séquence des mots et prédire le score avec SOFTMAX CLASSIFIER. L’ensemble des données utilisé est pour 5 questions de compréhension de la lecture en chinois appelé (CRCC) ont été obtenues à partir des examens finaux des élèves de 8e année et 2 question de compétition de Kaggle de compréhension de lecture.Les résultats ont démontré que l'intégration à la fois d'informations générales et spécifiques à un domaine améliorait considérablement les performances de classement automatique sur des questions semi-ouvertes.

[4] Le Système de (Zhang et al., 2016) a abordé la tâche de l'ASAG à travers l'ingénierie des fonctionnalités et l'exploration de meilleures approches ML telles que Réseau de croyances profondes (Deep Belief Network). Dans cette étude le système a exploré les fonctionnalités des modèles Réponse (Ans), Question(Ques), et Étudiant (Stu) individuellement et combinés. Le modèle Réponse (Ans) a identifié cinq fonctionnalités pour mesurer la similarité entre la réponse de l’étudiant et la réponse modèle. Ces cinq fonctionnalités sont : Différence de longueur / Idf apparié max / Similitude cosinus / Similitude texte pondérée / Analyse sémantique latente, le Modèle Question (Ques) inclue le KC (**Knowledge Components/Composants de connaissances)**pour distinguer entre les différents types de questions ,de plus une fonctionnalité « Difficulté de la question » qui représente le niveau de difficulté des questions et enfin le modèle Étudiant (Stu) comprend une combinaison de probabilités d’apprentissage spécifiques au KC et les scores des étudiants spécifiques au KC. Dans cette étude le système utilise un total de six méthodes de ML y compris le Réseau de croyances profondes **DBN** avec cinq autres approches de ML traditionnelles tel que : **SVM, LogR, NB, DT, ANN**. Deux expérience sont étudiées dans ce système dans la première expérience ils ont utilisé deux classificateurs qui sont **LogR** et **SVM**, ensuite, dans l'expérience 2, Ils comparent DBN à cinq classificateurs ML classiques sur le meilleur modèle de fonctionnalité produit dans l'expérience 1. La Précision(l’Accuarcy), la zone sous la courbe (AUC), la précision, le rappel(recall)et la mesure F sont utilisées pour évaluer la performance de divers classificateurs. Les résultats de cette étude ont montré que les comparaisons entre les différents modèles de classification montrent que le DBN surpasse toutes les autres méthodes sur l’Accuracy(précision) , l'Auc(zone sous la courbe), la précision et la mesure F. Lors du rappel(Recall), **DBN** est légèrement moins performant que **SVM**. Donc pour finir, les résultats suggèrent que DBN est le meilleur parmi les six classificateurs, suivi par SVM et NB est le moins performant.

[5] (Rocy Martinez,et al., 2013) le but de ce Système est de créer un système de notation de dissertation automatique (AES) en utilisant machine a vecteur de support (SVM) qui remplacera avec succès les évaluateurs humains. Les scores de la sortie sont divisés en catégorie et varie de 0 à 3 donc c'est un système multi-classes. Un classificateur SVM un-contre-un est utilisé dans cette étude. L'étape de l'extraction des fonctionnalités est une étape très importante dans ce processus car elle influencera sur les résultats et la précision du model. Pour la formation, les données d'entrée sont un vecteur de caractéristiques extrait des textes d'essai et les données de sortie sont les valeurs de score catégorielles évaluateurs humains, allant de 0 à 3. L'ensemble de données est composé de 1 726 essais (1450 pour la formation et 276 pour les test) et le score résolu varie de 0 à 3. Les performances du modèle ont été mesurées par la précision qui permet au système de de comparer le modèle avec des évaluateurs humains. Pour finir le modèle proposé a eu une précision de 67.75%, cette précision est comparable aux évaluateurs humains formés pour la tâche de notation des essais.

[6] (Kenton W. Murray,et al ., 2010) est un système qui a utilisé la technique régression linéaire pour sa facilité d'interopérabilité et il nous a montré comment elle peut être utilisée pour classer automatiquement les essais sur des tests standardisés et comment à partir des entités textuelles on peut prédire le score, noté y, sur la base des caractéristiques x extraites d'un essai donné. Les données se composent de 8 ensembles d'essais différents de longueur variables d'où les tests avaient été administrés aux étudiants américains de la 7e à la 10e année et étaient écrits en anglais. Pour l'extraction des fonctionnalités il extrait deux types de fonctionnalités de texte:

- Des caractéristiques simples et superficielles des essais, telles que la longueur des caractères et la longueur des mots, (Caractéristiques denses)
- N-grammes de partie du discours. (Caractéristiques clairsemées)

Concernant les mesures d'évaluation, les résultats sont évalués sur la corrélation de Pearson, l'erreur absolue moyenne et le kappa pondéré quadratique dont les paramètres des caractéristiques denses, clairsemées et combinées. Les résultats montrent que la longueur est l'une des caractéristiques les plus marquantes de tous les ensembles d'essais et le 1ér ensemble a eu un résultat de : r=83%, MAE=68%, Kappa=80% sur les caractéristiques combinées.

[7] (Wu et al., 2019) c'est un système de notation à réponse courte en chinois. Ils ont construit un système basé sur des approches standard d'apprentissage automatique (SVM et SVR) et ils le testent avec un corpus traduit de deux corpus publiquement disponibles en anglais(SciEntBank et Ensemble de données de structure de données fourni par **(**Mohler et al.,

2009),le système extrait d'abord les caractéristiques de similitude du texte et les caractéristiques sont utilisés dans un modèle vectoriel de support, les réponses du premier corpus sont notées de 0 à 5 tandis que les réponses du deuxième corpus sont Classées comme correctes / incorrectes. Pour une évaluation des résultats de régression, ils adoptent le coefficient de corrélation au carré et l'erreur quadratique moyenne. Pour une évaluation des résultats de classification, ils adoptent la précision. Et enfin Le résultat de l'expérience montre que les résultats sur les deux corpus différents sont prometteurs, où les résultats de la régression sur la version chinoise de (Mohler et al. 2011) montre que plus de fonctionnalités peuvent améliorer les performances, et le résultat de la classification sur la version chinoise de l'ensemble de données SemEval-2013, montre que la précision est presque la même. Donc nous pouvons dire que plus de fonctionnalités n'améliorent pas les performances.

[8] L'idée de système (Sultan et al.,2016) concerne l'extraction des bonnes propriétés d'entrée pour le calcul de similarité pour une meilleure correspondance entre la réponse référence (R) et la réponse d'étudiant (S) , qui sont : la similarité en cosinus en utilisant les intégration des mots (word embedding) entre R et S , de similarité entre les proportions à l'aide d'un aligneur de (Sultan et al., 2014) , Le rapport du nombre de mots dans S à celui de R , pondération des termes , afin d'utiliser ces propriétés dans un modèle d'une régression linéaire pour prédire un score réel sur l'ensemble de donnée proposée par (Mohler et al. 2011), et aussi pour assigner une étiquette à une réponse qui montre à quel point elle est appropriée comme réponse à la question, sur l'ensemble de donnée proposée à SemEval-2013 (Dzikovska et al., 2013), en utilisant un classificateur de forêt aléatoire (random forest classifier). Pour observer l'impact d'existence de mot commun dans la réponse et la question, les résultats sont recalculé après l'enlèvement les mots de la réponse existants dans la question (Question Demoting). Ce système a montré une amélioration importante et significative des performances par rapport à (Mohler et al. 2011) et à (Dzikovska et al., 2013).

[9] (SUZEN, et al.,2018) C'est un système basé sur la notation automatique des réponses courtes, les réponses sont regroupées dans des clusters en utilisant le K-means clustering pour prédire les notes en fonction des similitudes entre la réponse modèle et les réponses des étudiants. Ils calculent la distance entre la réponse modèle et celle de l'étudiant, ceci est basé sur le nombre de mots communs, ils évaluent en suite la relation entre ces similitudes et les notes attribuées par les correcteurs. Ils ont constaté dans cette étude que l'évaluation des enseignants dépend fortement du nombre de mots de vocabulaire que les élèves ont utilisés dans leurs réponses. Un tel système peut également être utilisé comme un processus d'apprentissage

supervisé et non-supervisé pour prédire les scores (Linéaire régression est utilisée pour la l'approche supervisé tandis que K-means est pour non supervisé). Dans l'apprentissage supervisé ils créent et évaluent un modèle pour prédire les notes des élèves, en fonction de la distance de Hamming entre la réponse du modèle et la réponse de l'élève. L'ensemble de donnés utilisé dans ce système comprend 29 réponses des étudiants de dix devoirs et deux examens de la classe d'introduction à l'informatique de l'Université de North Texas1. Pour conclure nous pouvons dire que le nombre de mots correctement utilisés a plus d'influence sur les notes que la sémantique ou l'ordre des mots.

[10] L'approche de (Saha et al., 2018) propose une combinaison des fonctionnalités coté jetons (Tokens) et fonctionnalités coté des phrases pour améliorer les performances de notation. Concernant les jetons, nous avons Le score de similitude en se basant sur les chevauchements des mots (Words Overlap) entre la réponse d'étudiant et la réponse modèle, et pour comprendre les réponses partiellement correctes des élèves les créateurs introduits un Histogramme des similitudes partielles (HoPS), son extension aux balises de partie de la parole (HoPSTags), ils utilisent aussi des informations de type question. Parlons maintenant sur les fonctionnalités cotées des phrases, les incorporations des phrases sont obtenues pour la réponse d'élève r, la réponse référence a et la question q tel que :

L'écart d'information entre l'élève et la réponse de référence est calculé par ($r * a$) et ($|| r - a |$), les nouvelles informations attendues dans la réponse ($r * q$ et $| r - q |$) et les nouvelles informations exprimées dans la réponse de l'élève ($a * q$ et $| a - q|$).

La combinaison de ces fonctionnalités sera utilisée dans un modèle de la Régression Logistique multinomiale avec l'ensemble de donnée SemEval-2013 et Mohler. Le système obtient des résultats meilleurs ou compétitifs en évaluation expérimentale, tel que la combinaison des fonctionnalités de niveau de jeton et de phrase montre une amélioration significative par rapport à une utilisation individuelle.

4.2.Systèmes pour la langue Arabe :

Dans le domaine éducatif en langue arabe nous avons constaté qu'il y a très peu de projets de recherche sur le classement des réponses courtes. La langue arabe est une langue connue par la richesse de son vocabulaire qui nécessite un travail approfondi pour l'étudier. Pour cela la tâche de notation automatique en langue arabe est devenu difficile à traiter en raison du manque de ressources de base, il n'y a pas assez de corpus accessible au public pour le classement des réponses courtes en Arabe, manque d'outils de traitement automatique de la langue. Tous ces

difficultés seront une sorte de motivation pour investir dans le domaine de la notation automatique des réponses courtes, enrichir les outils de manipulation de cette langue et aussi la recherche dans ce domaine.

-Parmi les investissements de notation automatique pour la langues Arabe il y a :

[1] Le système de (Hebah et al., 2017) est une approche de notation automatique pour les réponses courtes aux questions à rédaction Arabe , Le processus de notation est basé sur la similitude entre la réponse de l'élève et la réponse modèle, La racine des mots et les synonymes de chaque mot clé dans la réponse de l'élève et la réponse du modèle sont utilisés afin d'obtenir des résultats précis, L'idée principale derrière ce modèle est de calculer le poids de chaque terme dans chaque réponse par rapport à l'ensemble du corpus puis une mesure de similitude cosinus est utilisée à cette fin pour obtenir la note. Le système utilise l'ensemble des réponses d'un examen de programme officiel jordanien pour le cours d'histoire de la Jordanie (JH)comprenant 11 questions et 1 réponse modèle pour chaque question, les résultats expérimentaux ont montré que l'approche proposée a obtenu des scores compétitifs par rapport à d'autres approches.

[2] Les créateurs de système (Gomaa et al., 2014) font l'apparition de la notation automatiques des questions de test en Arabe , cette recherche se concentre sur l'application de plusieurs mesures de similarité entre la réponse d'élève et la réponse modèle comme les similarité basée sur les chaines , les similarités basé sur le corpus , la similarité basée sur la connaissance , en les comparant avec la note obtenus manuellement pour produire le score automatique final de la réponse de l'élève, le système traduit les réponses des étudiants en anglais pour pallier le manque de ressources de traitement de texte Langue arabe, les résultats obtenue par la traduction sont très proches à celle de texte original en arabe , la traduction humain est un peu plus performante que la traduction automatiques.

4.3.Travaux développés dans le cadre du même projet:

[1] Mesures de similarité syntaxiques pour un système d'évaluation automatique des réponses courtes (A. ABDALLAH et al., 2018) est un Système d'évaluation automatique des réponses courtes. Ce système prend en entrée une Réponse modèle (RM) et Réponse de l'étudiant(RE). Pour chaque RM et RE le prétraitement des réponses est fait (lemmatisation et la normalisation.), ensuite un ensemble des approches de similarités vont être appliqué sur les deux réponses et enfin le résultat de la similarité sera passé à la note que le professeur veut accorder à cette question soit avec le classifieur de K-means ou la multiplication (X5). Les word embeddings (WE) sont utilisés parmi les approches du « corpus based » pour combiner les

approches syntaxiques de ce travail. **STS** (String text similarity), **TFSS** (Term frequency in string similarity) sont des méthodes de similarité syntaxique proposées.

STS est une méthode basée sur l'algorithme de sous-séquence commune (LCS) avec des modifications et des normalisations. **TFSS** repose sur le modèle BOW, une phrase est représentée comme une collection de mots non ordonnée. Pour la notion du Dataset ils ont utilisé **« Cairo University dataset »** (Gomaa et al.,2013) comme dataset principal de ce travail. Le système a obtenu des meilleurs résultats avec le classifieur K-means avec K=11, **Dice** a donné la meilleure corrélation de Pearson avec un pourcentage de 82,62% en utilisant un stem léger pour data set Gomaa.

[2] Le travail élaboré lors de l'étude présentée par (F. R. OUKINA et al.,2019) s'agit d'un système de notation automatique des réponses courtes basé sur une approche non supervisée. Le système intègre un outil de manipulation de corpus de test. Cet outil permet de générer et convertir l'ensemble des questions, réponses modèles et réponses des étudiants élaborées par des enseignants en différents formats de textes bien structurés, afin de faciliter leurs utilisations comme dataset pour les systèmes de notation automatiques.

Le système intègre aussi un outil qui génère le corpus spécifique de domaine pour la création d'un espace sémantique. Par le bien de ce travail, le dataset « AR-Dataset » est construit et évalué dans une approche de notation automatique qui se base sur l'utilisation des caractéristiques obtenues par les différentes méthodes de similarité comme entrée pour un algorithme K-means afin de les classifier dans un ensemble de classes appropriées ayant une échelle de note.

Le système mène à un résultat de RMSE=1.05 et CP=68,38% par rapport à la moyenne des notes de deux enseignants.

4.4.Synthèse d'analyse des travaux connexes :

Dans cette section nous allons faire une étude comparative des travaux connexes mentionnés ci-dessus dans l'approche d'apprentissage automatique d'où pour chaque système nous extrayons les propriétés, l'ensemble de données, qui sont utilisés pour les modèles d'apprentissage automatique, les mesures pour évaluer la performance de modèle et les différentes techniques de calcul de similarité des phrases.

Dans ce qui suit, nous allons synthétiser les caractéristiques des systèmes notés dans le tableau **« Tableau 2 »**

N=° Article	Propriétés	Ensemble de données	Techniques ML	Métriques	Calcul Similarité des textes
1	Intégration de mots (wordembedding) de : - réponse d'étudiant - réponses références -Intégration des phrases pondérées (Weighting Sentence Embedding) -la somme pondérée des annotations de mots -score de similarité	135 questions de domaines des sciences physiques de SciEntsBank.	-KNN -LogR -BI-LSTM - FCNN	-MSE= 0.16 - Précision =76%	*-Similarité Cosinus* *-FCNN*
2	-Score de similarité entre réponse d'étudiant et réponse référence	-réponses écrites en portugais de 3 questions.	-LR	R= 0,811, MAE= 1,325, RMSE= 1,680	*-LSA* *-WordNet*
3	Intégration des mot (Word Embedding) de : - connaissances générale de domaine -connaissances spécifiques de domaine	- Corpus de compréhension de la lecture chinoise - compétition de Kaggle de compréhension de lecture	-LSTM -softmax classificateur	QWKappa=0.94	
4	- Différence de longueur. - IDF correspondant au maximum. - Similitude cosinus. - Similitude de texte pondérée. - Analyse sémantique latente. - Difficulté des questions	-158 étudiants Et 482 questions différentes. -ce corpus de formation comprenait 16228 réponses courtes.	-DBN -NB - LogR - DT - ANN -SVM	-Six Classifiers Précision = 85%,(avec DBN)	*-LSA* *- similitude de texte pondérée*
5	**1 / Terme-document** - Vecteurs de documents - Fréquence du terme - Fréquence du document - Fréquence inverse des documents - Pondération des fréquences des documents et des termes inverses - Réduction des fonctionnalités **2 / Complexité terminologique:** - Longueur de l'essai -Nombre de caractères- - Nombre de condamnations -Durée moyenne de la peine	-site Web de Kaggle (2012) -L'ensemble est composé de 1 726. -. 1 450 la formation et 276 pour les tests.	-SVM	Précision=67,75 %	*Non mentionné*

	- Longueur moyenne des mots - Nombre de clauses				
6	-Caractéristiques denses: - Nombre de points d'exclamation - Nombre de points d'interrogation - Nombre de mots « difficiles » (vocabulaire) - Nombre d'erreurs d'orthographe - Nombre de mots vides -Caractéristiques clairsemées: Les n-grammes de partie du discours. -fonctionnalités combinées(clairsemées et denses).	-Kaggle -réponses dépendantes de la source et réponses persuasives / narratives / expositoires.	-LR	Corrélation=83 % MAE=68% Kappa=80%	*Non mentionné*
7	-unigram_ rappel -unigram_precision - unigram_F_measure -log_bleu- rappel - log_bleu_precision - log_bleu_F_measure -différence dans la longueur de la phrase (caractère) - différence absolue de longueur de phrase (caractère) -différence dans la longueur de la phrase (terme) - différence absolue de longueur de phrase (terme)	-SemEval-2013 - (Mohler 2009),DataSet -Chinese Corpus Translation **.**	-SVM - SVR	**-SVR :** -R^2 : Toutes les fonctionnalités : 0.083041 -RMSE : Toutes les fonctionnalités : 1.173427 **- SVM :** -Précision : Toutes les fonctionnalités : 59.569%	*-Les approches traditionnelles de TE*
8	-score de similaritéentre les proportions de mot -score de similarité cosinus - Pondération des termes	-Mohler 2011 -Sem-Eval 2013	-LR -RF	r=0. 592 RMSE=0. 887	*-Alignement* *- Similarité cosinus*

	-le rapport du nombre de mots dans la réponse de l'élève à et la réponse de référence				
9	-fréquence du terme (TF/ termfrequency).	29 réponses des étudiants de dix devoirs et deux examens de la classe d'introduction à l'informatique de l'Université de North Texas1.	-k-moyen(k-means) - LR	r=0.82 -MSE pour tm: 0,09 -MSE pour MM : 0,1 -Précision pour mm = 0,17 -Précision pour tm=0.25	*-Distance hamming*
10	-Fonctionnalités au niveau de la phrase -Fonctionnalités au niveau de jeton	-Ensemble de données sur l'industrie à grande échelle -SemEval 2013 -Molher	-Régression logistique multinomiale	**L'ensemble de données de l'industrie à grande échelle :** Acc =0.6636, M-F1 =0.6309, W-F1 =0.6558 **SemEval:** Acc =0.7926, M-F1=0.7858, W-F1 = 0.7910 **MolherDataset:** CP=0.570, RMSE= 0.902	*- Similarité cosinus*

Tableau 2: Synthèse des travaux

➢ Observations et critiques :

Après un tour d'horizon sur les travaux dans le domaine de l'évaluation automatique, qui dépendaient de l'apprentissage automatique, afin de noter les réponses des étudiants, où il a été observé que la plupart des systèmes s'appuyaient sur l'apprentissage Automatique supervisé avec leurs différent types(Classification/régression), ils utilisaient différentes méthodes de calculs de similarité (syntaxique / sémantique).

LR et SVM étaient les algorithmes les plus utilisés dans l'ensemble des travaux, LR était pour la prédiction d'un score pour la réponse d'étudiant et SVM pour classifier une réponse d'étudiant à une classe correspondante. Cependant que la plupart utilisaient les scores de similarités et les pondérations des termes comme des propriétés « features ».

Nous constatons dans notre analyse, la rareté des systèmes qui reposent sur la langue arabe dans la contribution de l'aspect de la notation automatique et surtout ceux qui utilisent l'apprentissage automatique, d'un autre côté nous savons que la langue arabe est une langue très riche en vocabulaire, cette dernière a de nombreux problèmes et plusieurs lacunes dans le traitement des données mais aussi le manque de disponibilité des ensembles de données.

Tous ces difficultés seront une sorte de motivation pour investir dans le domaine de la notation automatique des réponses courtes, enrichir les outils de manipulation de cette langue et aussi la recherche dans ce domaine, de sorte que les travaux présentées ci-dessus nous donnent une large inspiration pour la conception de notre modèle afin d'utiliser les différentes techniques de travaux liées au même projet comme des outils dans notre implémentation.

5. Conclusion

Nous avons parlé au bout de ce chapitre sur les principaux aspects clés et les concepts qui aident à comprendre **et résoudre** les problèmes de la notation automatique ainsi que **les différents investissements** dans ce domaine. Tous ces ressources seront exploitées pour l'élaboration de la prochaine étape qui dépendra grandement de ce qui a été traité afin de concevoir notre système et de construire une méthodologie qui se base sur les forces tirées de cette vue et en travaillant sur les faiblesses pour pouvoir les corriger par la suite. Nous présentons dans la phase suivante la méthodologie proposée pour construire notre système avant d'entamer la phase de mise en œuvre et la discussion des résultats.

Chapitre 3 : Conception du système de l’Evaluation Automatique des Réponses Courtes

Cette partie est consacrée au système que nous souhaitons développer, il s'agit de l'évaluation automatique des réponses courtes en utilisant une approche supervisée de l'apprentissage automatique.

Nous allons présenter notre conception détaillée et les mécanismes qui y sont utilisés, tout en étudiant l'impact de l'apprentissage automatique sur l'amélioration des résultats obtenus précédemment pour l'approche ASAG.

Nous présentons la méthodologie sur laquelle nous allons construire notre système, puis, nous abordons dans la section 4 le plan suivi pour la mise en œuvre du système.

1. Méthodologie :

Le système que nous mettons en œuvre tente de bénéficier de la puissance et de l'efficacité de l'apprentissage automatique, tel que la qualité de ce dernier et l'obtention des résultats compétitifs dépend de l'ingénierie des caractéristiques et le type de modèle.

La figure **« Figure 9 »** : représente les processus suivis dans la conception de notre système

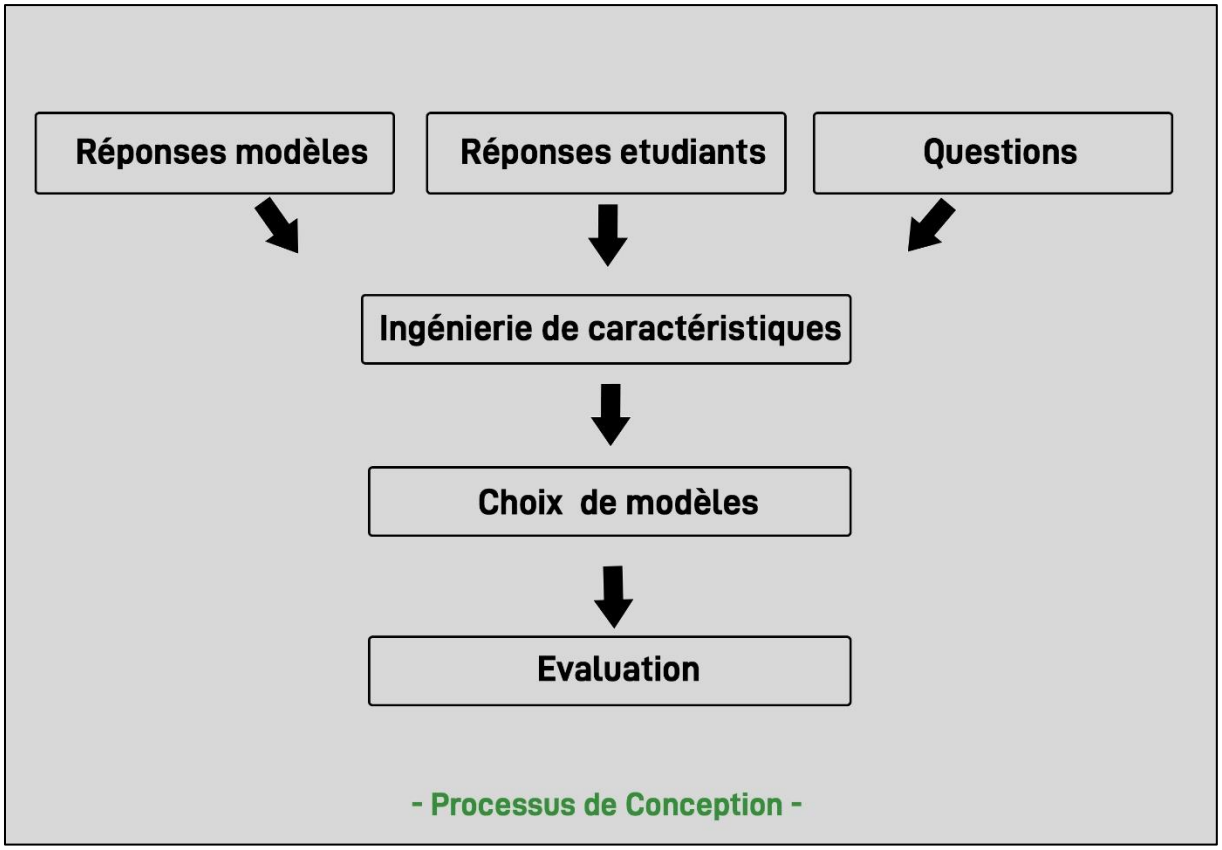


Figure 9: Plan suivi dans la conception

2. Ingénierie de caractéristiques:

Au stade de l'ingénierie des caractéristiques, notre étude repose sur l'extraction des caractéristiques à partir de la réponse de l'élève, la réponse modèle et la question qui jouent un rôle important dans la notation, qui influencent grandement sur le résultat final de l'élève. Nous proposons quatre types de caractéristiques : des caractéristiques liées au texte, des caractéristiques Déviation-Question, des caractéristiques liées au domaine spécifique et générale étudié et les pondérations des termes, comme il est présenté dans la figure **« Figure 10 »**

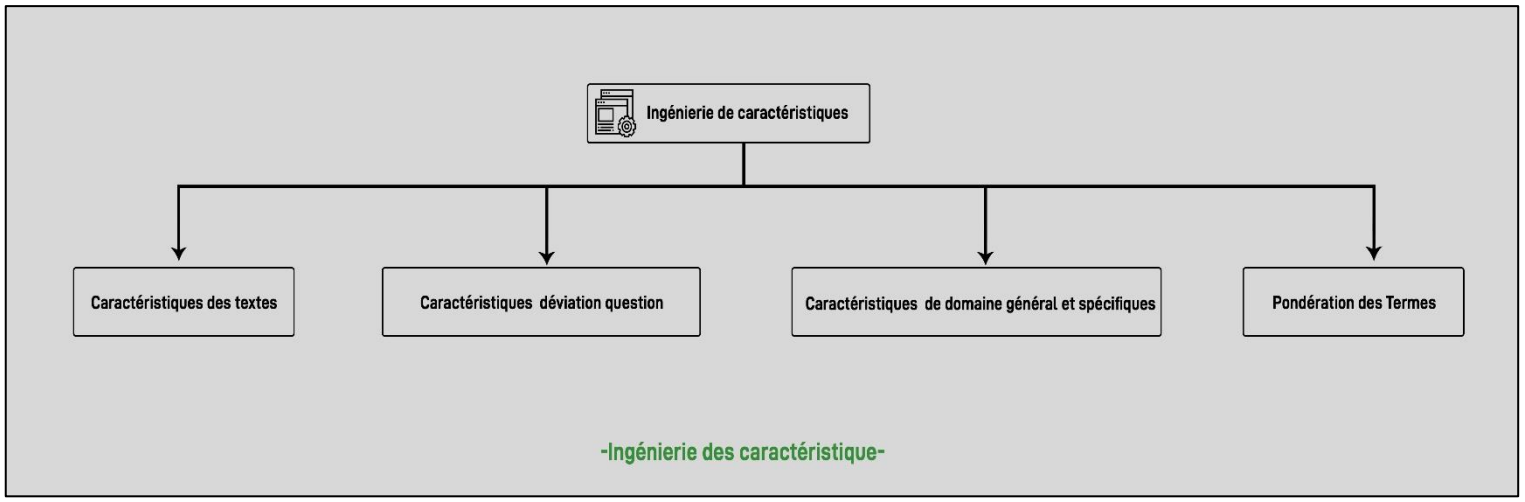


Figure 10: Types de propriétés proposées

Nous fournissons dans cette partie des descriptions sur les différentes caractéristiques proposées afin de donner le détail dans le chapitre **« Implémentation et résultats ».**

2.1.Caractéristiques des textes :

Ce sont les caractéristiques qui dépendent du traitement direct des textes et de l'extraction des caractéristiques qui les composent, elles sont représentées dans la figure suivante **« Figure 11 »:**

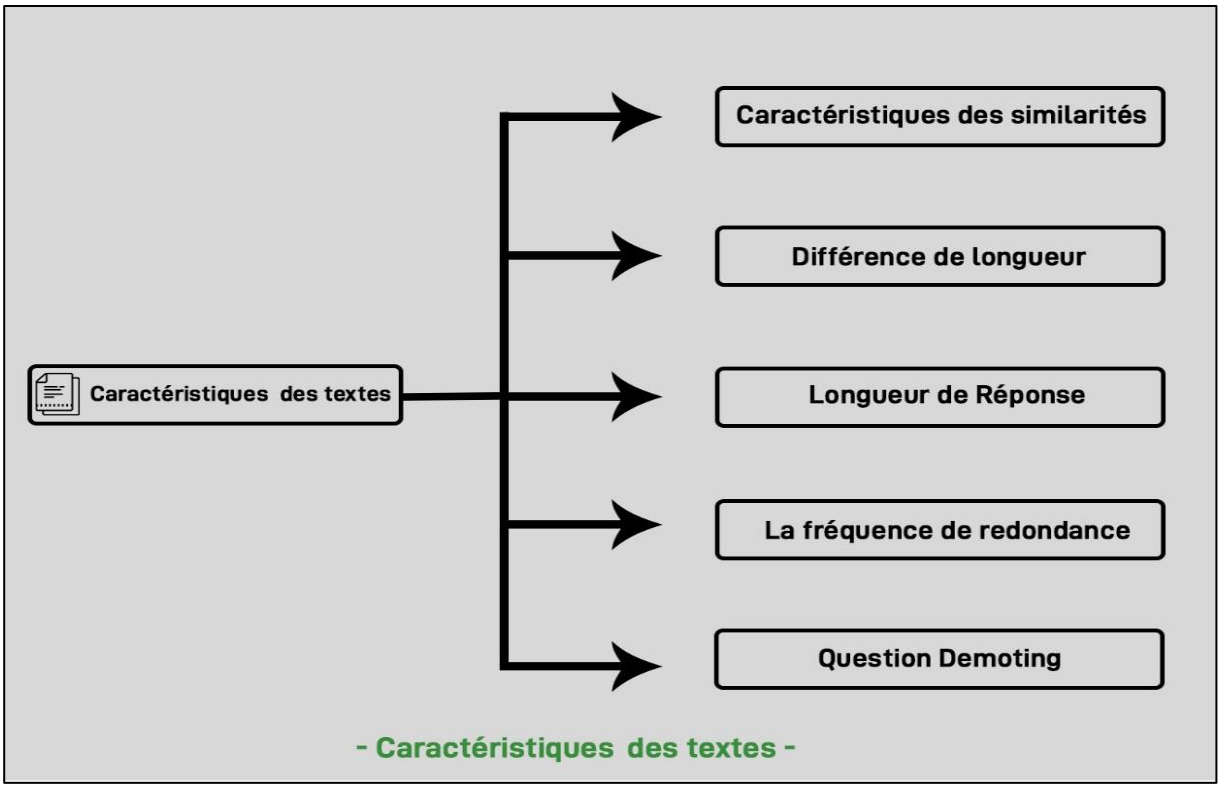


Figure 11: les différentes Caractéristiques des textes

a) **Les Similarités :** Elle constitue l'ensemble des similarités calculées basées sur un corpus et sur des chaines.

b) **Différence de longueur:** Elle représente la différence de la longueur en caractères entre la réponse de l'étudiant et la réponse modèle.

c) **Longueur Réponse :** C'est la longueur en caractères de la réponse étudiant.

d) **La fréquence de redondance :** C'est la somme des mots redondant (les mots qui se répètent plus qu'une fois) par rapport au nombre total des mots de la réponse d'étudiant, cette approche est pour exprimer que la fiabilité de la réponse d'étudiant n'est pas liée fortement à sa longueur tant qu'y a pas de redondance.

e) **Question Demoting :** Son principe est d'observer l'impact de la suppression des mots de la question dans l'ensemble des réponses.

2.2.Caractéristiques déviation-Question :

Parfois, la réponse modèle ne contient pas toutes les informations qui peuvent être extraites de la question et, par conséquent, la réponse de l'élève peut contenir plus de détail. Pour cela,

nous établissons une relation entre la question, la réponse de l'élève et la réponse modèle afin que :

(RM-q, RM*q) : Cette formule représente les nouvelles informations attendues dans la réponse modèle par rapport à la question.

(RE-q, RE*q) : les nouvelles informations exprimées dans la réponse de l'élève par rapport à la question.

(Gap*RE) : Pondération par l'écart d'information tel que **Gap=RM-q ou RM*q**

Où :
« * » : Exprime le produit terme à terme des vecteurs correspondants.
« - » : la différence terme à terme des vecteurs correspondants.

- **Difficulté de Question** : elle représente la difficulté d'une question par rapport à une autre, car les questions qui sont ambiguës entrainent des réponses souvent introduites selon la compréhension de la question par l'élève, ces questions sont considérées comme des questions difficiles. Par contre les questions posées directement qui entrainent des réponses directes et précises sont considérées comme des questions faciles.

2.3.Caractéristiques du domaine général et spécifiques :

Vue La richesse d'une langue et l'abondance de synonymes, se fier uniquement à la comparaison textuelle entre la réponse de l'élève et la réponse modèle, peut ne pas donner les résultats souhaités, donc se fier à la distribution sémantique des mots et à la proximité de leurs sens les uns par rapport aux autres, signifie que les mots possédant le même contexte ont le même sens. Le domaine spécifique est la création de **« l'espace sémantique »** des mots du sujet étudié en considérant la relation de ces mots les uns avec les autres. Le domaine général est le contexte des mots à travers leurs apparitions dans divers sujets selon certaines caractéristiques, où nous utilisons les **« Words Embeddings »** entrainés sur de grands volumes d'informations multi-domaines.

2.4.Pondération des Termes :

Les examens dans lesquels les réponses sont des textes courts, la notation est souvent basée sur l'existence de certains mots-clés qui indiquent l'exactitude de la réponse. Vue la complexité du mécanisme de saisie de la liste de mots-clés lié à chaque question pour

comparer sa présence dans la réponse de l'élève, nous nous sommes basés sur l'ajout des poids aux mots de deux manières, la fréquence des mots selon leur apparition dans le corpus spécifique de domaine, ainsi que le type linguistique du mot (verbe, nom, etc.…).

- **TF-MinMax :** La notion des **TF- MinMax** (fréquence des termes) est une méthode de pondération utilisé souvent dans le domaine de recherche d'information, elle permet d'évaluer l'importance d'un terme contenu dans un document (un texte) relativement à une collection ou un corpus donné. La pondération **TF-minmax** représente le nombre de fois qu'un mot apparaît dans un document, son poids est calculé en fonction de sa fréquence dans le corpus, il s'augmente proportionnellement au nombre d'occurrences du mot dans le document.

  Nous obtenons la normalisation minmax de TF pour chaque mot en utilisant la formule suivante:

$$TF_{min-max} = \frac{TF_{log}}{Max(TF_{log})}$$

  Où $\boldsymbol{TF_{log}}$ est le nombre de fois où le mot donné apparaît dans le corpus, et $\boldsymbol{Max(TF_{log})}$ est le nombre de fois où le mot le plus fréquemment utilisé dans le corpus y apparaît.

- **PosTag :** Part of speech tagging signifie « la partie du baliseur de discours » est un processus qui nous permet d'extraire les informations grammaticales importantes d'une phrase (marquage grammatical d'un mot (ou jetons) dans un texte) telle que le type des mots (verbe, nom, adjectif, etc.). Un mot peut avoir plus d'une partie du discours en fonction du contexte dans lequel il est utilisé. C'est pour cela que cette tâche n'est pas simple.

  Par exemple: Dans la phrase :

  «هي ممارسة تستخدم أساليب التلاعب النفسي (العقلي) لإيذاء و خداع الآخرين » ‘ **ممارسة**’ici est un nom, par contre dans la phrase « يمارس كل أنواع التهديد », ‘ **يمارس**‘ est un verbe.

Apres l'étape de l'ingénierie des caractéristiques nous passons maintenant à la partie du choix des modèles.

3. Les modèles proposés :

Le problème de la notation automatique peut être considéré comme un problème de régression où nous prédisons une note en fonction d'un ensemble de caractéristiques calculées, ou bien un problème de classification où nous affectons l'étudiant à une classe appropriée selon une échelle de notes, et comme l'apprentissage automatique supervisé aide à résoudre ces deux types de problèmes nous choisissons les modèles suivants afin que nous puissions sélectionner le modèle qui nous donne des résultats remarquables :

Pour La Régression **:LR, SVR, ANN, DTR**

Pour La Classification **: LogR, SVC, DTC**

La figure **« Figure 12 »** représente les différents modèles entrainés dans notre conception afin de valider le modèle le plus performant.

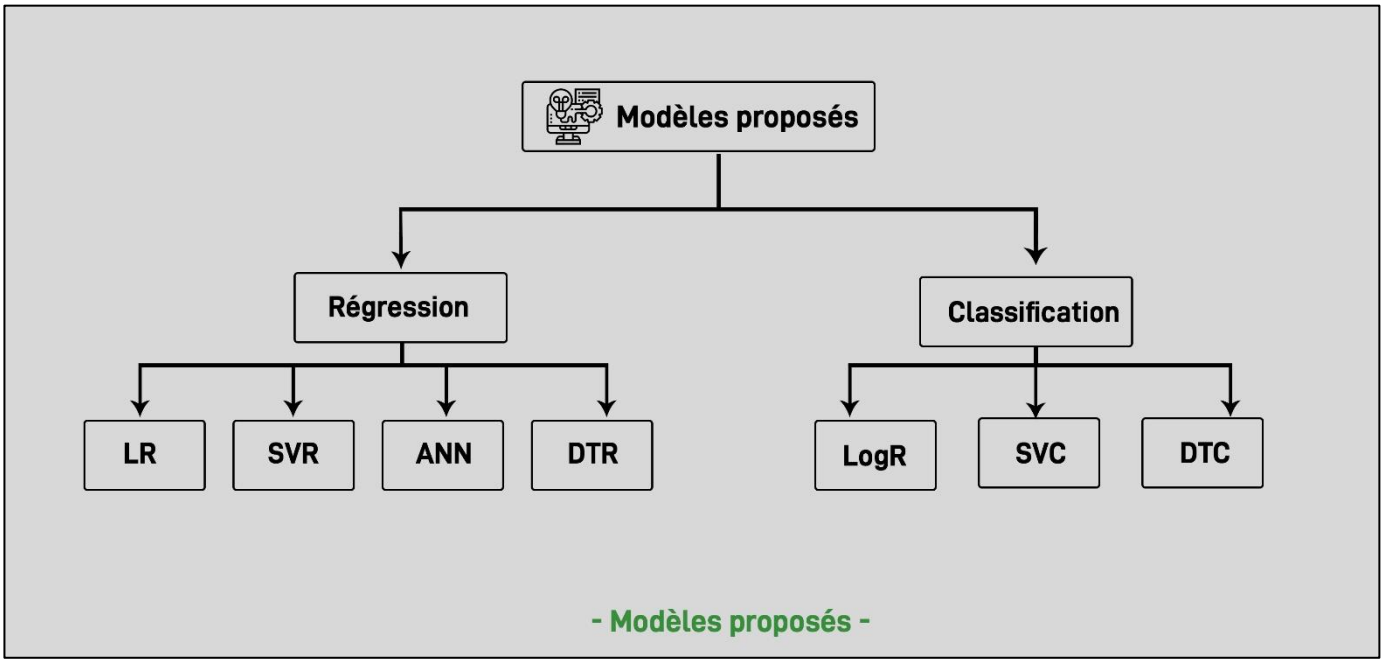


Figure 12:Représentation des modèles proposés

Les processus du déroulement de ces algorithmes sont en détails dans la partie **« Implémentation. »**

La partie suivante présente la manière d'exploitation de cette conception pour l'élaboration de notre système

4. Mise en œuvre des modèles :

À travers la planification présentée ci-dessus et les points inspirés dans l'ingénierie des caractéristiques, nous présentons les méthodes et les techniques que nous suivons pour les différentes phases de création de notre système. Nous fournissons le détail sur l'utilisation de l'apprentissage automatique dans la section **« 4.2 »** pour l'ajustement des données et le processus de détermination du meilleur modèle.

En se basant sur la méthodologie présentée précédemment, la mise en œuvre de notre système aura plusieurs étapes pour atteindre le modèle final, ce qui nous permet de prédire les notes automatiquement, ces étapes seront expliquées en détail ci-dessous, l'illustration suivante **« Figure 13 »** montre ces étapes :

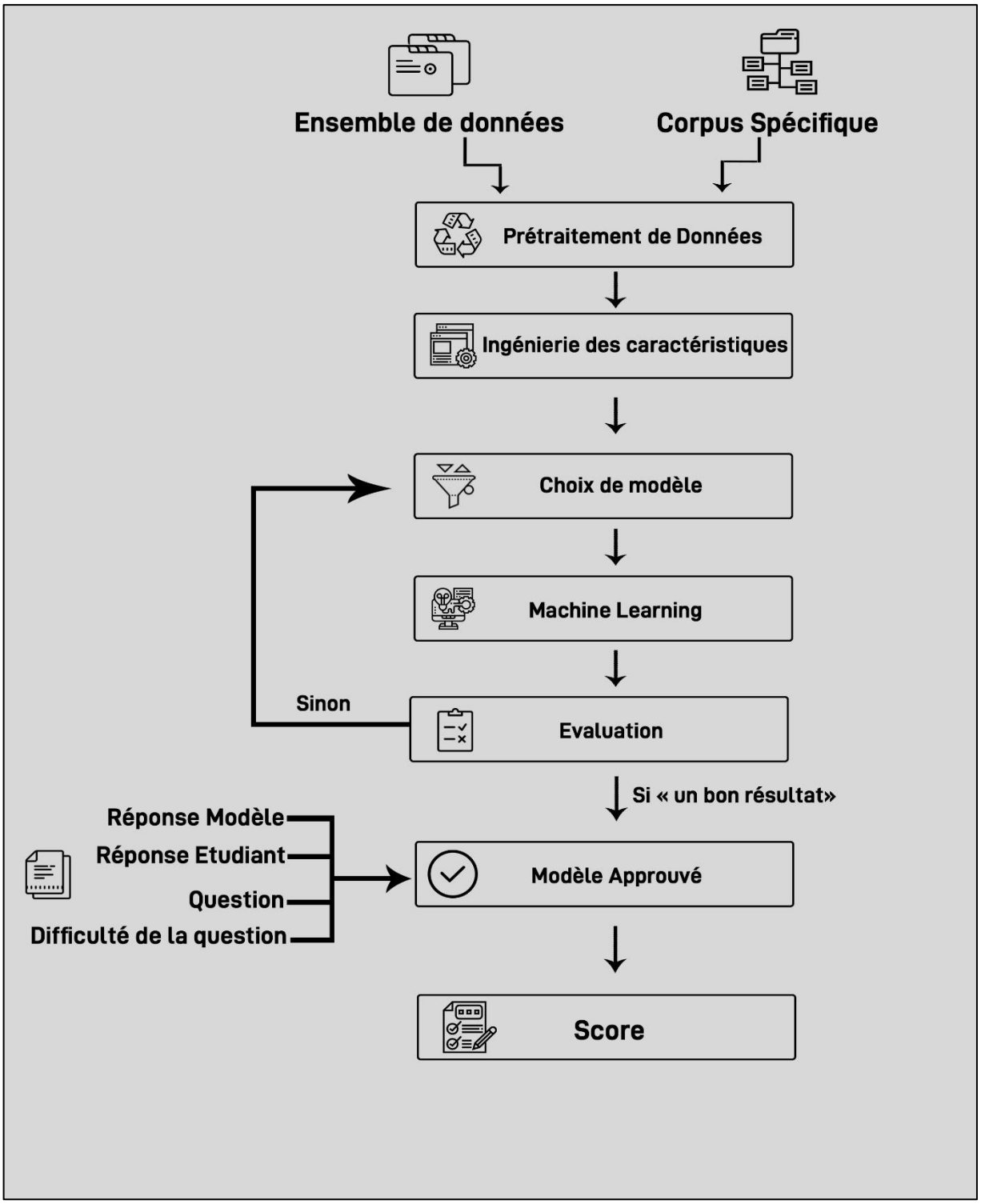


Figure 13: Aperçu de notre Système

Chaque processus sera bien expliqué et détaillé ci-dessous.

4.1.Prétraitement :

Nous utilisons tout au long dans notre travail **« AR-ASAG Dataset »** comme ensemble de données de base pour l'entrainement et le test des modèles proposés. Ce **Dataset** couvre les cours du module cybercriminalité. Pour étudier la performance et la fiabilité de notre modèle nous le testons aussi sur le **« Dataset Mohler ».**

La partie de la préparation de données est la partie la plus importante dans le processus de traitement du langage naturel. Il est nécessaire de nettoyer les données de texte pour mettre en évidence les attributs sur lesquels nous voulons que notre système d'apprentissage automatique reprenne. Le nettoyage (ou le prétraitement) des données comprend généralement un certain nombre d'étapes comme : la suppression des mots vides, des signes de ponctuation la Tokenisation, la normalisation, etc. Ces étapes sont illustrées dans la figure **« Figure14 »**

a) Nettoyage des données:

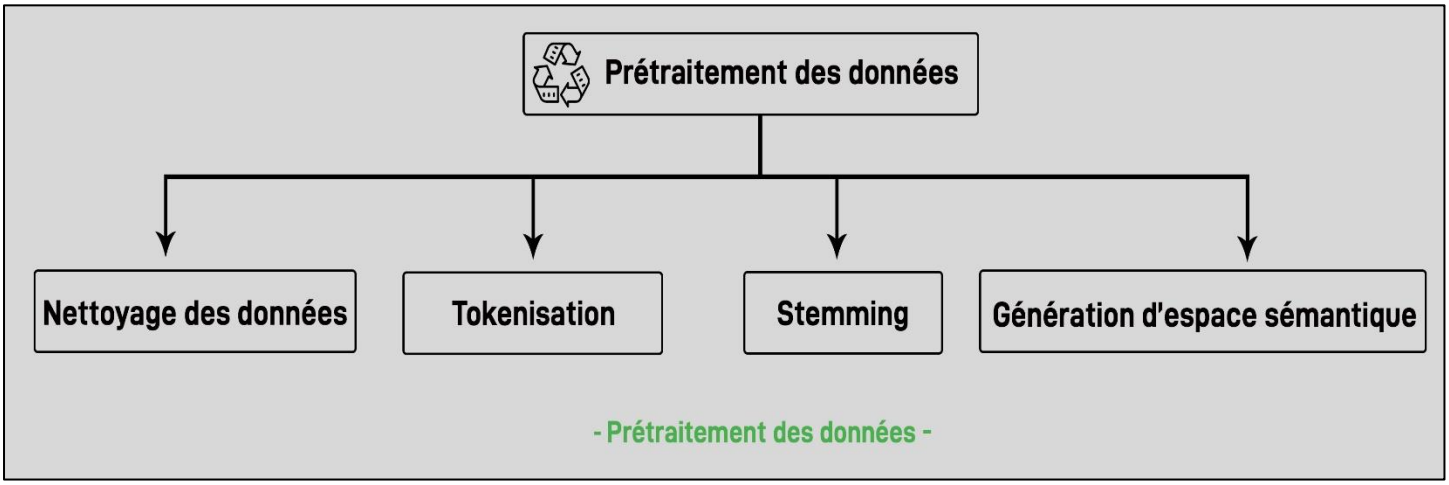


Figure 14: Représentation des différentes étapes de prétraitement des données

L'étape de nettoyage des données est la première étape fondamentale de notre système, elle se compose de deux grande parties essentielles qui sont **« suppression des mots vides »** et **« suppression des signes de ponctuation ».**

Après une étape de Nettoyage des données nous passerons à une étape non moins importante que la première qui est la **Tokenisation.**

b) Tokenisation :

La Tokenisation consiste à découper une séquence de caractères phrase, paragraphe ou un document texte entier en morceaux, appelés jetons.

Exemple :

هي كل سلوك غير قانوني يتم باستخدام الأجهزة الإلكترونية

Tokenisation

هي, كل, سلوك, غير, قانوني, يتم, باستخدام, الأجهزة ,الإلكترونية

Nous avons travaillé avec la tokenisation pour plusieurs raisons parmi ses utilisations nous avons :la facilité de calcul de similarité vu que nous allons travailler avec les vecteurs de contexte de chaque terme, l'utilisation des différentes approches de similarités syntaxiques qui s'applique sur les termes aussi, et enfin pour l'utilisation du Stemming dont nous en parlerons dans la partie suivante.

c) Stemming:

Le stemming fait généralement référence à la normalisation des mots dans leur forme de base ou sous forme de racine. La manipulation des textes est une approche très sensible surtout pour les mesures de similarités, un caractère de plus peut faire la différence, c'est pour cela nous pensons à utiliser le Stemming, et il a prouvé sa performance.

➢ **Types de stemming :**

Pour extraire la tige des mots, il existe deux types de stemming le premier type est le **« stemming léger-light stemming »,** le deuxième type est appelé **« stemming lourd-root stemming ».** Dans les deux types le but du stemming est de supprimer l'affixe (préfixes, infixes et suffixes) d'une chaîne, en renvoyant la tige du mot en sortie.

- **Stemming léger :** C'est le type le moins complexe il consiste à supprimer les affixes sans tenter de retourner la racine du mot.
- **Stemming lourd :** Consiste généralement à la suppression du préfixe et des suffixes connus. Il vise à renvoyer la racine réelle d'un mot, il inclut implicitement **"light stemming"**

Exemples :
(lourd)الأهداف= هدف
(lourd)الجرائم = جرم
(leger)الفيروسات= فيروس
(lourd)المراقبة = رقب
(leger)يتعلمون= علم

- **Les stemmers utilisés :** Dans notre travail nous utilisons différent stemmer connus en langue qui sont :

  - Isri Stemmer (pour un stemming lourd)
  - Tashapyne (pour un stemming lourd et léger)
  - PorterStemmer (Pour l'anglais)

d) Génération d'espace sémantique :

Dans cette étude nous essayons de bénéficier des outils déjà construits comme les générations d'espaces sémantiques développés par (S.ABDELAOUI, 2019) en utilisant la méthode **COALS.**

- **Les Etapes d'élaboration de l'ES :**

Pour générer la matrice d'espace sémantique, nous passons par plusieurs étapes qui sont illustré ci-dessus :

- **Collection des informations spécifiques de domaine :**

  Les informations de domaine que nous voulons étudier sont considérées comme la principale matière pour établir l'espace sémantique, de sorte que nous l'extrayons d'internet, de livres ou d'autres sources qui se concentrent sur le même domaine afin qu'ils soient convertis en texte et seront structuré en fichiers texte afin de faciliter le traitement.

- **Prétraitement de corpus :**

  Nous nous appuyons à ce stade également sur le facteur de nettoyage des données et la suppression des mots et symbole inutiles d'après l'ensemble de corpus de domaine spécifique puis les données prétraitaient seront ajoutées dans un seul fichier texte.

- **Génération des termes uniques** : Depuis le fichier texte extrait d'après l'étape précédente, nous tirons chaque mot de façon unique et le placer dans un fichier afin

que chaque numéro de ligne de mot dans ce fichier sera comme un index pour chaque ligne de la matrice d'espace sémantique.

- **Elaboration de Matrice d'espace sémantique :**

Dans la dernière étape, nous calculons les fréquences pour chaque mot dans le fichier des mots uniques en utilisant la méthode **COALS** à partir de leur apparition dans le fichier du domaine étudié, que nous avons créé dans la deuxième étape.

La figure **« Figure 15 »** montre un aperçu d'un espace sémantique :

	البرامج	السرقة	أمن	بيانات	الهاكر	الهندسة	جرائم	المستخدم	الانترنت
البرامج	0	0.23	0.06	0.1	0.62	0.01	0.33	0.2	0.9
السرقة	0.23	0	0.51	0.66	0.15	0.8	0.9	0.4	0.92
أمن	0.06	0.51	0	0.8	0.07	0.002	0.9	0.001	0.3
بيانات	0.1	0.66	0.8	0	0.9	0.51	0.01	0.66	0.23
الهاكر	0.62	0.15	0.07	0.9	0	0.06	0.62	0.4	0.15
الهندسة	0.01	0.8	0.002	0.51	0.06	0	0.15	0	0.51
جرائم	0.33	0.9	0.9	0.01	0.62	0.15	0	0.07	0.4
المستخدم	0.2	0.4	0.001	0.66	0.4	0	0.07	0	0.15
الانترنت	0.9	0.92	0.3	0.23	0.15	0.51	0.4	0.15	0

Figure 15: Matrice d'espace sémantique

4.2. L'approche Machine Learning :

L'Apprentissage automatique est l'aspect clé de tous les caractéristiques présentées précédemment comme caractéristiques des textes, caractéristiques déviation-question et les caractéristiques de domaine général et spécifiques pour la résolution du problème de la notation automatique dont le but est de prédire un score ou bien faire une classification sur une échelle de notes adéquate.

Apres l'étape de prétraitement et calcul des similarités en créant l'ensemble de données qui représente les caractéristiques sur lesquelles le modèle final est construit, nous enchainons la dernière partie qui est la partie d'apprentissage automatique d'où nous parlons sur le partitionnement des données en données d'entrainement et données de teste **(Data Split)** et la combinaison des fonctionnalités pour l'obtention de meilleur modèle. La figure Suivante **« Figure 16 »** représente le mécanisme suivis dans l'apprentissage automatique :

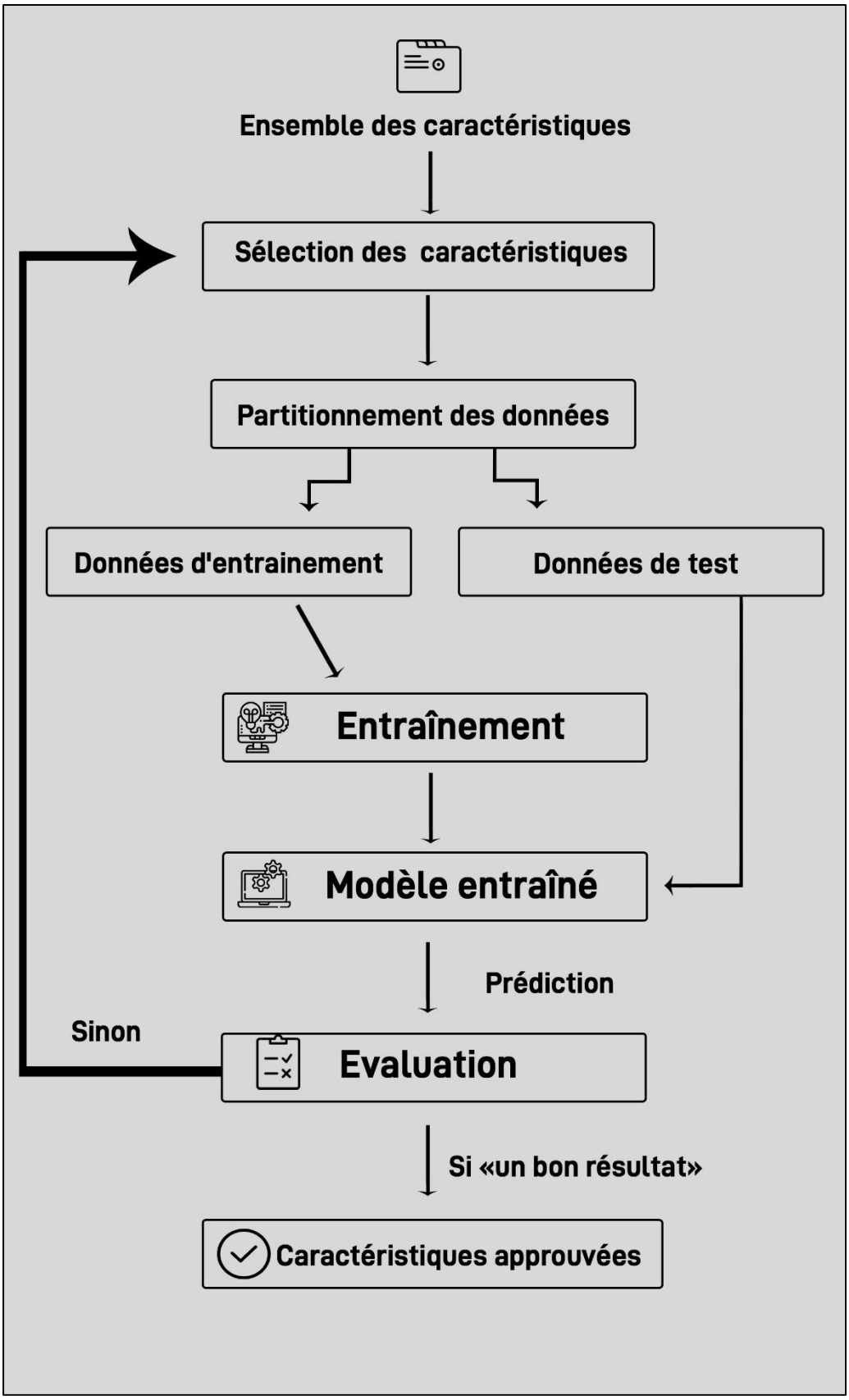


Figure 16: Mécanisme suivi dans l'apprentissage automatique de notre système

Nous expliquons le mécanisme de ce processus dans la partie suivante.

a) Sélection des caractéristiques:

Pour étudier l'impact des caractéristiques choisi sur la note de l'étudiant qui donnent des résultats plus proches que les résultats manuels, nous nous appuyons sur la création d'une harmonie entre les caractéristiques afin d'atteindre le meilleur résultat en termes de précision, c'est-à-dire la moindre différence entre la note manuelle et la note automatique.

Cette procédure est faite de sorte que nous gardons que les combinaisons de caractéristiques qui mènent notre modèle à avoir une précision la plus élevée.

La figure **« Figure 17 »** montre le processus détaillé d'extraction des caractéristiques**:**

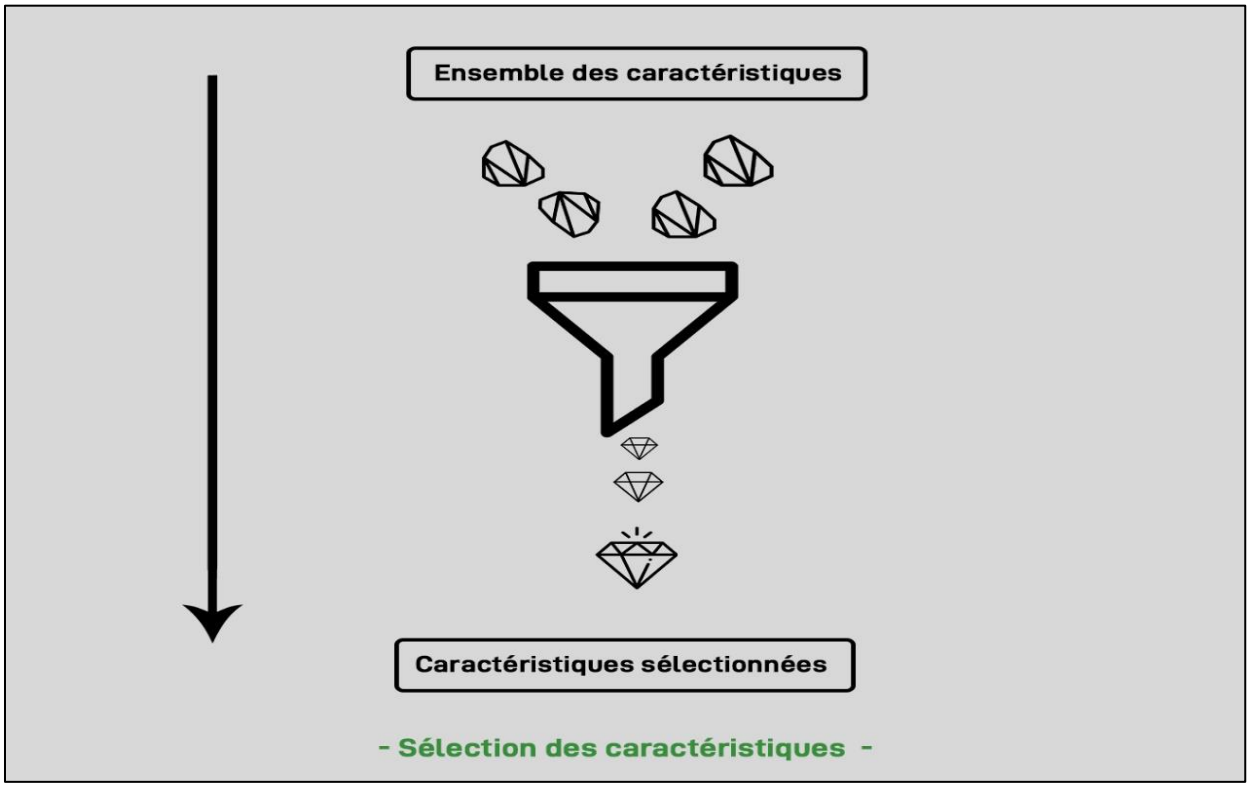


Figure 17: Processus de l'extraction des caractéristiques

b) Partitionnement des données :

Le split de données a un impact très important dans les modèles de l'apprentissage automatique ainsi, c'est une étape indispensable. Le partitionnement de données consiste à diviser l'ensemble de donnée en deux parties **« données d'entrainement »** et **« données de test »** ou de le diviser en trois parties **« données d'entrainement »**, **« données de**

validation » et **« données de test ».** La partie d'entrainement contient le grand nombre de couple que la partie test pour l'obtention d'un bon entrainement pour un meilleur résultat.

c) Entrainement du modèle :

L'étape d'entrainement consiste à ajuster les paramètres de l'algorithme choisi en apprenant à partir de l'ensemble de données et en essayant d'atteindre les facteurs des caractéristiques qui conduisent à des résultats proches aux résultats réels.

d) Test du modèle :

Nous prenons dans la partie test de modèle les données de test comme entrée dans l'algorithme afin de prédire des scores et valider la performance et l'efficacité de l'algorithme.

e) Evaluation :

Une fois le modèle est entrainé et testé il passe par une étape d'évaluation qui est une étape indispensable dans la validation des caractéristiques qui mènent à donner un meilleur algorithme.

L'évaluation se fait en utilisant les différentes mesures de calculs de la performance du modèle qui sont introduite dans le **« chapitre 2 section Concept Fondamentaux de l'apprentissage automatique. »**

5. Conclusion :

La méthodologie utilisée dans l'ingénierie de notre système a été discutée tout au long de ce chapitre, ainsi que la conception finale dans laquelle nous reprenons l'idée.

Dans ce qui suit, nous présentons les techniques utilisées dans l'application et la méthode d'implémentation avec les résultats seront présentées.

Chapitre 4 : Implémentation et Résultats

La création du système final vient après avoir parcouru les étapes expliquées lors de la conception en suivant la méthodologie présenté ci-dessus, avec une analyse des résultats atteints, et à partir de là, dans ce chapitre nous parlons des techniques utilisées, de la méthode suivie pour l'application et de l'efficacité de l'étude que nous avons menée.

1. Implémentation :

La partie implémentation englobe tous les processus suivis lors de la réalisation de notre système final tel que l'ensemble de données utilisé ainsi que sa structure, les techniques de prétraitement, l'extraction des caractéristiques, les processus de l'apprentissage automatique et nous parlons aussi sur l'environnement de développement.

1.1. DataSet :

Nous utilisons dans notre système **« AR-ASAG Dataset »** présenté par (F.R. OUKINA et al., 2019) comme ensemble de données de base pour l'entrainement et le test des modèles proposés. Ce Dataset couvre les cours du module cybercriminalité. Notre Dataset contient 48 questions, dont 48 réponses modèles et une collection de réponses étudiants d'où le total est 2133 réponses avec des notes manuelles comprises entre 0 et 5 évaluées par deux enseignants.

Pour étudier la performance et la fiabilité de notre modèle nous le testons aussi sur un autre Dataset **« Mohler-Dataset »** (Mohler et al., 2011) qui contient 87 questions et un total de 2442 réponses étudiants.

Le tableau **« Tableau 3 »** est un récapitulatif de ce que nous avons présenté ci-dessus :

Dataset	Nombre de questions	Nombre de Réponses Modèle	Nombre de réponses étudiants
AR-ASAG	48	48	2133
Mohler	87	87	2442

Tableau 3 :Informations sur les Dataset utilisées

1.2. Structuration de DataSet :

Pour standardiser notre travail, le rendre global et facile pour l'exploitation, nous structurons chaque dataset en deux fichiers **XML**, de sorte que les réponses des étudiants

sont dans un seul fichier tel que chaque réponse à un **« Question_ID »** qui indique l'identifiant de la question, **« Answer_ID »** qui indique l'identifiant de la réponse d'une question et **« Average_Mark »** qui indique la note manuelle de l'étudiant,

La figure 18 montre la structure du premier fichier **XML** de l'ensemble de données **AR-ASAG Dataset » :**

Figure 18: fichier XML qui représente l'ensemble de données« AR-ASAG Dataset »

La figure 19 montre la structure du premier fichier **XML** de l'ensemble de données **« Mohler Dataset » :**

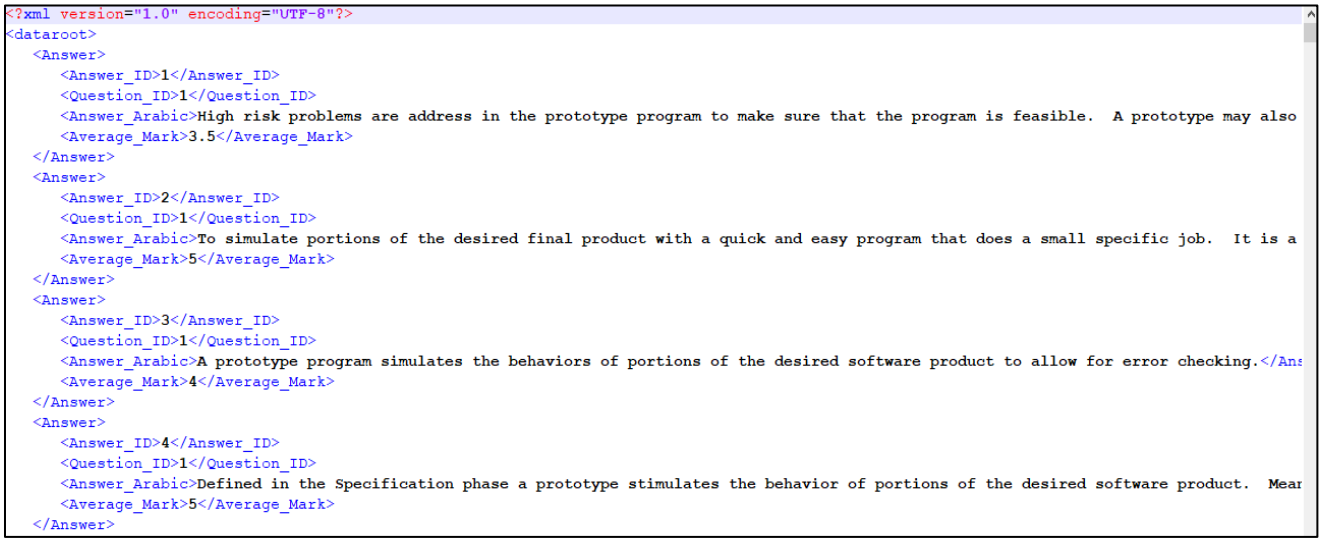


Figure 19: fichier XML qui représente l'ensemble de données « Mohler Dataset »

Le deuxième fichier contient tous les questions avec leurs réponses modèles telles que chaque question a un **« Question_ID »**.

La figure 20 contient la représentation des questions **de l'ensemble de données « AR-ASAG »**

Figure 20: fichier XML qui représente « Questions-Réponses Modèle AR-ASAG Dataset »

La figure 21 contient la représentation des questions **de l'ensemble de données « Mohler-Dataset »**

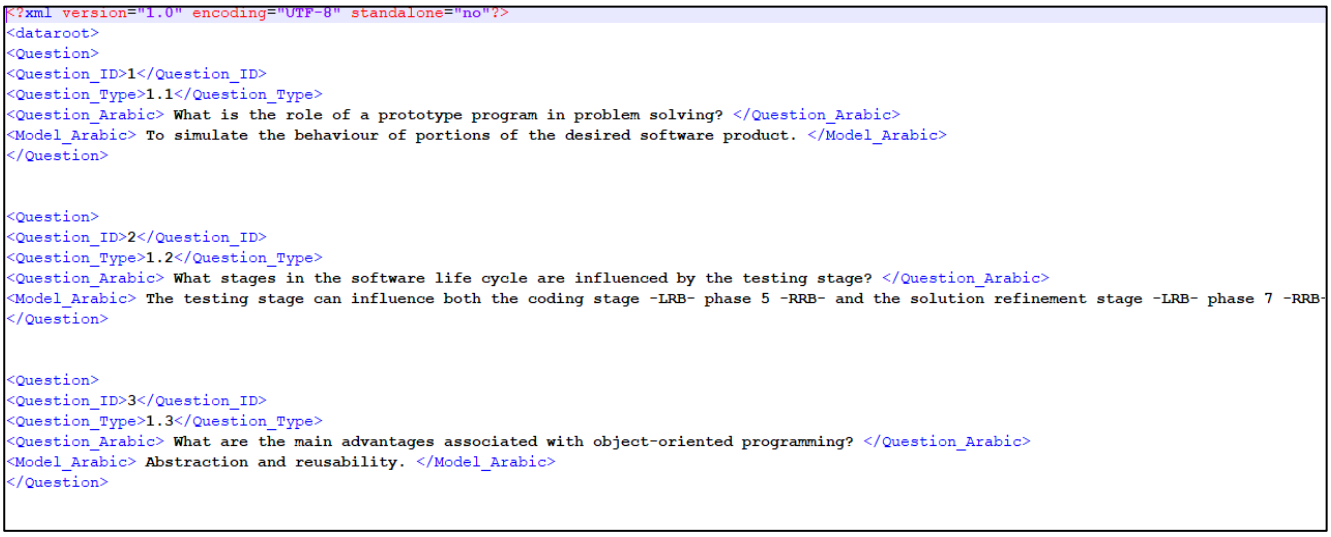


Figure 21: fichier XML qui représente « Questions-Réponses Modèle Mohler Dataset »

1.3. Les techniques utilisées dans le prétraitement :

- **Normalisation :** Plusieurs étapes de nettoyage et de normalisation du corpus combiné ont été appliquer pour texte collecté nous notons :
  - Nettoyage des caractères bruyants, des balises et suppression des signes diacritiques.
  - Normalisation des caractères arabes: la normalisation de (أ, إ, آ) à (ا) et (ة) à (ه).
  - Normalisation de tous les chiffres numériques.

Exemple :

-1تبيض الأموال ، هو إضفاء صفة المشروعية على الأموال القذرة المتأتية غالبا من التجارة غير المشروعة مثل : (المخدرات – بيع الأشياء المسروقة – تزوير العملة أو غيرها...) ، بتوظيفها في نشاطات وهمية على الانترنت.\n

Après la Normalisation

تبيض الاموال هو اضفاء صفه المشروعيه على الاموال القذره المتاتيه غالبا من التجاره غير المشروعه مثل المخدرات بيع الاشياء المسروقه تزوير العمله او غيرها بتوظيفها في نشاطات وهميه على الانترنت

- **StopWords :** une base de données qui contient **841mots** qui sont considérés comme des mots d'arrêt pour la langue arabe comme **: «** ... على ,هو ,هي **»**

  Pour l'anglais, nous utilisons l'ensemble des **stop Word** de la bibliothèque **NLTK** qui contient **179 mots** comme : **«** the, or, and, to … **»**

Exemple :

تبيض الاموال هو اضفاء صفه المشروعيه على الاموال القذره المتاتيه غالبا من التجاره غير المشروعه مثل المخدرات بيع الاشياء المسروقه تزوير العمله او غيرها بتوظيفها في نشاطات وهميه على الانترنت

Après la suppression des StopWords

تبيض الاموال اضفاء صفه المشروعيه الاموال القذره المتاتيه غالبا التجاره المشروعه المخدرات بيع الاشياء المسروقه تزوير العمله بتوظيفها نشاطات وهميه الانترنت

- **Stemming :** nous utilisons le stem léger et lourd **Isri**, **Tashaphyne(lourd – léger)** pour la langue arabe et **Porter Stemming**, **SnowBall** pour l'anglais.

- **Espace Sémantique :** nous générons la matrice d'espace sémantique d'après le corpus spécifique des cours de cybercriminalité. Nous faisons plusieurs expériences lors de la génération de la matrice des mots avec différentes tailles, 9000 mots, 20 000 mots et 30 000 mots pour l'arabe. Pour Mohler le corpus spécifique est obtenus d'après le web pour une taille de 15 000 et 30 000 mots.

1.4. Extraction des caractéristiques :

Cette étape regroupe tous les caractéristiques essentielles de notre système, nous parlons sur les techniques et les algorithmes du calcul de similarités utilisées, aussi sur les différentes pondérations appliquées dans ces calculs, et l'extraction de la difficulté de question en fonction des notes des étudiants.

a) Caractéristiques du domaine général et spécifique :

Les caractéristiques extraites d'après le domaine général et/ou spécifique sont les résultats de calculs de similarité sémantique où nous utilisons deux grands aspects de bases qui sont important dans le calcul de la similarité entre la réponse de l'étudiant et la réponse model.

La première approche c'est l'utilisation de **l'espace sémantique**, c'est une approche basée sur un corpus spécifique qui est le « **Corpus Cyber** »**,** les étapes de constructions sont expliquées dans **« Chapitre 2 section 2 »**

La deuxième approche est l'approche des **Word Embedding** (Incorporation de mots), nous fournissons dans cette partie une explication sur les incorporations de mots, nous parlons sur les techniques qui peuvent être utilisées pour apprendre un mot incorporé, la technique utilisée dans notre système, et les incorporations de mots préformés que nous avons exploités.

L'incorporation d'un mot est une représentation apprise d'un texte où les mots qui ont la même signification ont une représentation similaire. Ces mots sont représentés comme des vecteurs à valeur réelle dans un espace vectoriel prédéfini. Tout au long de notre travail nous intéressons sur la technique **Word2Vec** que nous utilisons pour l'approche des incorporations de mots d'où **Word2Vec** est une méthode statistique pour apprendre efficacement un mot autonome incorporé à partir d'un corpus de texte.

Deux architectures ont été initialement proposées pour apprendre les **Word2vec**, le modèle de sacs de mots continus (**CBOW**: continuous bag of words) et le modèle **skip-gram.**(Mikolov, et al., 2013)

Le modèle **CBOW** apprend l'intégration en prédisant le mot actuel en fonction de son contexte. Le modèle **skip-gram** apprend en prédisant les mots environnants à partir d'un mot courant.

La figure **« Figure 22 »** nous illustre les deux architectures des incorporations des mots :

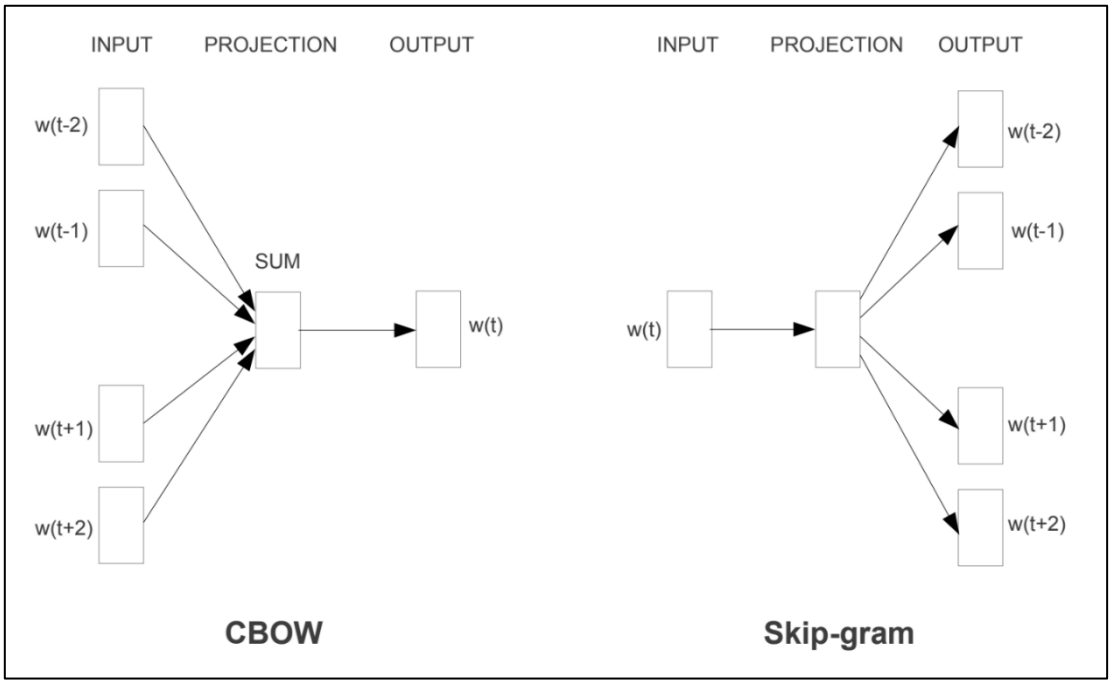


Figure 22: Architectures de modèles CBOW et skip-gram

En pratique, le modèle CBOW est plus rapide à apprendre, mais le modèle skip-gram donne généralement de meilleurs résultats. (Mikolov, et al., 2013)

Dans notre travail nous exploitons des incorporations de mots préformés, afin que nous puissions les utiliser dans nos calculs.

Notre système utilise deux incorporations de mots préformés le premier est **Word Embedding de zahran ,** Une grande quantité de textes arabes bruts provenant de**« Wikipédia arabe », « Corpus Gigaword arabe »** et d'autres sources est collectée dans le système de (Mohamed A. Zahran, et al., 2015) **.** Le nombre de mots dans **« Word Embedding de zahran »** est plus de **6 millions mots.**

Le deuxième est **« FastText Word Embedding »** pour la langue anglaise, il contient plus de **260000 mots.**Le vecteur correspond à chaque mot est un vecteur de **300 dimensions.** Après avoir eu les vecteurs de contexte nous calculons la similarité en appliquant **Cosine Similarity** pour ces vecteurs afin de calculer la similarité avec tous les autres mots d'où :

- Vecteur (réponse) = somme de n vecteurs mots
- Vecteur (référence) = somme de n vecteurs mots

Nous présentons un tableau **« Tableau 4 »** qui contient une représentation des vecteurs de contexte des mots :

Mots	**Vecteurs**
برنامج	[0., 0.582, 0.274, 0., 0., 0.,0., 0., 0., 0.]
المعلومات	[0.582, 0.322, 0.423, 0.27, 0., 0., 0., 0., 0., 0.]
المخترق	[0.274, 0.423,0.,0.33, 0.226, 0., 0., 0.,0.,0.]
المستخدم	[0., 0.27, 0.33, 0., 0.316, 0.21, 0., 0., 0., 0.]
البرامج	[0.,0.,0.226,0.316, 0., 0.316, 0.226, 0.084, 0., 0.]

Tableau 4: Représentation vectorielle des mots

Ces vecteurs de contexte peuvent être pondérer en utilisant des aspects étudié ci-dessus comme suit :

❖ **La Pondération des termes:**

Dans notre système nous avons recouru à la méthode de pondération **TF-MinMax** qui est généré à partir de corpus du domaine étudié et la partie du baliseur de discours **PosTag**. Nous utilisons **« StanfordCoreNlp »** pour la représentation des mots et par la suite nous pouvons attribuer des poids à chaque type de balises pour que nous puissions les utiliser dans nos calculs de similarité comme une pondération.

Nous constatons généralement que dans la littérature les verbes ont un impact important sur le sens de la phrase, pour cela nous donnons le poids le plus élevé pour les verbes qui est **0.5**, **0.3** pour les noms et **0,2** pour le reste. La somme de tous ces poids est égale à 1. **Verb+nom+autre=1.**

Nous pondérons chaque vecteur de contexte (vecteur obtenu par **l'ES** ou par les **WE**) des mots existant dans la réponse de l'étudiant par le poids approprié en utilisant la pondération **TF-MinMax** ou **PosTag.**

Exemple :

Le tableau 5 est un exemple des pondération (Minimax-PosTag)

« الجريمه الالكترونيه سلوك غير قانوني »

Mots	Poids MinMax	Poids PosTag
الجريمه	0.7048	0.3
الالكترونيه	0.7055	0.3
سلوك	0.614	0.2
غير	0.211	0.2
قانوني	0.5815	0.3

Tableau 5: Exemple des mots avec leurs poids

- **TFminMax :**

الجريمه=['0.077 0.095 0.265 0.077 0.032 0.045 0.032 0.032 0 0.045]*0.7048

الالكترونيه=['0.055 0.241 0.071 0.084 0.032 0 0 0.055 0.063 0.045 0]*0.7055

سلوك=['0.071 0.077 0.084 0.1 0.032 0.032 0.032 0 0 0 0 0.063 0]*0.614

غير=['0 0.032 0.032 0.045 0.063 0.045 0.032 0.032 0.045 0.045]*0.211

قانوني =['0 0.045 0 0.032 0.045 0.141 0.045 0.032 0.045 0.045 0.03]*0.5815

- **PosTag :**

الجريمه=['0.077 0.095 0.265 0.077 0.032 0.045 0.032 0.032 0 0.045] *0.3

الالكترونيه=['0.055 0.241 0.071 0.084 0.032 0 0 0.055 0.063 0.045 0] *0.3

سلوك=['0.071 0.077 0.084 0.1 0.032 0.032 0.032 0 0 0 0 0.063 0] *0.2

غير=['0 0.032 0.032 0.045 0.063 0.045 0.032 0.032 0.045 0.045] *0.2

قانوني =['0 0.045 0 0.032 0.045 0.141 0.045 0.032 0.045 0.045 0.03] *0.3

Le résultat est **la somme** de ces vecteurs de mots :

Résultat=vecteur (الجريمه* poids) +vecteur (الالكترونيه* poids) +vecteur (**سلوك** * poids) +vecteur (غير* poids) +vecteur (**قانوني** * poids) = Vecteur.

b) Caractéristiques des textes:

Parmi les caractéristiques des textes il existe de nombreuses techniques de similitudes syntaxique qui ont été détaillé dans la section **« Concepts fondamentaux liés à la similarité des textes court**s **»**. Nous avons utilisé les techniques suivantes:

- **Basée-terme :** Similarité **d'ordre**, **Dice,** Calcul **Matriciel.**
- **Basée-caractère :** Similarité de **Jaccard**, Distance de **Levenshtein**, Distance de **Jaro,** Long sous-chaine commune **(LCS)**, Similarité **STS**.

c) Caractéristiques d'extraction d'information:

- **Difficulté de Question** :

Il est calculé en fonction des notes, pour pouvoir déterminer la difficulté de la question **(Difficile- Moyenne- Facile)** tel que pour chaque question nous notons que:

Si la moyenne des notes de son ensemble des réponses est inférieure ou égale à 1.5 la question est considéré comme **difficile**, sinon si la moyenne est supérieure à 1.5 et inférieur à 3.5 la question est **moyenne**, sinon la question est **facile**.

1.5. Elaboration de fichier des caractéristiques :

Après avoir calculé toutes les caractéristiques extraites des réponses étudiants, réponses modèles et les questions, qui ont été sélectionnées dans la partie de conception, nous les enregistrons dans un fichier **Excel** afin de faciliter leurs utilisations comme données d'apprentissage automatique. D'où une ligne représente un enregistrement de données pour un seul étudiant.

La figure **« Figure 23 »** est un aperçu du fichier **Excel** des caractéristiques calculées:

STS_Sim	Dice_Sim	accard_Sir	Jaro_Sim	Ordre_Sim	LCS_Sim	we_Pos	Ve_Cosinu	e_TF_minn	Term_Sim	Chars_Sim	yntaxic_Sir	peQuestio	gthRespo	iff_Lengt	ondance_
0,329363	0,471429	0,76	0,621784	1	0,292308	0,912476	0,910316	0,910215	0,577143	0,420706	0,514568	1	74	135	0
0,70681	0,77707	0,916667	0,715685	1	0,584615	0,968416	0,963495	0,962489	0,821656	0,690822	0,769323	1	144	65	0
0,445398	0,449612	0,64	0,621742	1	0,292308	0,90246	0,90254	0,892949	0,55969	0,44691	0,514578	1	75	134	0
0,683794	0,75	0,913043	0,72999	1	0,546154	0,956665	0,954186	0,954097	0,8	0,671445	0,748578	1	116	93	0
0,517118	0,624204	0,84	0,724793	1	0,523077	0,949113	0,955142	0,947325	0,699363	0,595331	0,65775	1	176	33	0
0,474407	0,623377	0,913043	0,717265	1	0,415385	0,929248	0,925075	0,915532	0,698701	0,5501	0,639261	1	136	73	0
0,71833	0,783784	0,869565	0,769915	0,921875	0,6	0,964473	0,9702	0,968019	0,811402	0,704348	0,76858	1	141	68	0,095238
0,740712	0,795031	0,958333	0,72549	1	0,607692	0,96836	0,965714	0,966914	0,836025	0,71753	0,788627	1	137	72	0,095238
0,800874	0,835443	0,913043	0,773061	1	0,630769	0,976795	0,976799	0,977508	0,868354	0,749791	0,820929	1	139	70	0
0,602697	0,753927	0,92	0,763397	0,928571	0,508876	0,964974	0,964471	0,956464	0,788856	0,636862	0,728058	1	285	76	0,15
0,696137	0,686131	0,913043	0,73962	1	0,469231	0,957217	0,963433	0,963203	0,748905	0,650575	0,709573	1	101	108	0,133333
0,701716	0,770186	0,916667	0,795955	0,964286	0,623077	0,969038	0,968772	0,964189	0,809006	0,716949	0,772184	1	155	54	0
0,617186	0,671141	0,846154	0,706096	1	0,492308	0,965709	0,971267	0,96701	0,736913	0,616959	0,688931	1	123	86	0
0,349089	0,512821	0,769231	0,693862	0,777778	0,461538	0,936801	0,934107	0,937652	0,565812	0,500922	0,539856	1	141	68	0,26087
0,692659	0,812121	0,958333	0,806923	1	0,676923	0,963516	0,96823	0,958608	0,849697	0,7396	0,805658	1	171	38	0,16
0,511917	0,59854	0,869565	0,669051	0,833333	0,361538	0,928841	0,929052	0,922783	0,645499	0,530487	0,599494	1	109	100	0
0,534633	0,57971	0,76	0,674828	1	0,376923	0,943539	0,946245	0,940226	0,663768	0,531714	0,610946	1	101	108	0
0,37509	0,444444	0,72	0,621573	1	0,330769	0,905609	0,908814	0,916164	0,555556	0,445334	0,511467	1	105	104	0,25
0,652339	0,675862	0,956522	0,742538	1	0,469231	0,963488	0,96647	0,960876	0,74069	0,640989	0,700809	1	112	97	0
0,360168	0,525	0,846154	0,728948	1	0,446154	0,924293	0,932264	0,923191	0,62	0,514962	0,577985	1	155	54	0,090909

Figure 23: Aperçu du Fichier Excel des caractéristiques calculées

1.6. Processus de l'apprentissage automatique :

Le processus suivis dans la partie de l'apprentissage automatique du notre travail passe par l'étape de partitionnement de données, extraction de caractéristiques, les modèles utilisées et l'évaluation des modèles.

a) L'étape de **« partitionnement de données »** consiste à diviser nos ensembles de données en **80%** données d'entrainements et **20%** données de test

Le tableau suivant **« Tableau 6 »** montre le partitionnement de donnée en terme de nombre de couples :

Dataset	Couples d'entrainement	Couple de test	Total
AR-ASAG	1706	427	2133
Mohler	1953	489	2442

Tableau 6: Partitionnement des données

b) La seconde étape est l'étape de **« extraction des caractéristiques »** pour que nous puissions par la suite sélectionner la meilleure combinaison pour nos modèles guidés par l'entrainement.

Dans le Tableau **« Tableau 7 »,** nous montrons les différentes expériences de l'ensemble des caractéristiques utilisées dans notre système final qui nous aident à sélectionner les meilleures caractéristiques à partir des résultats de chaque expérience.

Combinaisons	Sim Syntaxique	Sim sémantique	ES	WE	Déviation-Question	ENTRAINEMENT
Combinaison 1	Oui	Oui	Oui	Oui	Oui	Avec Toutes les caractéristiques
Combinaison 2	Non	Oui	Oui	Oui	Oui	Sans similarité syntaxique
Combinaison 3	Oui	Non	Oui	Oui	Oui	Sans similarité sémantique
Combinaison 4	Oui	Oui	Non	Oui	Oui	Sans L'espace sémantique
Combinaison 5	Oui	Oui	Oui	Non	Oui	Sans Word Embedding
Combinaison 6	Oui	Oui	Oui	oui	Non	Sans Déviation-Question

Tableau 7: Différente combinaisons de caractéristiques pour l'entrainement

- **Le but de ces expériences consiste à étudier l'impact des différentes caractéristiques sur le score :**
  - La première expérience combine toutes les caractéristiques calculées précédemment pour obtenir un premier résultat que nous puissions par la suite le comparer avec les autres expériences.
  - Nous voulons étudier dans le reste des expériences l'impact de chaque ensemble de caractéristique à part pour voir l'effet de ces caractéristiques sur les résultats.
  - Grâce à ce que nous avons fait à partir d'expériences précédentes, nous avons pu extraire les caractéristiques appropriées qui nous ont donné les meilleurs résultats dans tous les modèles que nous avons entrainés.

c) Afin d'atteindre le modèle final ayant le meilleur résultat, nous mettons en œuvre de nombreux algorithmes, parmi ces modèles nous avons testé : **LR, SVR, DTR, Mlpregressor(c'est le modèle ANN)** de la bibliothèque **«Scikit-learn»**(

https://scikit-learn.org/) afin de prédire une note automatiquement en utilisant la propriété **«Average_Mark »** où nous en dépendions comme sortie de notre modèle dans la partie **entrainement**, tandis que pour la classification nous optons pour les modèles **LogR, SVC, DTC** pour classer la note selon une classe d'échelle tel que nous testons pour deux ensembles de classe , le premier ensemble **« classenote1 »**est constitué de 5 classes de notes ayant l'échelle **[0-1,1-2,2-3,3-4,4-5]** , le deuxième ensemble**« classenote2 »** a 10 classes de notes ayant l'échelle **[0-0.5, 0.5-1,1-1.5,1.5-2,2-2.5,2.5-3,3-3.5,3.5-4,4-4.5,4.5-5].**

d) La dernière étape est destinée non seulement pour la validation du modèle (pour avoir le modèle le plus performant) mais aussi pour le choix de la combinaison des caractéristiques qui nous mène à identifier le modèle final.

Le but de notre évaluation est d'avoir une précision élevée avec une petite marge d'erreur cette dernière détermine la différence entre les notes prédites et les notes manuelles données par les enseignants. Pour le faire plusieurs métriques d'évaluations sont utilisées nous citons : **RMSE, Recall, Accuarcy, Corrélation de Pearson.**

1.7. Environnement du développement :

En raison du volume de travail fourni et des ressources utilisées, le développement avec nos machines personnelles était pour nous comme un obstacle, car la RAM de 8 Go ne supporte pas le gigantesque de l'espace sémantique de 30 000 mots et le Word embeddings de 6 millions mots, ces difficultés ont été relativement surmontées, à l'aide du serveur fourni par **Google**, appelé **Colab**, qui fournit 35 Go du RAM. Les calculs de la consommation des ressources matérielles sont présentés dans la section **« Résultats et discussions »**

L'environnement de développement utilisé dans les machines personnelles, c'est **« Pycharm »** utilisant le langage **Python.**

Concernant le domaine de l'apprentissage automatique, nous nous sommes appuyés sur la bibliothèque **« SK-learn »** (https://scikit-learn.org/) avec ses divers algorithmes et outils d'évaluation.

Côte Statistique, les graphes présentés sont construits par la bibliothèque **« Matplotlib »** (https://matplotlib.org)

Les résultats de la partie **implémentation** sont discuté dans la section « **Résultats et discussions** » :

2. Résultats et discussions :

Avant d'obtenir un résultat final et le plus efficace, nous avons pris de nombreuses observations en faisant plusieurs expériences pour étudier l'impact de la présence et de l'absence de certaines techniques et caractéristiques.

Les observations les plus marquantes se reposent sur l'étendue de l'effet de Stemming sur le calcul des similarités, la taille de l'espace sémantique et les caractéristiques d'entrée du modèle d'apprentissage automatique qui ont conduit au meilleur résultat et les résultats obtenus pour chaque type de question ainsi que les résultats des ressources matérielles consommées.

2.1 Impact de la notion de Stem sur les calculs des similarités :

Nous voulons dans cette partie étudier l'impact de l'utilisation de la notion du Stem dans le calcul des similarités surtout que les similarités constituent les caractéristiques les plus importantes dans nos modèles d'entrainement. Dans le **tableau 8**, nous présentons les résultats de similarités syntaxiques prises indépendamment en variant le stem.

Nous constatons que les similarités basées chaines sont très sensibles, un caractère de plus, peut rendre la similarité moins exacte, exemple

يعاقب القانون مرتكب الجريمة الالكترونية

Et

يعاقب القانون مرتكب الجرائم الالكترونية

Or que le stem lourd va générer la racine des mots d'où الجرائم et الجريمة ont le même mot racine جرم

Résultats des Calculs de Similarité syntaxique			
Techniques de similitudes syntaxique	**Stem Utilisée**	**Corrélation**	**RMSE**
Similarité d'ordre	Tashpyhne lourd	6.13%	2.27
	Sans Stem	14.36%	2.33
DICE	Tashpyhne lourd	60.50%	1.19
	Sans Stem	60.75%	1.16
Jaccard	Tashpyhne lourd	50.18%	1.21
	Sans Stem	47.73%	1.38
JARO	Tashpyhne lourd	53.77%	1.25
	Sans Stem	53.42%	1.30
LEVENSHTEIN	Tashpyhne lourd	49.23%	1.71
	Sans Stem	46.14%	1.72
LCS	Tashpyhne lourd	53.11%	1.53
	Sans Stem	51.18%	1.49
STS	Tashpyhne lourd	53.09%	1.55
	Sans Stem	52.10%	1.75

Tableau 8: Résultats des Calculs de Similarité syntaxique (Dataset AR-ASAG)

Dans le **tableau 9**, nous présentons les résultats de similarités sémantique prises indépendamment en variant le stem.

Résultats des Calculs de Similarité Sémantique			
Techniques de similitudes	**Stem Utilisée**	**Corrélation**	**RMSE**
Espace Sémantique			
TF-MINMAX	Tashpyhne léger	43%	1,77
	Sans Stem	33,88%	2,17
Calcul Matriciel	Tashpyhne léger	49,24%	1,92
	Sans Stem	45,89%	2,14
POS	Tashpyhne léger	41.50%	1.81
	Sans Stem	32,96%	2.19
COSINUS	Tashpyhne léger	43,99%	1,75
	Sans Stem	33,13%	2,17
Incorporation des mots(Word Embedding)			
COSINUS	Sans Stem	61,95%	1.85
	Tashpyhne léger	59.28%	1.79
POS	Sans Stem	60,34%	1.84
	Tashpyhne léger	58.46%	1.76
MINMAX	Sans Stem	62,05%	1.81
	Tashpyhne léger	56.43%	1.66

Tableau 9: Résultats des Calculs de Similarité Sémantique

Les résultats obtenus avec l'approches des WE sont bons sans la notion de stem, cela est dû à l'ensemble de mots générés par un domaine général sans Stem. Par contre le stem léger est imposé pour l'espace sémantique car il est généré à partir d'un corpus spécifique prétraité et stemmé par le stem léger pour le but d'enlever les suffixes et préfixes qui causent à la non identification des mêmes mots, par exemple: جريمة et الجريمة sont considérés comme deux mots différents.

La partie suivante est dédié pour les différents résultats d'apprentissage automatique ainsi que leurs interprétations.

2.2 Résultats des modèles d'apprentissage automatique:

Après un certain nombre de tentatives et d'ajustement pour le choix du modèle et des caractéristiques, le modèle final a été défini ainsi que les propriétés qui donnent le meilleur résultat.

Nous entamons la section des résultats obtenus en appliquant l'apprentissage automatique, où nous montrons les modèles utilisés, le meilleur-modèle, les caractéristiques ayant les plus d'impacts sur les résultats et une analyse approfondie du meilleur model obtenu.

Les caractéristiques les plus adéquates pour tous les modèles sont :

- Caractéristiques des textes : STS, Jaccard, Dice, Jaro, Longueur de réponse, Fréquences de redondance, Différence de longueur
- Caractéristiques du domaine Général et Spécifique : Calcul matriciel (WE), Cosinus Similarité (WE et ES),
- Pondération des termes :TF- Minimax (WE et ES), PosTag (WE et ES)
- Déviation-Question:
  - Difficulté de Question
  - Pondération par l'écart d'information(WE)=Cosinus (Gap*RE, RM) tel que Gap=RE-RM et Gap=RM*RE Pour les vecteurs de phrase obtenus par WE.
  - Informations exprimées et attendues(ES)= Cosinus similarité entre (RM *Q, RE*Q) Pour les vecteurs de phrase obtenus par l'ES.
  - Écart d'informations entre RE et RM =Cosine (RE-Gap, RM-Gap) et Cosine (RE*Gap, RM*Gap) tel que Gap=RE-RM et Gap =RM*RE Pour les vecteurs de phrase obtenus par WE.

Tel que la validité de ces caractéristiques est dû à une étude d'ablation exprimée dans la section « **2.5 »**. Le tableau **« Tableau 10 »** présente les résultats de ces caractéristiques sur les différents algorithmes d'apprentissage automatique sur le **« Dataset Arabe »**

d’où nous remarquons que le **"score de test" calculé par « Sk-learn »** dans les modèles de classification **est égal** à **« Accuarcy » et « Recall » :**

Régression				
Types de modèle	**Scores Entrainement**	**Scores Test**	**Corrélation**	**RMSE**
LR	**59,25%**	**60,18%**	**77,95%**	**0,8967**
SVR	53,88%	57,25%	75,70%	0,92
DTR	57,59%	46,74%	68,56%	1,03
Mlpregressor	54,45%	56,97%	76,08%	0,92
Classification				
Types de modèle	**Scores Entrainement**	**Scores Test**	**Corrélation**	**RMSE**
LogR	56,09%	44,49%	66,24%	1,10
SVC	49,35%	42,38%	68,76%	1,04
DTC	49,06%	42,15%	62,60%	1,18

Tableau 10: Résultats des modèles d’apprentissage automatique

D’après le tableau **« Tableau 10 »** notre meilleur modèle est « **Ridge Linear Regression** » de la bibliothèque **« SK-learn »,** en utilisant la méthode **« PolynomialFeatures »** de degré deux qui sert à générer des caractéristiques polynomiales et d'interaction, tel que ce processus se fait par la génération d’une matrice des caractéristiques de toutes les combinaisons polynomiales à partir de notre fichier des propriétés dont le degré est inférieur ou égal au degré spécifié (dans notre cas c’est deux).

Exemple : si un échantillon d'entrée est bidimensionnel et de la forme [a, b], les entités polynomiales de degré 2 sont [1, a, b, a^2, ab, b^2].

Nous remarquons aussi que le modèle **Mlpregressor** nous a donné des résultats satisfiables comparant avec le modèle **LR**. C'est un modèle des réseaux de neurone simple, nous entrainons ce modèle avec : une fonction d'activation égale à **'identity'** pour le calque caché cela signifie une activation utile pour implémenter un modèle linéaire, un terme de régularisation = 0.001, taux d'apprentissage '**constant'** = 0.001 et un optimiseur '**lbfgs'** qui est de la famille des méthodes quasi-Newton.

Dans le **Tableau 11** nous présentons une comparaison du notre meilleur modèle obtenu avec une approche non supervisée (approche de (F. R. OUKINA et al.,2019)).

Résultat du meilleur Modèle				
Type du modèle	**Score entrainement**	**Score Test**	**RMSE**	**Corrélation**
Ridge Linear Regression	**59.25%**	**60.81%**	**0.89**	**77.95%**
Ancien modèle 2019 Approche non supervisé(F. R. OUKINA et al.,2019)	-	-	1.05	68,38

Tableau 11: Présentation du meilleur modèle « Dataset Arabe AR-ASAG »

Comparé aux résultats obtenus dans un travail précédent (F. R. OUKINA et al.,2019) qui utilise une approche non supervisée combinée à des similarités sémantiques basées corpus nous constatons une amélioration de +9.57% en corrélation de Pearson et une nette amélioration de l'ordre de +0.16 sur la dispersion de l'erreur quadratique. Ce qui confirme bien l'impact d'une approche supervisée dans l'amélioration de la précision.

- **Analyse approfondie sur les écarts entre scores manuels et scores automatique :** Pour une interprétation détaillée sur les résultats du meilleur modèle nous fournissons **le Tableau 12** qui présente les écarts entre les notes manuelles et automatiques sur une échelle de 10 (0-0.5, 0.5-1, …, 4 - 4,5, 4.5-5). Cette analyse est faite pour un jeu de test composé de **427 couples.**

Différence score auto/ score manuel	Nombre de couples	Pourcentage
Zéro (scores égaux)	1	0.234192037470726 %
Inférieur ou égale à 0.5 (et supérieur à 0)	175	40.98360655737705 %
Inférieur ou égale à 1 (et supérieur à 0.5)	136	31.850117096018735 %
Inférieur ou égale à 1.5 (et supérieur à 1)	79	18.501170960187352 %
Inférieur ou égale à 2(et supérieure à 1.5)	21	4.918032786885246 %
Inférieur ou égale à 2.5(et supérieure à 2)	12	2.810304449648712 %
Inférieur ou égale à 3(et supérieure à 2.5)	3	0.702576112412178 %
supérieure à 3	0	0%

Tableau 12: Résultats statistiques Manuel-Auto

Le tableau montre que dans 91,56 % des cas, l'écart est inférieur ou égal à 1,5 sur une échelle de notation de 5 points. Ce qui est raisonnable moyennant la subjectivité déjà humaine dans le processus de notation. Dans 73,06% des cas, la différence est inférieure ou égale à 1. Toutefois, dans 8,43% des cas, la différence dépasse 1.5. Aucune différence supérieure à 3 n'est enregistrée.

Le travail portera dans l'avenir à diminuer au maximum ce pourcentage de 8,43% sur les écarts.

Pour une visualisation clarifiée de cette analyse nous mettons la **Figure 24** qui est une illustration qui représente des scores manuels et automatiques pour un échantillon de 20 étudiants :

Vert = Représente les notes manuelles

Rouge =Représente les notes automatiques

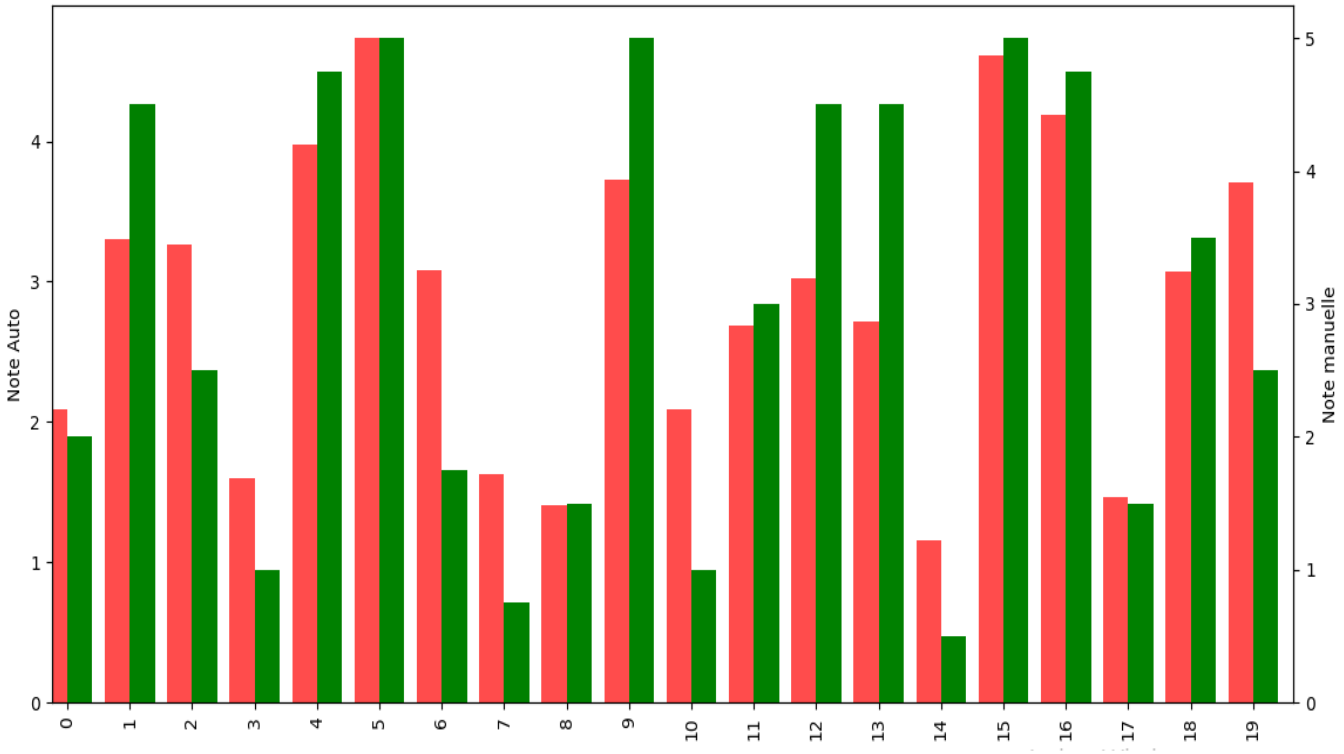


Figure 24:échantillon de la différence entre les notes manuelle et automatique « Dataset-Arabe »

La figure 24 montre l'écart entre la note manuelle et la note automatique prédite obtenue par la phase de test de l'apprentissage automatique, où nous observons une convergence des résultats entre la notation manuelle et automatique et cela nous donne une vision clarifiée des résultats de l'analyse approfondie faite dans le tableau ci-dessus **« tableau 13 »** où nous voyons bien que la différence entre les notes est très proche.

2.3 Résultat statistique du meilleur modèle par type de question :

Pour une analyse plus précise des résultats obtenus, nous faisons une étude statistique sur les résultats de chaque type de question et ceci afin d'extraire le type de question qui contribue à l'efficacité du modèle et le type de question dans lequel les résultats sont moins efficaces, sachant que la collection de questions de notre ensemble de données contient 5 types de questions.

Les résultats statistiques par type de question sont présentés dans le tableau **« tableau 13 »** ces résultats sont présentés dans les tableaux ci-dessous :

Résultat statistique par type de question				
Type de question	**Score entrainement**	**Score Test**	**RMSE**	**Corrélation**
عرف	59,25%	42,79%	0.96	67,15%
إشرح	59,25%	65,44%	0.86	81,60%
ما النتائج	59,25%	74,29%	0.71	87,09%
علل	59,25%	55,61%	0.98	75,78%
ما الفرق بين	59,25%	14,58%	0.9	44,14%

Tableau 13: Résultat statistique par type de question

La Figure **« Figure 25 »** Représente un histogramme des RMSE pour chaque type de question

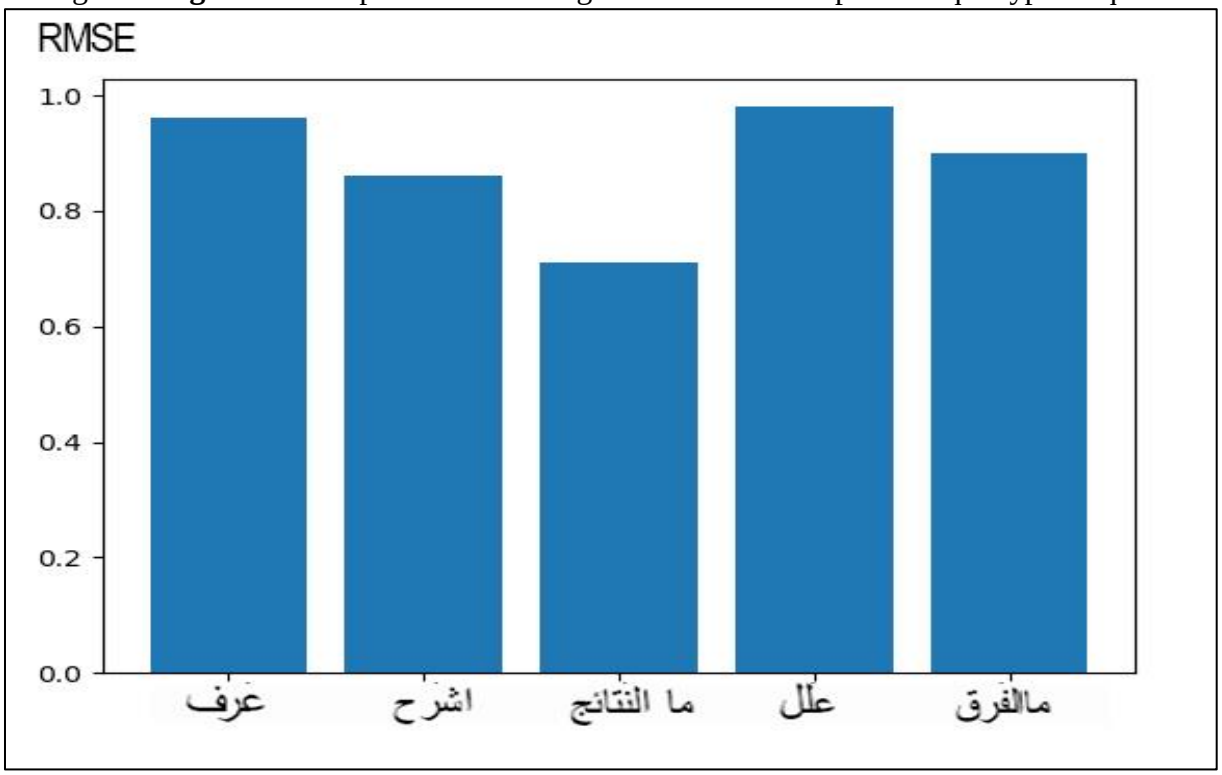


Figure 25: Histogramme des RMSE par type de question

Analyse approfondie du type **«عرف »** pour 57 réponses est représenté dans le tableau **« tableau 14 »** :

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	0	0%
Inférieur ou égale à 1	40	70.17543859649123 %
Inférieur ou égale à 2 (et supérieur à 1)	13	22.80701754385965 %
Inférieur ou égale à 3(et supérieure à 2)	4	7.017543859649122 %
Supérieur à 3	0	0%

Tableau 14: Analyse approfondie du type « عرف »

Analyse approfondie du type **«اشرح »** pour 172 réponses est représenté dans le tableau **« tableau 15 »** :

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	1	0.5813953488372093 %
Inférieur ou égale à 1	133	77.32558139534883 %
Inférieur ou égale à 2 (et supérieur à 1)	32	18.6046511627907 %
Inférieur ou égale à 3(et supérieure à 2)	6	3.488372093023256 %

Supérieur à 3	0	0%

Tableau 15:Analyse approfondie du type «اشرح »

Analyse approfondie du type **«ما النتائج »** pour 49 réponses est représenté dans le tableau **« tableau 16 »** :

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	0	0%
Inférieur ou égale à 1	40	81.63265306122449 %
Inférieur ou égale à 2 (et supérieur à 1)	9	18.367346938775512 %
Inférieur ou égale à 3(et supérieure à 2)	0	0%
Supérieur à 3	0	0%

Tableau 16: Analyse approfondie du type « ما النتائج »

Analyse approfondie du type «علل » pour 101 réponses **est représenté dans le tableau « tableau 17 » :**

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	0	0%
Inférieur ou égale à 1	63	62.37623762376238 %
Inférieur ou égale à 2 (et supérieur à 1)	35	34.65346534653465 %

Inférieur ou égale à 3(et supérieure à 2)	3	2.9702970297029703 %
Supérieur à 3	0	0%

Tableau 17: Analyse approfondie du type «علل»

Analyse approfondie du type **«ما الفرق »** pour 48 réponses est représenté dans le tableau **« tableau 18 »** :

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	0	0%
Inférieur ou égale à 1	35	72.91666666666667 %
Inférieur ou égale à 2 (et supérieur à 1)	11	22.916666666666668 %
Inférieur ou égale à 3(et supérieure à 2)	2	4.166666666666667 %
Supérieur à 3	0	0%

Tableau 18: Analyse approfondie du type «ما الفرق»

D’après les tableaux présentés ci-dessous nous constatons que le type de question « ما الفرق » a des résultats faibles par contre le type « ما النتائج » suivis de « إشرح » ont des résultats très significatifs. Ceci peut être expliqué par le fait que la question «ما الفرق » est plus vague dans l’expression de la réponse. La différence de critères de comparaison entre l’enseignant et l’étudiant peut impliquer un écart dans les deux réponses. Ceci indique qu’il faut insister sur la meilleure façon de poser une question ouverte qui doit directement cibler des concepts précis.

2.4 Résultats du meilleur modèle sur les différentes tailles de l’espace sémantique :

Le nombre de mots utilisés dans l'espace sémantique affecte directement les résultats obtenus, plus le nombre de mots formés est élevé, plus les résultats seront précis, pour

étudier la validité de cette proposition, nous avons calculé les résultats du modèle principal en utilisant trois espaces sémantiques ayant la taille 9000 mots, 20 000 mots, 30 000 mots (respectivement). Rappelons que l'espace sémantique de 9000 mots a été repris d'un travail précédent (F. R. OUKINA et al.,2019) et que celui de 20000 mots a été généré par une plateforme de gestion de corpus (A., Ben Hamida,2020) un travail qui se déroule en parallèle au notre. Les résultats sont présentés dans le **Tableau 19**:

Résultats du meilleur modèle avec les différentes taille de l'espace sémantique			
Notre Espace Sémantique (30K mots)			
Score entrainement	**Score Test**	**RMSE**	**Corrélation**
59.25%	**60.81%**	**0.89**	**77.95%**
Espace sémantique (A., Ben Hamida,2020) (20K mots)			
58.70%	58.98%	0.91	77.10%
Espace sémantique (F. R. OUKINA et al.,2019) (9K mots)			
58.82%	58.74%	0.91	76.94%

Tableau 19: Comparaison des résultats sur différents espaces sémantiques

Les résultats obtenus favorisent l'espace sémantique ayant la plus grande taille, ce qui est logique puisque la taille de l'espace sémantique signifie le nombre de mots et cela conduit à de meilleurs résultats. De là, nous remarquons que plus les mots de l'ES augmente, plus le calcul sera précis, ce qui signifie que l'augmentation est proportionnelle. Cela est dû au fait que la richesse linguistique joue un rôle important dans l'amélioration des résultats

2.5 Validité et performance des caractéristiques choisies (Etude d'ablation) :

La conception et l'ingénierie de notre système, nous l'avons construit sur plusieurs points principaux (le domaine spécifique/général, les pondérations des termes, l'extraction d'information). Pour étudier la force et l'efficacité de ces points, nous avons calculé les résultats du modèle finals en présence et en l'absence de ses caractéristiques. Cette étude d'ablation nous permet de confirmer l'importance des caractéristiques retenues par rapport

au modèle final. **Le Tableau 20** présent les résultats obtenus qui confirment bien l'importance des caractéristiques retenues.

Tableau 20 :Représentations des différents résultats d'étude d'ablation

Résultats d'étude d'ablation				
Types de modèle	**Scores Entrainement**	**Scores Test**	**Corrélation**	**RMSE**
LR(le modèle final)	**59,25%**	**60,18%**	**77,95%**	**0,8967**
Sans l'espace sémantique	57,44%	58,29%	76,79%	0.91
Sans les Word embeddings	55,87	57,28%	75 ,95%	0.92
Sans Pondérations des termes	58,47%	58,94%	77,16%	0.91
Sans déviation question	57,77%	59,07%	77,24%	0.9

Pour extraire l'effet de la présence des mots qui composent la question dans la réponse de l'étudiant, nous avons recalculé toutes les caractéristiques précédentes en supprimant tous les mots communs entre la question et la réponse **(Question Demoting),** nous constatons que les résultats étaient faibles par rapport aux résultats précédents. Ce qui justifie le fait d'avoir écarté le « Question-demoting » des caractéristiques du modèle final.

2.6 Application du modèle avec la langue anglaise :

Pour mesurer l'efficacité de notre système et son indépendance à la langue arabe, il a été testé à l'aide d'un ensemble de données en langue anglaise, où le meilleur résultat a été atteint grâce aux caractéristiques suivantes :

- Fonctionnalités des textes : STS, Jaccard, Dice, Jaro, Longueur de réponse, Fréquences de redondance, Différence de longueur
- Fonctionnalité du domaine Général et Spécifique : Calcul matriciel (WE), Cosinus Similarité (WE et ES),
- Pondération des termes :TF- Minimax (WE), PosTag (WE et ES)
- Déviation-Question :

➢ Difficulté de Question,

- Pondération par l'écart d'information =Cosinus (Gap*RE, RM) tel que Gap=RE-RM et Gap=RM*RE. Pour les vecteurs de phrase obtenus par WE et ES.
- Informations exprimées et attendues = Cosinus (RM *Q, RE*Q) et Cosinus (RM-Q, RE-Q) pour les vecteurs de phrase obtenus par WE et ES.
- Écart d'informations entre RE et RM =Cosine (RE-Gap, RM-Gap) et Cosinus (RE*Gap, RM*Gap) tel que Gap=RE-RM et Gap =RM*RE pour les vecteurs de phrase obtenus par WE et ES.

Ces caractéristiques sont très proches des caractéristiques de la langue arabe modèle, et cela confirme la validité de la méthodologie utilisée et les caractéristiques sélectionnées, le **Tableau 21** montre le résultat obtenu :

Résultat du meilleur modèle avec Mohler Dataset			
Score entrainement	**Score Test**	**RMSE**	**Corrélation**
49. 25%	**41,80%**	**0.83**	**65,04%**

Tableau 21: résultat du meilleur modèle pour Mohler-dataset

Pour une interprétation détaillée sur les résultats du meilleur modèle nous fournissons le **Tableau 22** qui représente, les résultats statistiques pour 10 classes d'échelles.

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	62	12.67%
Inférieur ou égale à 0.5 (et supérieur à 0)	191	39.05%
Inférieur ou égale à 1 (et supérieur à 0.5)	140	28.62%
Inférieur ou égale à 1.5 (et supérieur à 1)	52	10.63%
Inférieur ou égale à 2(et supérieure à 1.5)	31	6.33%
Inférieur ou égale à 2.5(et supérieure à 2)	11	2.24%
Inférieur ou égale à 3(et supérieure à 2.5)	2	0.408%
supérieure à 3	0	0%

Tableau 22:Résultats statistiques pour Mohler-dataset pour 10 classes d'échelles

Le tableau montre que dans 91% des cas, l'écart est inférieur ou égal à 1,5 sur une échelle de notation de 5 points. Ce qui est raisonnable moyennant la subjectivité déjà humaine dans le

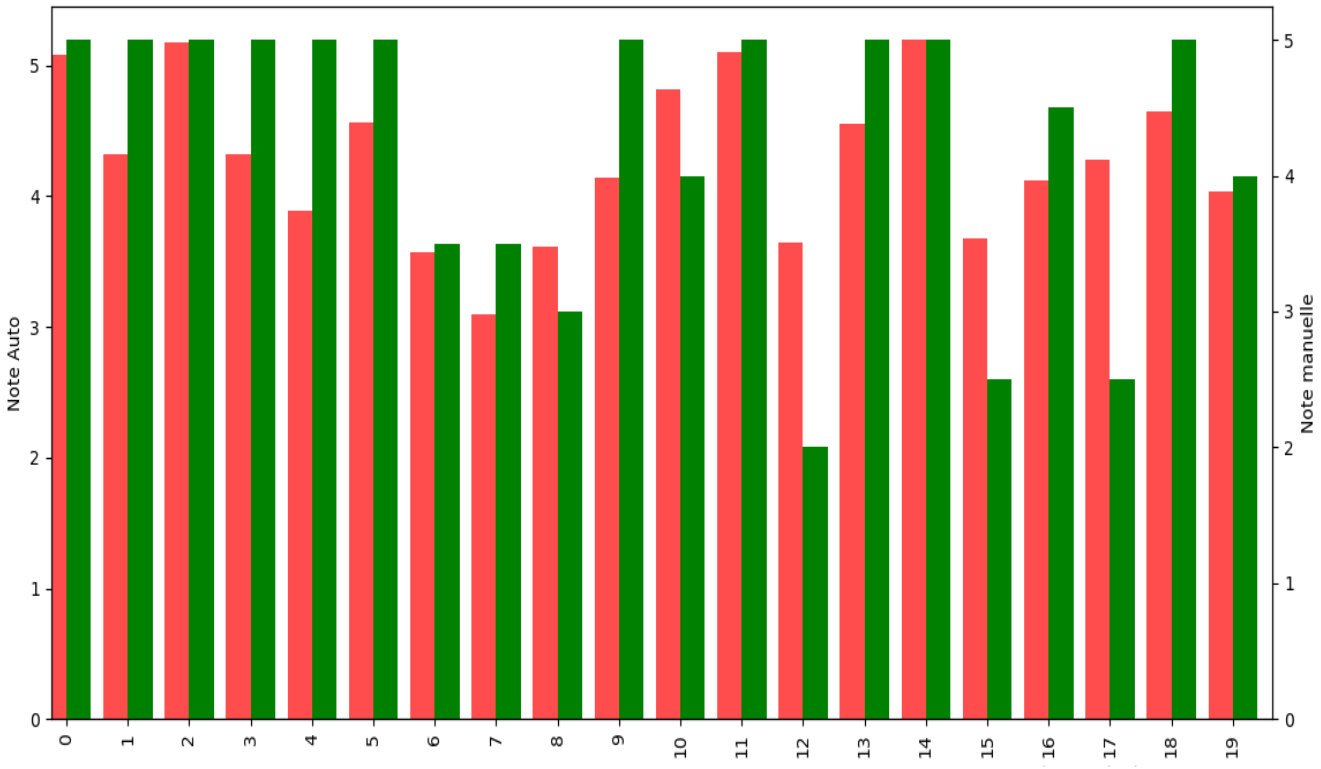
processus de notation. Dans 80% des cas, la différence est inférieure ou égale à 1. Toutefois, dans 9% des cas, la différence dépasse 1.5. Aucune différence supérieure à 3 n'est enregistrée.

Pour une visualisation clarifiée de résultat obtenu, nous mettons la **Figure 26** qui est une illustration qui représente des scores manuels et automatiques pour un échantillon de 20 étudiants :

Vert= Représente les notes manuelles

Rouge =Représente les notes automatiques.

Figure 26:échantillon de la différence entre les notes manuelle et automatique Mohler



La figure 26 nous donne une vision clarifiée des résultats statistique de l'analyse approfondie faite dans le tableau ci-dessus **« tableau 22 »** où nous voyons bien que la différence entre les notes manuelle et automatique prédite par le système est très proche. Cela implique la validité des résultats de notre analyse approfondie.

2.7 Discussion des résultats avec Mohler dataset :

Les résultats obtenus en langue anglaise sont des résultats encourageants par rapport à l'ensemble des systèmes qui s'est appuyé sur le même ensemble de données Mohler, comme il est présenté dans le **tableau 23.** Rappelons que les résultats de **(**Ramachandran 2015) ont

été extraits directement de son papier bien qu'il n'ait pas été inclus dans notre analyse de l'état de l'art.

Comparaison du notre modèle avec les autres systèmes avec le Dataset Mohler		
Systèmes développés	**RMSE**	**Corrélation**
Notre Modèle	**0.83**	**65,04%**
Mohler 2011	0.97	51.80%
Sultan 2016	0.85	63%
Ramachandran 2015	0.86	61%
Swarnadeep Saha 2018	0.9	57%

Tableau 23:Comparaison du notre modèle avec les autres systèmes avec le dataset Mohler

2.8 Résultats obtenus après intégration de plusieurs réponses modèles :

En parallèle avec notre travail et dans le cadre du même projet, se déroule le travail de (O. HAMEL et al.,2020) qui consiste à générer automatiquement, à partir d'une réponse modèle de l'enseignant, plusieurs réponses modèles de même sens mais avec des formulations différentes. L'idée ici est de comparer la réponse de l'étudiant non pas à une seule réponse modèle mais à plusieurs et en garder le meilleur score. Pour améliorer la performance de notre outil d'évaluation et valider leur travail nous avons intégré la génération des paraphrases à notre outil. Nous avons obtenu les résultats suivants pour le Dataset Arabe illustrés dans le tableau **24**

Métriques d'évaluation	**Sans Paraphrases (Notre système Initial)**	**Avec paraphrases (Notre système avec intégration du Générateur de paraphrases)**
Corrélation	0,7794	**0,8892**
RMSE	0,8968	**0,6955**

Tableau 24: Résultats obtenus après intégration de plusieurs réponses modèles

Nous relevons le taux de corrélation selon le coefficient de Pearson, ainsi que le taux de dispersion calculé par RMSE pour notre modèle avec régression de [r=0,7794 & RMSE = 0.8968] avant intégration et [r=**0,8892** & RMSE = **0,6955**]. L'amélioration dépasse même la corrélation humaine et l'erreur humaine enregistrée pour le dataset arabe lui-même [r=**0,84** & RMSE = **0,83**]. Le tableau **25** représente l'analyse approfondie pour ce résultat de combinaison pour une échelle de 10 classes :

Différence score auto/ score manuel	**Nombre de couples**	**Pourcentage**
Zéro (scores égaux)	2	0.46 %
Inférieur ou égale à 0.5 (et supérieur à 0)	249	58.31%
Inférieur ou égale à 1 (et supérieur à 0.5)	105	24.59 %
Inférieur ou égale à 1.5 (et supérieur à 1)	54	12.64%
Inférieur ou égale à 2(et supérieure à 1.5)	13	3.04 %
Inférieur ou égale à 2.5(et supérieure à 2)	4	0.93 %
Inférieur ou égale à 3(et supérieure à 2.5)	0	0%
supérieure à 3	0	0%

Tableau 25:Résultats statistiques après intégration de plusieurs réponses modèles

D'après ce tableau nous observons une amélioration par rapport aux résultats mentionnés dans **le tableau 12** avec une augmentation de 4,46% sur les cas ayant l'écart inférieur ou égal à 1,5 sur une échelle de notation de 5 points, pour les cas ayant la différence inférieure ou égale à 1, ils sont augmentés avec une pourcentage de 10,30%, le résultat le plus important c'est la diminution des cas ayant l'écart qui dépasse 1.5 avec une marge de 4%. Remarquons aussi qu'aucun écart supérieur à 2.5 n'est enregistré.

2.9 Consommations des ressources matérielles (Etude de performance) :

Afin d'estimer les ressources consommées et le temps nécessaire pris de l'exécution de notre système final pour la prédiction des notes pour les nouvelles collections de données, nous calculons le temps d'exécution pour plusieurs nombres de tests 50, 100,200, 300 et

427 couples de tests afin d'évaluer l'efficacité de notre système en matière de temps et ressources consommées durant la phase de l'exécution et de son intégration pratique dans un environnement de E-learning.

Le tableaux **(Tableau 26)** montre le temps d'exécution, la consommation de la RAM et la consommation du processeur pour les calculs de l'espace sémantique.

Informations concernant les calculs de similarité de l'ES			
Nombre de couples	**Temps d'exécution**	**CPU usage**	**RAM usage**
50 couples	11 min	7.7 %	7.5 %
100 couples	15 min	32.1 %	6.5 %
200 couples	30 min	36.3 %	6.9 %
300 couples	41 min	32.1 %	6.5 %
427 couples	49 min	40.6 %	7.1 %
Ram Total= 35 Go / Disque=107 Go / sur Google Colab			

Tableau 26: La consommation des ressources pour l'utilisation de l'ES

Le tableaux **(Tableau 27)** représente le temps d'exécution, la consommation de la RAM et la Consommation du processeur pour les calculs des Word embeddings

Informations concernant les calculs de similarité du WE			
Nombre de couples	**Temps d'exécution**	**CPU usage**	**RAM usage**
50 couples	30secondes	6.9 %	73.4 %= 20GO
100 couples	41 secondes	13.2 %	73.4 %=20GO
200 couples	1min 23Secondes	12.3 %	73.4 %=20GO
300 couples	2 min et 2 secondes	18.1 %	73.4 %=20GO
427 couples	2 min et 57 secondes	20.9 %	73.4 %=20GO
Ram Total= 25,5 Go / Disque=107 Go / sur Google Colab **Temps d'exécution du load(model de zahran + StanfordCoreNlp) == 12min et 27 secondes**			

Tableau 27: La consommation des ressources pour l'utilisation du WE

En raison du volume de données liées aux calculs, que ce soit en utilisant l'ES ou les WE, nous remarquons à travers les tableaux présentés ci-dessus que le temps passé lors du calcul de similarité en utilisant l'es est de 49 minutes et environ 3 minutes consacrées au calcul de la similitude avec WE pour 427 couples de données. Nous pouvons dire que ce temps consommé lors de ces calculs est logique vu le volume important de la quantité de données traitées. Nous en concluons que ces résultats sont quelque chose de motivant, en particulier pour l'étude de l'apprentissage automatique dans le domaine de l'éducation.

3. Conclusion :

Après l'ensemble des études qui nous a menés à l'implémentation finale, nous avons pu déterminer le bon modèle avec l'ensemble des caractéristiques affectant sur la notation.

Notre étude a prouvé sa capacité à s'adapter à la langue anglaise, et cela à travers les résultats que nous avons présentés précédemment, qui démontrent la supériorité de notre modèle sur les systèmes déjà développés et évalués sur le même dataset.

Notre travail va être implémenté dans un plugin sur la plateforme Moodle en utilisant le travail de (E. H. SNOUSSI et al.,2019). Ainsi notre meilleur modèle va être intégré au moteur de question de Moodle comme nouveau type de question: ShortAnswerLR. Deux extensions seront rajoutées : ShortAnswerLR-Ar (pour l'arabe) et ShortAnswerLR -En

Chapitre 5 : Conclusion Générale

Le domaine de l'évaluation automatique est un large domaine qui nécessite plusieurs ressources et plusieurs techniques avec une compréhension correcte et une bonne ingénierie.

La difficulté de traiter la langue, en particulier les langues qui se distinguent par la richesse de leur banque linguistique, comme la langue arabe, pour cela nous avons concentré dans cette étude principalement sur la langue arabe, en essayant d'enrichir les travaux. A cet égard, nous nous sommes attachés à profiter du grand développement de l'apprentissage automatique et de son intégration dans le domaine de l'éducation pour l'explorer dans notre travail.

Nous avons pensé à s'appuyer sur l'espace sémantique formé à partir de corpus du domaine spécifique qui contribue grandement aux résultats des comparaisons entre les textes sans abandonner le domaine général pour créer l'équilibre et l'harmonie autour de la distribution sémantique des mots pour la bonne prédiction de leur validité.

Nous nous sommes également appuyés sur la similitude textuelle directe entre les textes en comparant les chaines de caractères à travers leur composition littérale. La présence des mots-clés a à un grand impact sur la validité de la réponse d'étudiant, cela est traduit par la propriété de donner du poids aux termes. Comme nous avons conçu l'idée d'extraire des informations en coordonnant entre la réponse de l'élève et la question, ainsi qu'entre la réponse-modèle et la question.

L'efficacité du système final a été étudiée sur deux ensembles de données, l'un pour la langue arabe et l'autre pour la langue anglaise, les résultats ont montré l'efficacité du système par rapport aux systèmes précédents qui reposaient sur le même ensemble de données.

Grâce à notre modèle final, nous avons atteint ce qui est requis, à savoir l'évaluation automatique des réponses des étudiants, et cela est unifié pour les deux langues différentes, qui sont l'arabe et l'anglais.

Le système final est accessible et prêt pour l'intégration dans n'importe quelle plate-forme numérique et le tester pour l'Arabe et l'anglais.

En perspective nous pensons qu'il serait intéressant d'explorer encore les techniques de l'apprentissage profond (le Deep Learning) qui deviennent promoteurs dans le cadre de l'éducation.

References:

Arora aman. (2019). Quadratic Kappa Metric explained in 5 simple steps | Kaggle. https://www.kaggle.com/aroraaman/quadratic-kappa-metric-explained-in-5-simple-steps

A. ABDALLAH, K. GAROUDJA, « Mesures de similarité syntaxique pour un système d'évaluation automatique des réponses courtes : Application à la langue arabe ». Mémoire master USDB 1. 2017/2018.

Atif, J. (2016). Data Mining / ML e.

A., Ben Hamida, « Vers une plateforme de gestion des corpus et d'analyse de texte en langue arabe. », Mémoire de Master TAL, USDB1. 2019/2020.

Avinash Navlani. (2018, December 28). Decision Tree Classification in Python - DataCamp. https://www.datacamp.com/community/tutorials/decision-tree-classification-python

Ayush Pant. (2019, January 22). Introduction to Logistic Regression - Towards Data Science. https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148

Borovcnik, Manfred, Hans-Joachim Bentz, and Ramesh Kapadia. 1991. Chance Encounters: Probability in Education A Probabilistic Perspective.

Berry, Michael, Azlinah Mohamed, and Bee Wah Yap. 2019. Unsupervised and Semi-Supervised Learning Supervised and Unsupervised Learning for Data Science.

Burstein, J., Kaplan, R., Wolff, S., and Lu, C. (1996). Using Lexical Semantic Techniques to Classify Free- Responses. In E. Viegas, editor, Proceedings ofthe ACL SIGLEXWorkshop on Breadth and Depth ofSemantic Lexicons, pages 20–29, Santa Cruz, California. Association for Computational Linguistics.

Bachman, L. F., Carr, N., Kamei, G., Kim, M., Pan, M. J., Salvador, C., and Sawaki, Y. (2002). A Reliable Approach to Automatic Assessment of Short Answer Free Responses. In S.-C. Tseng, T.-E. Chen, and Y.-F. Liu, editors, Proceedings ofthe Nineteenth International Conference on Computational Linguistics, volume 2 of COLING '02, pages 1–4, Taipei, Taiwan. Association for Computational Linguistics.

Barrón-Cedeno, A., Rosso, P., Agirre, E., Labaka, G., 2010. Plagiarism detection across

distant language pairs. In: Proceedings of the 23rd International Conference on Computational Linguistics, August. Association for Computational Linguistics, pp. 37–45.

Chaitanya Reddy Patlolla. (2018, December 10). Understanding the concept of Hierarchical clustering Technique. https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec

Callear, D., Jerrams-Smith, J., and Soh, V. (2001). CAA of Short Non-MCQ Answers. In M. Danson and C. Eabry, editors, Proceedings of the Fifth Computer Assisted Assessment Conference, pages 1–14, Loughborough, United Kingdom. Loughborough University

Cowie, J. and Wilks, Y. (2000). ZIn R. Dale, H. Moisl, and H. Somers, editors, Handbook ofNatural Language Processing, chapter 10, pages 241–260. Marcel Dekker, New York City, New York, first edition.

Dice, L.R., 1945. Measures of the amount of ecologic association between species.Ecology 26 (3), 297–302.

Dzikovska Myroslava O., Rodney Nielsen, Chris Brew,Claudia Leacock, Danilo Giampiccolo, Luisa Ben-tivogli, Peter Clark, Ido Dagan, and Hoa Trang Dang. 2013. Semeval-2013 Task 7: The Joint Stu- dent Response Analysis and 8th Recognizing Textual Entailment Challenge. In SemEval.

Domingos, P. (2012). A few useful things to know about machine learning. Communications of the ACM, 55(10), 78–87. https://doi.org/10.1145/2347736.2347755

Dorian Pyle, C. S. J. (n.d.). An executive's guide to machine learning | McKinsey. Retrieved March 4, 2020, from https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/an-executives-guide-to-machine-learning

D. L. T. Rohde, L. M. Gonnerman, et D. C. Plaut, « An Improved Method for Deriving Word Meaning from Lexical », Cogn. Psychol., vol. 7, p. 573-605, 2004.

Ellis. Batten. (1966). "The imminence of... grading essays by computer". The Phi Delta Kappan. 47 (5): 238–243.

E. NEGRE, « COMPARAISON DE TEXTES: QUELQUES APPROCHES... », 2013.

E. H. SNOUSSI , A.A. MADANI, «Développement d'un PLUGIN d'évaluation automatique

des réponses courtes pour une plateforme de télé-enseignement ». Mémoire master USDB 1. 2018/2019.

Fellbaum, C. (1998). WordNet: An electronic lexical database. Massachusetts: MIT Press. Available

F. R. OUKINA , I. AMAR SETTI, «Elaboration d'un corpus de test pour un système d'évaluation automatique des réponses courtes». Mémoire master USDB 1. 2018/2019.

Gomaa et A. Fahmy, « Automatic scoring for answers to Arabic test questions », Comput. Speech Lang., vol. 28, 2013.

Gomaa, W. H., & Fahmy, A. A. (2014). Automatic scoring for answers to Arabic test questions. Computer Speech and Language, 28(4), 833–857. https://doi.org/10.1016/j.csl.2013.10.005

Gabrilovich and S. Markovitch. 2006. Overcoming the brittleness bottleneck using Wikipedia: Enhanc- ing text categorization with encyclopedic knowl- edge. In Proceedings ofthe National Conference on Artificial Intelligence (AAAI), Boston.

Gütl, C. (2007). e-Examiner: Towards a Fully-Automatic Knowledge Assessment Tool Applicable in Adaptive E-Learning Systems. In P. H. Ghassib, editor, Proceedings ofthe Second International Conference on Inter- active Mobile and Computer Aided Learning,

Gusfield, D., 1997. Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology. Cambridge University Press.

Gabrilovich, E., Markovitch, S., 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: Proceedings of the20th International Joint Conference on Artificial Intelligence, January, vol. 6, p. 12.

Hall, P.A., Dowling, G.R., 1980. Approximate string matching. ACM Computing Surveys (CSUR) 12 (4), 381–402.

Hebah, Rababah, and Ahmad T. Al-Taani. 2017. ICIT 2017 : The 8th International Conference on Information Technology : Internet of Things IoT : Conference Proceedings : May 17th - 18th, 2017, Amman, Jordan.

Hou, W.-J., Tsao, J.-H., Li, S.-Y., and Chen, L. (2010). Automatic Assessment of Students' Free-Text Answers with Support Vector Machines. In M. Ali, C. Fyfe, N. García-

Pedrajas, and F. Herrera, editors, Proceedings of the Twenty-Third International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, volume 1, pages 235–243, Cordoba, Spain. Springer.

Jason Brownlee. (2019, August 12). Supervised and Unsupervised Machine Learning Algorithms. https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

Jonas Muller and Aditya Thygarajan, “Siamese Recurrent Architecture for learning sentence similarity”, AAAI-16

Jaccard, P., 1901. Etude comparative de la distribution florale dans une portion des Alpes et du Jura. Impr. Corbaz.

Jaro, Matthew A. 1989. “Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida.” Journal of the American Statistical Association 84(406):414–20.

Jiang, J., and Conrath, D. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings of the International Conference on Research in Computational Linguistics.

Kenton W. Murray, & Orii, N. (2010). Automatic essay scoring system. US Patent 7,831,196, 1–8.

Karl , Pearson (20 June 1895). "Notes on regression and inheritance in the case of two parents". Proceedings of the Royal Society of London. 58: 240–242.

Kulhare, S. (2017). Deep Learning for Semantic Video Understanding. Pan 2004, 1–7. http://scholarworks.rit.edu/theses%0Ahttp://scholarworks.rit.edu/theses Recommended

Kolb, P., 2008. Disco: A multilingual database of distributionally similar words. In: Proceedings of KONVENS-2008, Berlin.

Kohavi, R., and Provost, F. 1998. On Applied Research in Machine Learning. In Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process, Columbia University, New York, volume 30

Leacock, C. and Chodorow, M. (2003). C-rater: Automated Scoring of Short-Answer

Questions. Computers and the Humanities, 37(4), 389–405.

Leacock, C., and Chodorow, M. 1998. Combining local context and WordNet sense similarity for word sense identification. In WordNet, An Electronic Lexical Database. The MIT Press

Landauer, T.K., Dumais, S.T., 1997. A solution to Platos problem: the latent semantic analysis theory of acquisition, induction and representationof knowledge. Psychological Review 104 (2), 211–240.

Lesk, M. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In Proceedings of the SIGDOC Conference 1986.

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In M.-F. Moens and S. Szpakow- icz, editors, Proceedings of the First Text Summarization Branches Out Workshop at ACL, pages 74–81, Barcelona, Spain. Association for ComputationalLinguistics.

Lin, D. 1998. An information-theoretic definition of similarity. In Proceedings of the International Conf. on Machine Learning.

Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. Proceedings of the National Conference on Artificial Intelligence, 1, 775–780.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, January 16). Efficient estimation of word representations in vector space. 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems. http://arxiv.org/abs/1310.4546

Mohamed A. Zahran, Ahmed Magooda, Ashraf Y. Mahgoub, Hazem Raafat, Mohsen Rashwan, and Amir Atyia, A. (2015). Computational linguistics and intelligent text processing: 16th international conference, CICLing 2015 Cairo, Egypt, april 14–20, 2015 proceedings, part I. Lecture Notes in Computer Science (Including Subseries Lecture

Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9041(April). https://doi.org/10.1007/978-3-319-18111-0

Madnani, N., Burstein, J., Sabatini, J., and Reilly, T. O. (2013). Automated Scoring of a Summary Writing Task Designed to Measure Reading Comprehension. In J. Tetreault, J. Burstein, and C. Leacock, editors, Proceedings ofthe Eighth Workshop on Innovative Use ofNLP for Building Educational Applications, pages 163–168, Atlanta, Georgia. Association for Computational Linguistics.

Mishra, A. (2018, November 1). Metrics to Evaluate your Machine Learning Algorithm. Retrieved 2020, from https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234

Mitchell, T., Russell, T., Broomhead, P., and Aldridge, N. (2002). Towards Robust Computerised Marking of Free- Text Responses. In Proceedings of the Sixth Computer Assisted Assessment Conference, pages 233–249, Loughborough, United Kingdom.

Mohler, M. and Mihalcea, R. (2009). Text-to-text Semantic Similarity for Automatic Short Answer Grading. In A. Lascarides, C. Gardent, and J. Nivre, editors, Proceedings of the Twelfth Conference of the European Chapter of the Association for Computational Linguistics, pages 567–575, Athens, Greece. Association for Computational Linguistics

Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to Grade Short Answer Questions Using Semantic Similarity Measures and Dependency Graph Alignments. In ACL

Mikolov, T., et al. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Md Arafat Sultan, Steven Bethard, and Tamara Sum- ner. 2014. Back to Basics for Monolingual Align- ment: Exploiting Word Similarity and Contextual Ev- idence. Transactions ofthe Association for Computational Linguistics, 2 (May).

Markovitch, Egozi, Ofer; Shaul; Gabrilovich, Evgeniy (2011). "Concept-Based Information Retrievalusing Explicit SemanticAnalysis" (PDF). ACM Transactions on Information Systems. 29 (2): 1–34. doi:10.1145/1961209.1961211..

Miller, G.A., 1995. WordNet: a lexical database for English. Communications of the ACM 38

(11), 39–41

Nagesh Singh Chauhan. (2019, October 3). Introduction to Artificial Neural Networks(ANN) - Towards Data Science. https://towardsdatascience.com/introduction-to-artificial-neural-networks-ann-1aea15775ef9

Nau, Jonathan, Aluizio Haendchen Filho, and Guilherme Passero. 2017. "EVALUATING SEMANTIC ANALYSIS METHODS FOR SHORT ANSWER GRADING USING LINEAR REGRESSION." PEOPLE: International Journal of Social Sciences 3(2): 437–50. https://grdspublishing.org/index.php/people/article/view/570.

Neill, S. P., & Hashemi, M. R. (2018). Ocean Modelling for Resource Characterization. In Fundamentals of Ocean Renewable Energy. https://doi.org/10.1016/b978-0-12-810448-4.00008-2

Neill, Simon P., and M. Reza Hashemi. 2018. Ocean Modelling for Resource Characterization.

Needleman, S.B., Wunsch, C.D., 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology 48 (3), 443–453.

Suzen, Neslihan, Alexander Gorban, Jeremy Levesley, and Evgeny Mirkes. 2018. "Automatic Short Answer Grading and Feedback Using Text Mining Methods." 1–20.

O. HAMEL, S. LAMARI, «Les Réseaux de Neurones pour La Génération Automatique de Paraphrases». Mémoire master USDB 1. 2019/2020.

Peterson, J.L., 1980. Computer programs for detecting and correcting spelling errors. Communications of the ACM 23 (12), 676–687.

Patil, Pranjal, and Ashwin Agrawal. 2018. Auto Grader for Short Answer Questions. pages 1–10, Amman, Jordan.

Patwardhan, S., 2003. Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. University of Minnesota, Duluth (doctoral dissertation).

Resnik, P. 1995. Using information content to evaluate semantic similarity. In Proceedings of

the 14th International Joint Conference on Artificial Intelligence.

Rocy Martinez, HeeDong Hong -, and Daewon Lee -. 2013. “Automated Essay Scoring System By Using Support Vector Machine.” International Journal of Advancements in Computing Technology 5(11): 316–22.

Ramachandran, Jian Cheng, and Peter Foltz. 2015. Identifying Patterns For Short Answer Scor- ing using Graph-based Lexico-Semantic Text Match- ing. In SemEval.

Saha, S., Dhamecha, T. I., Marvaniya, S., Sindhgatta, R., & Sengupta, B. (2018). Sentence level or token level features for automatic short answer grading?: use both. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10947 LNAI(April), 503–517. https://doi.org/10.1007/978-3-319-93843-1_37

S.ABDELAOUI,« Généralisation d’approches basées Espace Sémantique pour un système d’évaluation des réponses courtes adapté à la langue arabe». Mémoire master USDB 1. 2018/2019.

Steven. Burrows, I. Gurevych, and B. Stein, The Eras and Trends of Automatic Short Answer Grading, vol. 25. 2015.

Swamynathan. 2017. Mastering Machine Learning with Python in Six Steps.

Tony Yiu. (2019, June). Understanding Random Forest - Towards Data Science. https://towardsdatascience.com/understanding-random-forest-58381e0602d2

T.K. Landauer and S.T. Dumais. 1997. A solution to plato’s problem: The latent semantic analysis the- ory of acquisition, induction, and representation of knowledge. Psychological Review, 104.

Theobald, Oliver. 2017. Machine Learning For Absolute Beginners.

Turney, P., 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: Proceedings of the Twelfth European Conference on MachineLearning (ECML).

Winkler, W.E., 1990. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage.

Wu, Z., and Palmer, M. 1994. Verb semantics and lexical selection. In Proceedings of the Annual Meeting of the Association for Computational Linguistics.

Willmott, C. and Matsuura, K.: Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in assessing average model performance, Clim. Res., 30, 79–82, 2005.

Wu, Shih-Hung, W.-F. (2019). A Short Answer Grading System in Chinese by Support Vector Approach. 125–129. https://doi.org/10.18653/v1/w18-3718

Zhang, Y., Shah, R., & Chi, M. (2016). Deep learning + student modeling + clustering: A recipe for effective automatic short answer grading. Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016, 562–567.

Zhang, L., Huang, Y., Yang, X., Yu, S., & Zhuang, F. (2019). An automatic short-answer grading model for semi-open-ended questions. Interactive Learning Environments, 1–14. https://doi.org/10.1080/10494820.2019.1648300