

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Chadli Bendjedid
Faculté des Sciences et de la Technologie
Département d'Informatique



الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي
جامعة الشاذلي بن جديد
كلية العلوم و التكنولوجيا
قسم الاعلام الالى

MEMOIRE

Présenté par

Arif Rami

Pour l'obtention de diplôme de

MASTER

Filière : Informatique

Spécialité : Systèmes Informatiques Intelligents

Thème

Preference Elicitation of learners to personalize the learning process in MOOCs.

Elicitation des préférences des apprenants pour personnaliser le processus
d'apprentissage dans les MOOCs

Soutenu le : 22/ 06 / 2024

Devant le Jury composé de :

Qualité	Nom et Prénom	Grade	Université
Président	Mr. Touahri Dj	MAA	Chadli Bendjedid El-Tarf
Rapporteur	Mme. Maatalah M	MCB	Chadli Bendjedid El-Tarf
Examineur	Mme. Sahki A	MAB	Chadli Bendjedid El-Tarf

Année Universitaire : 2023/2024

i. Remerciements

Je tiens à exprimer ma profonde gratitude envers tous ceux qui ont contribué à ma formation et à la

réalisation de ma thèse. Tout d'abord, je remercie chaleureusement mon encadrant, **Mme. MAATALLAH**

pour son soutien constant, ses conseils éclairés et son expertise précieuse tout au long de ce parcours.

Je souhaite également adresser mes remerciements les plus sincères à tous mes professeurs qui ont

contribué à ma formation. Leur savoir, leur passion pour l'enseignement et leur dévouement ont été une

source d'inspiration et ont grandement enrichi mon parcours académique.

Un sincère remerciement également à tous les membres du jury d'avoir accepté l'évaluation de mon

travail avec attention et objectivité.

J'adresse mes remerciements à mes collègues de recherche pour leur collaboration et leur aide précieuse

durant cette aventure académique.

Enfin, je suis reconnaissant(e) envers ma famille pour leur amour, leur soutien inconditionnel et leur

encouragement tout au long de cette étape importante de ma vie.

Merci du fond du cœur à toutes ces personnes qui ont contribué à ma réussite académique.

ii. Dédicace

À mes parents, vous avez été mes premiers mentors, mes guides et mes plus grands soutiens. Votre amour inconditionnel, vos encouragements constants et votre confiance en moi m'ont donné la force nécessaire pour surmonter les défis et atteindre mes objectifs.

À mes parents, mes chers camarades wassim, dhia,
Houdna, ainsi qu'à tous mes collègues,

Je tiens à dédier ces mots empreints de gratitude et d'affection. Votre présence et votre soutien ont été des piliers essentiels tout au long de mon parcours académique.

À mes frères, vous avez été mes complices, mes confidentes et mes inspirations. Votre soutien indéfectible et votre présence bienveillante ont été un moteur dans ma quête de réussite.

À mes chers camarades, et à tous mes collègues, nous avons partagé des moments d'apprentissage, de travail acharné et de rires. Votre collaboration, vos discussions stimulantes et votre solidarité m'ont aidé à grandir académiquement et personnellement.

Ensemble, vous avez formé une équipe solidaire qui m'a soutenu tout au long de mon cheminement. Vos encouragements, et votre confiance ont été les moteurs qui m'ont poussé à donner le meilleur de moi-même.

Du fond du cœur, je vous remercie pour votre amour, votre soutien et votre amitié indéfectibles. Cette réussite est également la vôtre.

Avec une gratitude infinie.

Rami

iii. Résumé

Dans ce manuscrit, nous avons mené une analyse et une étude approfondie des domaines du e-learning, plus particulièrement les MOOCs, et des systèmes de recommandation en mettant l'accent sur la fusion entre eux afin d'améliorer la satisfaction des apprenants en se concentrant sur l'élicitation de leur préférences dans le processus de la recommandation. Pour cette étude, nous avons utilisé le modèle BERT base uncased de la bibliothèque Transformers de Hugging Face, développé par Google et dédié pour le traitement des requêtes, des commentaires, la prédiction et la génération des textes en anglais.

L'objectif de ce travail est de proposer une solution au problème de la diversification des préférences des apprenants, en exploitant les avantages de BERT pour extraire les préférences à partir des commentaires fournis par les apprenants. Ensuite, ces préférences seront incorporées dans le processus de la recommandation afin d'améliorer la pertinence des recommandations de cours et d'augmenter la satisfaction des apprenants. Le système a été implémenté en utilisant le langage de programmation Python.

Une série de tests a été menée afin de démontrer les performances de notre système, suivie d'une comparaison avec les modèles BiLSTM et LSTM. Les résultats obtenus étaient satisfaisants en termes de classification des sentiments et des recommandations générées.

Mots clés : Élicitation des préférences, MOOCs, Systèmes de recommandation, E-learning, BERT base uncased, Pertinence, diversité.

iv. Abstract

In this manuscript, we conducted an in-depth analysis and study of the fields of e-learning, particularly MOOCs, and recommendation systems, focusing on the integration between them to enhance learner satisfaction by eliciting their preferences in the recommendation process. For this study, we utilized the BERT base uncased model from the Hugging Face Transformers library, developed by Google, specifically designed for processing queries, comments, prediction, and text generation in English.

The objective of this work is to propose a solution to the issue of diversifying learner preferences by leveraging BERT's capabilities to extract preferences from learner comments. These preferences are then incorporated into the recommendation process to improve the relevance of course recommendations and increase learner satisfaction. The system was implemented using the Python programming language.

A series of tests was conducted to demonstrate the performance of our system, followed by a comparison with the BiLSTM and LSTM models. The results obtained were satisfactory in terms of sentiment classification and the recommendations generated.

Keywords: Preference elicitation, MOOCs, Recommendation systems, E-learning, BERT base uncased, Relevance, Diversity.

ملخص v.

في هذه المخطوطة، أجرينا تحليلاً ودراسة متعمقة لمجالات التعلم الإلكتروني، وتحديدًا المساقات التعليمية الإلكترونية المفتوحة المصادر، وأنظمة التوصية مع التركيز على الدمج بينهما لتحسين رضا المتعلمين من خلال التركيز على استنباط تفضيلاتهم في عملية التوصية. واستخدمنا في هذه الدراسة النموذج الأساسي غير المدمج من مكتبة المحولات الخاصة بـ "عناق الوجه" التي طورناها جوجل والمخصصة لمعالجة الاستعلام والتعليق والتنبؤ وتوليد النصوص باللغة الإنجليزية.

والهدف من هذا العمل هو اقتراح حل لمشكلة تنويع تفضيلات المتعلمين، من خلال استغلال مزايا BERT لاستخراج التفضيلات من التعليقات التي يقدمها المتعلمون. سيتم بعد ذلك دمج هذه التفضيلات في عملية التوصيات من أجل تحسين ملاءمة توصيات الدورة التدريبية وزيادة رضا المتعلمين. تم تنفيذ النظام باستخدام لغة برمجة بايثون.

وقد تم إجراء سلسلة من الاختبارات لإثبات أداء نظامنا، تليها مقارنة مع نموذجي BiLSTM و LSTM كانت النتائج التي تم الحصول عليها مُرضية من حيث تصنيف المشاعر والتوصيات التي تم إنشاؤها.

الكلمات المفتاحية: استنباط التفضيلات، MOOCs، أنظمة التوصيات، التعلم الإلكتروني، قاعدة BERT غير المبنية على أساس

Table des matières

Remerciements.....	i
Dédicace.....	ii
Résumé.....	iii
Abstract	iv
v.....	ملخص
Table des matières.....	vi
Liste des figures	x
Liste des tableaux.....	xii
Liste des acronymes.....	xiii
Introduction Générale.....	1
Chapitre 1 : Etat de l'Art	
1.1. Introduction.....	3
1.2 Les plateformes e-learning.....	3
1.2.1 C'est quoi une plateforme E-Learning	3
1.2.3 Histoire et évolution de la plateforme e-learning.....	3
1.2.3 Les types du e-learning	4
1.2.3.1 E-Learning Asynchrone (collaboratif).....	4
1.2.3.2 E-Learning synchrone	4
1.2.4 Les tendances du e-learning.....	4
1.2.4.1 La formation à distance	5
1.2.4.2 Enseignement par correspondance.....	5
1.2.4.3 L'Apprentissage Massivement Ouvert (Les MOOCs)	5
1.2.4.3.1 Définition.....	5
1.2.4.3.2 Brève histoire.....	5
1.2.4.3.3 Typologies.....	6
1.2.4.3.3.1 cMOOCs.....	6
1.2.4.3.3.2 xMOOCs.....	6
1.2.4.3.4 Les MOOCs et le E-Learning.....	6
1.2.4.3.5 Amélioration de l'apprentissage avec les MOOCs.....	8

1.2.4.5.6 Les modèles d'affaires des plateformes MOOCs	9
1.2.4.5.6.1 Un modèle d'affaires, quèsaco ? dans Les MOOCs.....	9
1.2.4.5.7 Les principales plateformes.....	11
1.2.5 Le principe de E-Learning.....	12
1.2.5.1 Principes de contiguïté.....	12
1.2.5.2 Principe du multimédia.....	13
1.2.5.3 Principe de Modalité.....	13
1.2.5.4 Principe de Cohérence.....	13
1.2.5.4 Principe de la Voix.....	14
1.2.5.5 Principe de Redondance.....	14
1.3 Les systèmes de recommandations dans le E-Learning.....	15
1.3.1 Introduction.....	15
1.3.2 Définition du système de recommandation	15
1.3.3 Les types systèmes de recommandation dans le E-Learning	15
1.3.3.1 recommandation basées sur le contenu.....	15
1.3.3.2 Techniques de recommandation basées sur le filtrage collaboratif	16
1.3.3.3 Les techniques de recommandation basées sur les connaissances.....	16
1.3.3.4 Les techniques de recommandation hybrides.....	17
1.3.4 Travaux connexes sur les systèmes de recommandation dans le domaine de l'apprentissage en ligne.....	18
1.3.4.1 Systèmes de recommandation pour l'éducation personnalisée.....	18
1.3.4.2 Analyse des données d'apprentissage.....	18
1.3.4.3 Systèmes de recommandation adaptatifs.....	18
1.3.4.4 Évaluation des systèmes de recommandation dans l'éducation.....	19
1.4 L'Elicitation de préférences des apprenants dans les MOOCs.....	20
1.4.1 Introduction.....	20
1.4.2 Définition.....	20
1.4.3 Les méthodes utilisées dans l'élicitation des préférences.....	21
1.4.3.1 Le feedback des pairs.....	21
1.4.3.2 Suivi des réponses aux quiz et aux évaluations.....	21
1.4.3.3 Analyse de l'activité en ligne	21
1.4.3.4 Le profilage des utilisateurs.....	21

1.4.3.5 La rétroaction implicite et explicite.....	22
1.4.3.6 Les enquêtes et les questionnaires.....	22
1.4.4 Les Difficultés et les Problèmes de l'Élicitation des Préférences des apprenants dans les moocs.....	22
1.4.4.1 Problème de démarrage à froid.....	22
1.4.4.2 Le problème de la nouveauté.....	23
1.4.4.3 Problème de sérendipité.....	23
1.4.4.4 Problème de la quantité de données limitée	23
1.4.4.5 Le problème de la qualité des données.....	23
1.4.4.6 Problème du manque de fiabilité.....	23
1.4.5 L'élicitation des préférences à partir des commentaires.....	24
1.4.6 Travaux existants sur l'élicitation des préférences à partir des commentaires..	24
1.4.7 Le modèle BERT base uncased	25
1.4.7.1 Définition.....	25
1.4.7.2. Entrées du modèle.....	26
1.4.7.3Encodeur de bert.....	26
1.4.7.2 Architecture de BERT base uncased.....	27
1.5 conclusion	28
Chapitre 2 : Conception et implémentation	
2.1 Introduction.....	29
2.2 Contexte.....	29
2.3 Problématique.....	29
2.4 Solution proposée.....	29
2.5 Motivation et objectifs.....	30
2.6 Conception du système	30
2.6.1 PHASE1 : L'Elicitation Des Préférences.....	31
2.6.1.1Etape 1 Préparation Des Données.....	32
2.6.1.2 Etape 2 Configuration Du Modèle ber-base-Uncased.....	37
2.6.1.3 Etape 3 Apprentissage du modèle Bert base Uncased.....	41
2.6.1.4 Evaluation du modèle bert-base-uncased.....	42
2.6.2 PHASE 2 : Génération Des Recommandations basées Préférences diversifiées.	44

2.6.2.1 Le Filtrage Basé sur les Sentiments.....	44
2.7 Conclusion.....	45
Chapitre 3 : Expérimentations et tests	
3.1 Introduction.....	46
3.2 Base de données (DataSet).....	46
3.3 Outils de développements.....	46
3.3.1 Le Langage de programmation (python).....	46
3.3.2 Les bibliothèques utilisées.....	47
3.3.3 Le matériel utilisé.....	48
3.4 Les métriques d'évaluation.....	48
3.4.1 Accuracy.....	48
3.4.2 Matrice de confusion.....	49
3.5 Evaluation du système bertrec	49
3.5.1 Evaluation du modèle bert-base-uncased.....	49
3.5.1.1 Performance du système pour 3 époques.....	49
3.5.1.2 Performance du système pour 10 époques.....	50
3.5.1.3 Performance du système dans 15 époques	52
3.5.1.4 Discussion des résultats obtenus.....	53
3.5.2 Teste de modèle bert base Uncased	53
3.5.3 Comparaison des résultats.....	56
3.5.3.1 Modèle 1 : Modèle LSTM.....	57
3.5.3.2 Modèle 2 : Le modèle BiLsTm.....	58
3.5.3.3 Comparaison entre les trois modèles.....	59
3.6 Génération Des Recommandations basées Préférences diversifiées et normale..	60
3.6.1 Réglage des paramètres.....	60
3.6.1.1. Analyse des résultats.....	61
3.7 Conclusion.....	63
Conclusion générale.....	64
Références	
A. Références Bibliographiques.....	65
B. Références Web (Techniques).....	70

Liste des figures

Figure 1.1.les cinq groupes d'agents qui sont susceptibles de graviter autour d'une plateforme MOOCs.....	10
Figure 1.2. Entrées du modèle.....	26
Figure 1.3 explique les deux encodeurs de bert.....	27
Figure1.4. BERT base uncased -model-architecture.....	28
Figure2.1. Elicitation des préférences à partir des commentaires.....	31
Figure 2.2. Génération de la recommandation basée sur les commentaires.....	31
Figure 2.3. Les étapes de l'élicitation des préférences dans le système bert base uncased....	32
Figure 2.4. Chargement et lecture du fichier CSV « course_info ».....	33
Figure 2.5. Chargement et lecture du fichier CSV « comments.csv ».....	33
Figure 2.6. Suppression des commentaires qui ne sont pas en anglais avec la bibliothèque langdetect.....	34
Figure 2.7. La création d'un tableau sentiments à partir de rate.....	35
Figure 2.8. Le nettoyage des données.....	36
Figure 2.9. Nettoyage des données en utilisant la bibliothèque « re » et « spacy ».....	37
Figure 2.10. L'importation des bibliothèques nécessaire.....	38
Figure 2.11. Chargement du Tokenizer et du Modèle Bert base Uncased.....	38
Figure 2.12. La tokenization des commentaires.....	39
Figure 2.13. Préparation des Données pour l'Entraînement et la Validation.....	39
Figure 2.14. tokenization des Données pour BERT.....	40
Figure 2.15. Création de la Classe Dataset pour BERT.....	41
Figure 2.16. Définition des Métriques pour l'Évaluation du Modèle.....	41
Figure 2.17. Réglage des paramètres d'entraînement.....	42
Figure 2.18: Évolution des performances du système.....	43
Figure 2.19. Performance d'entraînement du système.....	43
Figure 2.20. Les étapes de la recommandation.....	44
Figure 3.1. Performance du système en 3 époques.....	49
Figure 3.2. Perte Vs. Précision du système en 3 époques.....	49

Figure 3.3. Performance du système en 10 époques.....	51
Figure 3.4. Précision Vs. Perte en 10 époques.....	51
Figure 3.5. Performance du système en 15 époques.....	52
Figure 3.6. Précision Vs. Perte en 15 époques.....	53
Figure 3.7. Le teste de modèle bert base uncased.....	56
Figure 3.8. Performance du modèle LSTM.....	57
Figure 3.9. Performance du modèle BiLSTM.....	58
Figure 3.10. Comparaison entre les trois modèles.....	59
Figure 3.11. Courbe de précision rappel pour alpha= 0.5.....	60
Figure 3.12. Courbe de précision rappel pour alpha= 0.6.....	61
Figure 3.13. Courbe de précision rappel pour alpha= 0.8.....	62

Liste des tableaux

Tableau 1 : Les principales plateformes des MOOCs.....	11
Tableau 2. Tableau récapitulatif des travaux sur l'élicitation des préférences à partir des commentaires.....	25
Tableau 3. Les paramètres optimaux d'entraînement.....	48
Tableau 4. Résultats du modèle avec architecture LSTM et dropout.....	57
Tableau 5. Résultats du modèle avec architecture Bidirectionnelle LSTM et dropout.....	58
Tableau 6. Les résultats des 3 modèle.....	59

Liste des acronymes

MOOCs	Cours en ligne ouverts et massifs (Massive Open Online Courses)
LSTM	Mémoire à long court terme (Long Short-Term Memory)
BiLSTM	Mémoire à long court terme bidirectionnelle (Bidirectional Long Short-Term Memory)
NLp	Traitement du langage naturel (Natural Language Processing)
Spoc	Cours en ligne privés et de petite taille (Small Private Online Course)
CooC	Cours en ligne ouverts (Corporate Online Open Course)
LMS	Système de gestion de l'apprentissage (Learning Management System)
IBM	Société internationale des machines informatiques (International Business Machines Corporation)
CF	filtrage colaboratif ((Collaborative Filtering)
MLM	Modèle de langage masqué (Masked Language Model)
CSV	Valeurs Séparées par des Virgules (Comma-Separated Values)
GPU	Unité de traitement graphique (Graphics Processing Unit)
GPT	Transformateur Génératif Pré-entraîné (Generative Pre-trained Transformer)
Roberta	Approche d'Optimisation Robuste de l'Entraînement Préliminaire de BERT (Robustly Optimized BERT Pretraining Approach)

Introduction Générale

Dans le monde éducatif contemporain, les méthodes d'enseignement et d'apprentissage ont radicalement évolué, passant d'un modèle traditionnel centré sur l'éducation en présentiel à des approches plus flexibles et technologiquement avancées. Cette transformation est principalement catalysée par l'essor des technologies de l'information et de la communication (TIC), qui ont révolutionné la manière dont les connaissances sont transmises et acquises. Alors que l'enseignement traditionnel se limitait à la salle de classe et à l'assiduité, l'ère numérique a introduit des paradigmes nouveaux, notamment l'apprentissage en ligne ou e-learning.

Le e-learning, également connu sous le nom d'apprentissage électronique, tire parti des technologies numériques pour faciliter l'acquisition de connaissances et de compétences à distance. Cette approche permet aux apprenants de bénéficier d'un accès flexible à une variété de ressources pédagogiques, de participer à des cours en ligne interactifs, et de suivre des programmes de formation à leur propre rythme, indépendamment de leur emplacement géographique. Les MOOCs (Massive Open Online Courses), qui représentent une évolution majeure dans ce domaine, permettent un accès ouvert et libre à une vaste gamme de cours proposés par des institutions de renom. Ils rendent ainsi l'éducation de qualité accessible à une audience mondiale.

Dans ce contexte de transformation, les systèmes de recommandation jouent un rôle crucial. Ils visent à personnaliser l'expérience d'apprentissage en proposant des recommandations de contenus et de ressources qui correspondent aux préférences et besoins individuels des apprenants. Cette personnalisation est essentielle pour maintenir l'engagement des apprenants et améliorer l'efficacité de leur apprentissage. L'élicitation des préférences des apprenants, c'est-à-dire la capacité de déterminer leurs goûts, leurs intérêts et leurs besoins à partir de leurs interactions et de leurs commentaires, est une tâche complexe surtout lorsqu'il s'agit de préférences multiples et diverses. Cependant, l'élicitation des préférences a prouvé son importance fondamentale pour les systèmes de recommandation et dans le domaine de l'e-learning. Cela soulève la question centrale de ce projet :

Comment peut-on éliciter les préférences variables des apprenants dans les MOOCs afin de générer des recommandations pertinentes?

Afin de répondre à cette problématique, l'étude présentée dans ce manuscrit vise à résoudre la problématique de l'élicitation des préférences des apprenants dans les MOOCs en utilisant des méthodes avancées de traitement du langage naturel. À travers l'utilisation du modèle BERT Base Uncased pour l'analyse des commentaires en anglais, l'objectif principal est d'extraire de manière précise et efficace les préférences individuelles des apprenants. Ces préférences sont ensuite employées dans un processus de recommandation, qui se divise en deux catégories : une recommandation traditionnelle et une recommandation diversifiée, en fonction des feedbacks et des interactions des apprenants avec les cours.

Le processus commence par la collecte et le prétraitement des données d'interaction des apprenants, comprenant les évaluations, les commentaires et les notes. Un modèle de traitement du langage naturel est entraîné sur ces données pour comprendre et extraire les préférences variables des apprenants. Ensuite, des techniques d'apprentissage automatique sont appliquées pour développer un modèle prédictif capable de recommander des cours adaptés à chaque apprenant.

Ce modèle est continuellement ajusté en fonction des retours des apprenants sur les recommandations fournies, assurant ainsi une amélioration continue de la pertinence et de la précision du système de recommandation. Enfin, une évaluation approfondie du système est réalisée sur un Dataset des cours et commentaires associés, en comparant les résultats avec d'autres modèles de l'état de l'art pour affiner encore d'avantage ses performances.

Cette approche cherche à offrir une expérience d'apprentissage en ligne plus personnalisée et enrichissante, répondant aux besoins individuels des apprenants tout en exploitant pleinement les avantages des technologies de pointe en traitement du langage naturel et en apprentissage automatique. De plus, elle vise à fournir des recommandations diversifiées et personnalisées afin d'améliorer l'expérience d'apprentissage dans les MOOCs et d'augmenter la satisfaction des apprenants.

Ce mémoire est organisé en trois chapitres comme suit :

- **État de l'art sur les Systèmes de Recommandation dans le E-Learning :**

Nous explorons les différentes approches et techniques d'élicitation des préférences, en mettant l'accent sur les systèmes de recommandation dans les MOOCs. Nous détaillons également les concepts fondamentaux du modèle Bert base Uncased et ses applications dans le traitement du langage en anglais.

- **Conception et Implémentation du Système :**

Ce chapitre présente notre approche de recommandation diversifiée basée sur l'élicitation des préférences des apprenants, en détaillant l'architecture du modèle, les algorithmes utilisés, ainsi que le processus de conception et d'implémentation de notre système de recommandation. Nous expliquons ainsi en profondeur la nouvelle approche du filtrage Basé sur les préférences et les sentiments pour la génération des recommandations des cours.

- **Expérimentations et tests :**

Nous décrivons les tests réalisés pour évaluer les performances de notre système comparant ses résultats avec ceux d'autres modèles de référence, tels que LSTM et BiLSTM. Nous analysons les résultats obtenus, discutons des avantages et des limitations de chaque approche.

A la fin, nous terminerons par une conclusion générale et quelques perspectives

1.1. Introduction

Dans l'ancien et dans les années précédentes l'enseignement /l'apprentissage est limité juste à l'éducation présentielle et l'assiduité. Mais aujourd'hui et avec un univers d'extraordinaire changement des savoirs, des cultures, des technologies et avec une révolution de développement. L'enseignement connaît de nombreuse mutation qui le font l'apparition de nouveaux mode d'apprentissage c à d l'enseignement en ligne /à distance qui est connaît sous le nom de e-learning.

C'est l'apprentissage qui utilise les technologies de l'information et de la communication pour permettre aux apprenant d'acquérir des connaissances et des compétences à distances cette approche transforme les salles, les classes et les structures éducatives sous formes plateformes en ligne, des outils internatifs et des ressources numériques.

Le e-learning est surtout adapté pour développer des compétences cognitives et avec méthodes spécifiques des compétences interpersonnelles. La formation en e-learning peut s'effectuer en auto apprentissage (didacticiel d'apprentissage numérique) ou être conduite par un facilitateur.

1.2. Les plateformes e-learning

1.2.1. C'est quoi une plateforme E-Learning :

Un plateforme E-Learning peut prendre différent termes, au début c'est un environnement virtuel numérique crée pour facilite l'apprentissage à distance en intégrant des outils technologies et des fonctionnalités éducative. Par ailleurs Allons des système open sources comme Moodle, canvas et, en passant par les Moocs (massive open online courses) telle que coursera et edx.[1]

Sa particularité réside dans sa capacité a une diversité de contenue pédagogique, telle que les vidéos interactifs, les image et les formus de discussion, offrant aux apprenants un accès flexible des ressources éducatives. [2]

Finalement Ces plateformes d'e-learning varient en fonction de leurs caractéristiques et de leurs objectifs spécifiques. Elles peuvent être utilisées dans divers contextes éducatifs, qu'il s'agisse d'institutions académiques, de formations professionnelles en entreprise, ou même d'initiatives d'auto-apprentissage. Les fonctionnalités principales incluent la gestion de cours, la diffusion de contenu multimédia, la communication en ligne, l'évaluation des performances, et la création d'espaces virtuels propices à l'interaction et à la collaboration. [3]

1.2.2. Histoire et évolution de la plateforme e-learning :

L'évolution du e-learning est une réponse aux changements sociétaux, adoptant de nouveaux outils et de nouveaux environnements d'apprentissage. Les technologies de l'information et de la communication ont ouvert l'accès à une multitude de ressources mondiales. Son parcours historique commence avec les premiers éducateurs comme Confucius et traverse différentes théories et méthodes d'enseignement. [4]

Chapitre 1 Etat de l'art

L'enseignement à distance, une forme moderne d'éducation, a émergé au 19e siècle avec des cours par correspondance, rendant l'éducation accessible à un public plus large. Des institutions telles que le Centre National d'Enseignement à Distance en France, l'Université Nationale d'Éducation à Distance en Espagne et l'Open University au Royaume-Uni ont été créées pour répondre aux besoins des apprenants éloignés. [5]

L'avènement d'Internet a révolutionné l'apprentissage à distance, avec des plateformes comme Moodle en 2001, permettant aux apprenants d'accéder à l'éducation depuis n'importe où. Les progrès technologiques ont également conduit au développement du e-learning assisté par ordinateur dans les années 1970, offrant une flexibilité et un accès illimité aux connaissances. [6]

L'émergence des MOOCs (Massive Open Online Courses) à l'université Stanford en 2013 a marqué une nouvelle ère dans l'e-learning, offrant des cours en ligne ouverts à un large public. Ces avancées témoignent de l'impact positif des TIC sur l'éducation, rendant l'apprentissage plus accessible, efficace et économique. [7]

1.2.3. Les types du e-learning :

1.2.3.1.E-Learning Asynchrone (collaboratif) :

Implique des interactions et des collaborations entre les apprenants et les enseignants qui ne se produisent pas en temps réel et aussi met l'accent sur des éléments fondamentaux du nouveau style pédagogique qui est rendu possible par la technologie c à d que les interlocuteurs puissent étudier à leur propre rythme sans impératif de temps. [8]

1.2.3.2. E-Learning synchrone :

Il s'agit d'une communication dans tous les participants sont connectés au même temps aussi les outils de l'interactivité sont accessibles pour faciliter l'échange et la compréhension comme (chat vidéo conférence)

Il caractérise par :

- **Temps réel** toutes les interactions se produisent simultanément, permettant un échange immédiat d'informations,
- **Interactivité Directe** : Les participants ont la possibilité de poser des questions, fournir des réponses et participer de manière active au cours de la communication.
- **Outils d'Interactivité** : L'emploi de différentes ressources, tels que le chat en direct, la vidéoconférence et les tableaux blancs virtuels, simplifie la facilité de communication interactive. [9]

1.2.4. Les tendances du e-learning

Les tendances du e-learning représente les évolutions et les nouvelles façons qui se produite sur l'apprentissage en ligne depuis son apparition au dix-neuvième siècle (19S). L'objectif principale de cette tendance au bien de cette évolution c'est 'améliorer l'expérience d'apprentissage en ligne en exploitant les progrès technologiques et en adoptant les méthodes pédagogiques les plus efficaces. Cette amélioration vue plusieurs étapes :

1.2.4.1. La formation à distance :

La formation à distance permet aux étudiants d'apprendre sans la présence physique d'un enseignant dans un établissement, favorisant l'interaction en temps réel, les questions et la participation active. [48] Ce mode d'enseignement autonomise les apprenants en leur offrant la flexibilité d'acquérir des connaissances à leur rythme. Initiée au milieu du 19e siècle, la formation à distance a évolué à travers divers supports comme le courrier, les cassettes, la radio et la télévision, pour finalement s'intégrer à l'enseignement assisté par ordinateur. [10]

1.2.4.2. Enseignement par correspondance

L'enseignement par correspondance est une méthode éducative (modalité pédagogique) ou les étudiants reçoivent les contenus éducatifs (les cours, les exercices et les supports pédagogiques par les biais de la poste, du courrier électronique ou d'autres moyens de communication. Elle permet aussi d'atteindre un public plus étendu, sans être limitée par la localisation géographique des apprenants. L'évolution des technologies de l'information et de la communication a profondément modifié l'enseignement par correspondance.

En 1840, Isaac Pittman, pionnier de l'enseignement à distance, a introduit le premier cours par correspondance en Angleterre. Cet événement marque le début de cette méthode d'enseignement qui a rapidement gagné en popularité en Europe puis dans le monde entier. Dans ce système, un établissement et ses professeurs fournissent l'enseignement à distance à travers des cours envoyés à l'élève. Parfois, l'interaction entre le professeur et l'apprenant se limite à l'échange de devoirs ou à un soutien tutoriel. Dans certains cas, on utilise aussi le terme de télé-enseignement [11]

1.2.4.3. L'Apprentissage Massivement Ouvert (Les MOOCs)

1.2.4.3.1. Définition

Au cours des dernière années une nouvelle tendance du E-Learning est apparue (est l'événement le plus marquant de l'année 2012 dans le monde de l'enseignement supérieur en ligne) qui s'appelle MOOC Massive Online Open Courses (Apprentissage Massivement Ouvert).

Un MOOC est un modèle éducatif en ligne qui offre un accès gratuit à un grand nombre de participants via Internet pour peut accueillir un grand nombre d'inscrits simultanément. Les MOOCs proposent habituellement une diversité de contenus, comprenant des vidéos de cours, des lectures, des discussions en ligne et des évaluations. Ils offrent en même temps une flexibilité permettant aux participants de suivre leur propre rythme d'apprentissage [1]

Chapitre 1 Etat de l'art

Dans le contexte universitaire, les MOOCs peuvent être envisagés soit comme une alternative aux grands cours en amphithéâtre, soit comme un support à une méthode pédagogique inversée en petits groupes. La pédagogie inversée implique que le professeur n'enseigne pas le cours traditionnel en amphithéâtre, mais utilise un MOOC à la place. En classe, l'enseignant intervient de manière interactive pour répondre aux questions des étudiants, fournir des éclaircissements et des compléments, etc. Pour les MOOCs destinés à un large public, l'interaction au sein des groupes peut être assurée par des tuteurs, formant ainsi ce qu'on appelle un MOOC tutoré.[13]

1.2.4.3.2. Brève histoire

Dès les années 2000, les premières expériences de MOOC ont émergé, débutant avec le MIT et ensuite à Stanford en 2008. Ces premiers MOOCs se caractérisaient principalement par la mise en ligne de contenu, parfois monotone. En 2008, Georges Siemens et Stephen Downes ont introduit un cours où les étudiants étaient libres d'enrichir le contenu, marquant ainsi le début du MOOC.

Le professeur Sebastian Thrun de Stanford a lancé un cours sur l'intelligence artificielle qui a attiré 160 000 étudiants, dépassant largement ses attentes de 10 000 étudiants. Cela a marqué le début d'investissements massifs dans les MOOC avec la création d'Udacity, edX et Coursera, chacun développant des approches pédagogiques et des modèles de financement différents. Ces initiatives américaines ont été suivies par une expansion mondiale des MOOC.

Pour mieux comprendre les différentes approches pédagogiques, Georges Siemens a proposé une classification des MOOC en deux grandes familles : les cMOOC et les xMOOC. Les xMOOC proposent des cours en ligne, tandis que les cMOOC encouragent les participants à créer le contenu. Les enseignants utilisent souvent les deux approches simultanément, en fonction des besoins pédagogiques.

Quelques standards ont commencé à émerger suite aux premiers retours d'expérience datant de fin 2012 :

- Les vidéos durent rarement plus de 20 minutes, obligeant les professeurs à découper leurs cours.
- Les plateformes intègrent des systèmes de récompenses et des forums pour stimuler les interactions entre les étudiants et maintenir leur motivation.
- Les étudiants sont évalués au cours de la formation, soit par leurs pairs, soit via des auto-évaluations à travers des QCM ou des tests interactifs.

L'évaluation par les pairs a également trouvé sa place dans des écoles physiques telles que l'école 42. Bien que ces pratiques soient encore émergentes, on peut anticiper le développement d'un savoir-faire dans la création et l'animation de MOOC. La révolution de l'enseignement initiée par les MOOC est en constante évolution et continuera à connaître de nombreux développements. À mesure que les retours d'expérience se multiplient, les MOOCs évolueront et se transformeront pour trouver des formes optimales, influençant également de manière significative le domaine de la formation.

1.2.4.3.3. Typologies :

1.2.4.3.3.1.cMOOCs :

Les cMOOCs, ou cours en ligne ouverts et massifs connectivistes, représentent une nouvelle approche pédagogique adaptée à l'ère numérique. Cette étude examine comment cette approche convient à des apprenants aux compétences, motivations et dispositions diverses. À travers des entretiens semi-structurés avec 29 participants du cMOOC Change11, l'étude identifie trois types d'engagement : participation active, participation passive et observation. Elle met également en lumière des facteurs clés facilitant l'engagement, tels que la confiance, l'expérience antérieure et la motivation. Cette recherche enrichit notre compréhension de l'apprentissage dans les cMOOCs en fournissant des données empiriques précieuses. Les résultats soulignent l'adaptabilité de cette modalité d'apprentissage à la diversité des apprenants. Ces conclusions sont utiles pour les concepteurs de futurs cMOOCs afin d'adapter l'expérience d'apprentissage en fonction des besoins variés des apprenants. [14]

1.2.4.3.3.2.xMOOCs :

xMOOCs, ou eXtended Massive Open Online Courses, sont un type de cours en ligne caractérisé par la livraison de contenu structuré et souvent des conférences, des quiz et des évaluations. Contrairement aux cMOOCs, qui mettent l'accent sur l'interaction des participants et l'apprentissage collaboratif, les xMOOCs suivent généralement un modèle d'enseignement plus traditionnel en mettant l'accent sur la diffusion de contenu des formateurs aux apprenants. Ces cours sont souvent offerts par les universités et les plateformes d'apprentissage en ligne et peuvent fournir des certificats ou des crédits à la fin. [15]

1.2.4.3.4. Les MOOCs et le E-Learning

En pleine expansion, le e-learning se décline aujourd'hui sous de très nombreuses formes. Qu'il soit intégral ou « blended », il ne manque pas d'adeptes. L'une des formes les plus connues de ce mode d'enseignement étant sans doute le « MOOC » ou « massive open online courses » dont les variantes sont également multiples. [16]

Il y a quelques années à peine, les MOOC étaient largement méconnus, mais ils ont rapidement émergé comme l'une des méthodes les plus prometteuses pour diffuser le savoir. Initialement introduits dans le milieu universitaire, ces cours en ligne ont ensuite conquis l'ensemble de l'enseignement supérieur et suscitent désormais un intérêt croissant dans le monde des affaires. Parallèlement, ces cours "massifs" en ligne ont évolué sous différentes formes, au point de dérouter même les experts chevronnés de l'e-learning. Les Spoc(Small Private Online Course (Petit Cours Privé en Ligne)), Cooc(Corporate Open Online Course (Cours en Ligne Ouvert pour les Entreprises)) et autres acronymes similaires ont proliféré ces dernières années, soulignant à la fois la vitalité du secteur et la nécessité d'une régulation adéquate.[17]

Cependant, toutes ces formes d'enseignement ne représentent qu'une fraction du panorama global de l'e-learning. En effet, ce mode d'apprentissage englobe une multitude de modalités pédagogiques, adaptées de manière variable à certains secteurs d'activité ou à certaines disciplines. Mais où réside la déférence entre le E-Learning et les MOOCs ?

Chapitre 1 Etat de l'art

Le digital Learning englobe deux modalités clés : l'e-learning et les MOOC. L'e-learning, également appelé FAD (Formation À Distance), se réfère aux formations en ligne limitées à un nombre spécifique d'apprenants. Les MOOC, ou Massive Open Online Courses, sont des formations en ligne ouvertes à tous, diffusées entièrement sur internet. Au-delà de leurs définitions, la différence entre MOOC et e-learning peut parfois sembler subtile. Cependant, il est essentiel de comprendre ces nuances pour faire un choix éclairé entre ces approches de formation digitale. [18]

La différence entre le e-learning et les moocs ne réside pas dans la valeur didactique inhérente de la technologie, ma plutôt aux opportunités intéressantes des couts et de la massive des inscrits.

Le e-Learning présente par ces caractéristiques :

- **Le cours en ligne fermé :** Le cours en ligne fermé utilise une plateforme LMS pour diffuser des modules de formation à un public spécifique, offrant une flexibilité d'apprentissage individuel ou collectif. Dispensés par des organismes ou établissements, ces cours exigent une inscription préalable et bénéficient de la guidance d'un formateur. L'élément distinctif réside dans le suivi interactif du tuteur via la messagerie privée, avec des ressources comprenant des exercices interactifs et des animations. Les corrections manuelles par le formateur sont souvent nécessaires pour les cours diffusés sur la plateforme.[19]
- **Le cours en ligne ouverte :** Un cours en ligne accessible à tous, sans obligation d'inscription préalable, est souvent offert par des organismes de formation ou des établissements scolaires. La validation ne repose pas nécessairement sur des certifications ou diplômes. Contrairement aux cours fermés, aucun suivi par un professeur n'est requis, les contenus étant principalement des exercices et des quiz avec des réponses automatiques. [20]
- **Les ressources d'apprentissage :** Les formations sont regroupées dans une bibliothèque numérique. Les apprenants peuvent choisir librement et récupérer des informations dans une thématique donnée, construisant ainsi leur propre formation en fonction des ressources disponibles. Ce processus favorise l'auto-formation et l'autonomie totale des apprenants.

Dans l'autre côté les MOOCs caractérise par :

- Les MOOCs représentent un hybride entre les cours en ligne fermés et ouverts. Ils offrent une formation en ligne gratuite avec un accès à une diversité de ressources. Ils permettent aux apprenants de naviguer librement parmi ces ressources et de mettre fin à leur formation à tout moment. Une communauté peut se former autour des MOOCs, favorisant les interactions entre apprenants via des espaces d'échange tels que les forums ou blogs. La principale distinction réside dans la quantité de ressources diffusées et suivies, contrairement aux cours fermés qui ont un nombre défini d'apprenants, les MOOCs accueillent un public plus large. Les auteurs de MOOCs fournissent des ressources sous forme de quiz accompagnés de forums de discussion. [21]

1.2.4.3.5. Amélioration de l'apprentissage avec les MOOCs :

Cependant, il est indéniable que la version actuelle des MOOC ne satisfait pas pleinement toutes les promesses énoncées. Les objectifs d'apprentissage ne sont pas toujours atteints et ne correspondent pas toujours aux caractéristiques clés de l'enseignement traditionnel. Les interactions entre les étudiants et les enseignants, ainsi que celles entre les pairs, sont souvent moins riches dans les environnements en ligne par rapport aux environnements traditionnels. De plus, l'absence de cohérence entre les différents cours proposés rend difficile la certification d'un MOOC en tant que programme d'études complet et reconnu. En fin de compte, les apprenants ne sont pas encore pleinement adaptés à cette forme d'apprentissage qui exige beaucoup d'eux. Elle nécessite une maîtrise des outils informatiques ainsi qu'un engagement et une initiative plus importants. Cela explique en partie les taux de réussite relativement faibles observés jusqu'à présent. [22]

L'écart dans les résultats d'apprentissage entre les MOOC et l'enseignement Cependant, à mesure que de nouvelles technologies plus appropriées deviennent disponibles ainsi que de bonnes pratiques pédagogiques, les méthodes d'enseignement traditionnelles devraient être réduites. Basé sur des fondamentaux solides. [23] Attendu que La flexibilité opérationnelle démontrée par ce segment contraste fortement avec celle La rigidité est répandue dans les établissements d'enseignement supérieur traditionnels.

- **Parmi les stratégies d'amélioration de l'apprentissage, on peut inclure :**

- 1. Le modèle cognitif de la récupération de l'information :** Le modèle cognitif de la récupération de l'information met en avant l'importance des tests fréquents dans le processus d'apprentissage, soulignant que ceux-ci ne servent pas seulement à évaluer la compréhension des étudiants mais ont également un impact positif sur leur apprentissage. En rappelant l'information de la mémoire à court terme, les tests favorisent une meilleure rétention à long terme des connaissances, surpassant ainsi d'autres méthodes comme l'utilisation de schémas conceptuels [24]. Les MOOCs intègrent cette approche en proposant des questionnaires, des tests à choix multiples, et des forums interactifs, facilitant ainsi la restitution d'informations et favorisant une meilleure compréhension des sujets abordés. [25]
- 2. Un apprentissage centré sur l'étudiant :** L'approche centrée sur l'étudiant, introduite depuis 2009 avec le communiqué de Leuven/Louvain-la-Neuve et intégrée au processus de Bologne, met l'accent sur la personnalisation de la formation pour répondre aux besoins individuels de chaque apprenant. Les plateformes MOOCs ont le potentiel de favoriser cette approche en offrant une expérience d'apprentissage plus personnalisée, adaptée aux spécificités cognitives des étudiants [26]. Les MOOCs adoptent une approche pédagogique axée sur la maîtrise, utilisant des quizzes et des programmes d'intelligence artificielle pour optimiser l'apprentissage. Les étudiants avancent seulement lorsqu'ils maîtrisent les concepts précédents. De plus, les MOOCs s'adaptent aux besoins individuels des apprenants et à leur évolution cognitive en offrant une variété de ressources, notamment des vidéos de

cours à vitesse réglable et des formats d'apprentissage multiples. Cette approche diversifiée favorise une compréhension approfondie des sujets, bénéfique pour les étudiants habitués à gérer plusieurs tâches simultanément [27]

Un outil d'expérimentation et d'amélioration de l'apprentissage : Les données recueillies sur les plateformes MOOCs offrent des perspectives d'innovation significatives, comprenant les résultats des quizzes, les tests finaux, ainsi que l'utilisation et les modèles d'interaction avec les outils disponibles. Ces informations sont cruciales pour comprendre l'évolution de l'apprentissage à grande échelle et guider le développement de nouvelles méthodes pédagogiques. [28] Vu le nombre important d'étudiants actifs Les plateformes MOOCs, offrent un terrain propice à l'expérimentation en répartissant aléatoirement les participants pour comparer l'efficacité des approches pédagogiques. Cette méthode d'expérimentation en situation réelle, inspirée des recherches d'Esther Duflo, John List et Roland Fryer en économie, permet de mieux évaluer les impacts des différentes approches pédagogiques, contournant ainsi les limites des analyses empiriques classiques dans le domaine de l'éducation. [29]

- 3. Une pédagogie hybride :** Dans le contexte de l'enseignement combinant l'apprentissage en ligne et en classe, l'enseignant joue un rôle de facilitateur dans le processus d'apprentissage des étudiants, ce qui nécessite des compétences pédagogiques spécifiques et une maîtrise des outils informatiques. Cette approche vise à fusionner les avantages des deux méthodes d'enseignement, mais elle peut également représenter un investissement en termes de temps et de ressources financières pour les enseignants [30].

1.2.4.3.6. Les modèles d'affaires des plateformes MOOCs :

1.2.4.3.6.1. Un modèle d'affaires, quésaco ? dans Les MOOCs:

Le modèle économique des MOOC (Massive Open Online Courses) est une structure économique qui définit la manière dont les plateformes d'apprentissage en ligne génèrent des revenus et gèrent les coûts. Il s'agit de monétiser les services par le biais d'abonnements payants, de partenariats institutionnels et de la vente de certifications. Les MOOC doivent également gérer les dépenses liées à la création de contenu, à la maintenance de la plateforme et au marketing. Dans le domaine de l'informatique, il s'agit de tirer parti des technologies pour optimiser les parcours d'apprentissage et offrir une meilleure expérience à l'utilisateur.

Ces plateformes sont conçues pour faciliter l'interaction entre plusieurs groupes distincts d'utilisateurs, qui sont les principaux acteurs de la plateforme. En l'absence de celle-ci, ces interactions pourraient être difficiles voire impossibles à réaliser. Chaque groupe d'utilisateurs valorise les services de la plateforme en fonction de l'engagement des autres groupes. Ainsi, la plateforme agit comme un intermédiaire essentiel qui crée de la valeur. Le défi économique réside dans la capacité à capturer efficacement cette valeur créée par l'interaction entre les différents acteurs.31

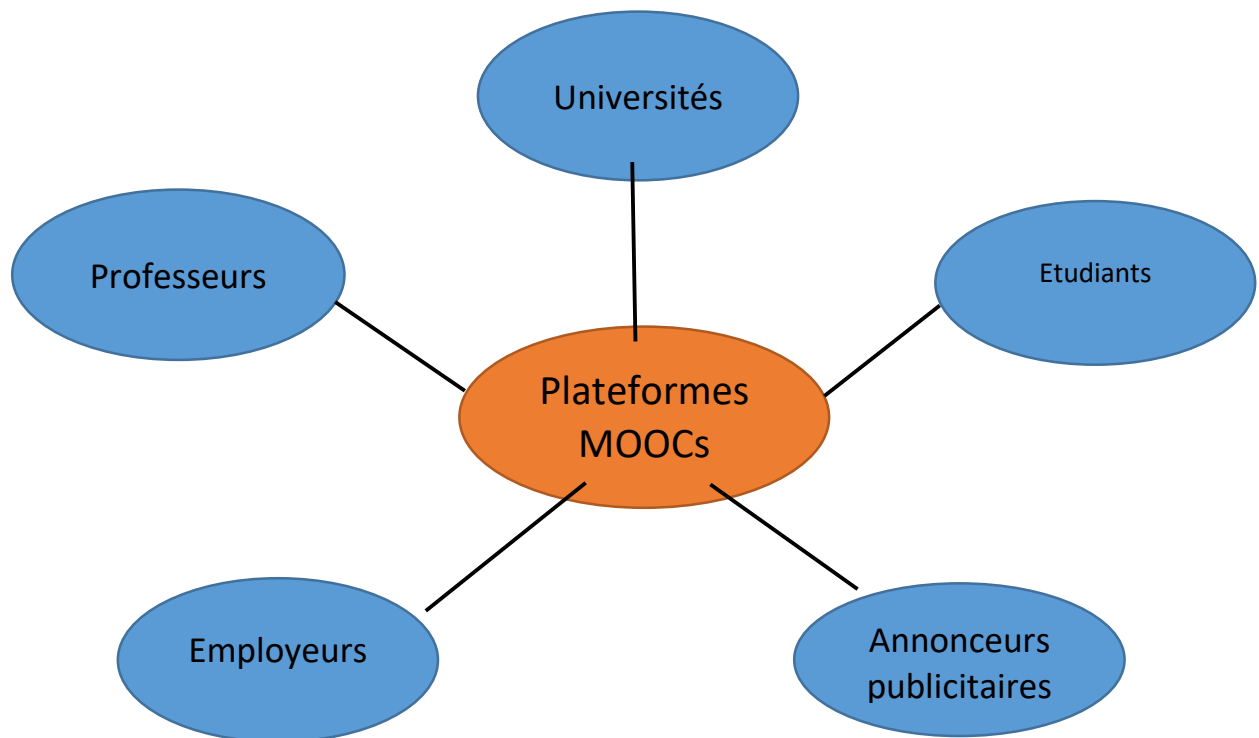


Figure 1.1 : les cinq groupes d'agents qui sont susceptibles de graviter autour d'une plateforme MOOCs

Explication de la figure : La figure 1 illustre les cinq groupes d'agents qui peuvent être impliqués autour d'une plateforme de MOOCs : en facilitant l'interaction entre les étudiants, les professeurs et les universités, ces plateformes ont également la capacité de connecter ces trois groupes aux employeurs et aux annonceurs. Pour comprendre les fonctions assurées par la plateforme (et comment elle pourrait monétiser ces fonctions), il est crucial d'identifier les attentes mutuelles entre chaque groupe et ce qu'ils peuvent mutuellement apporter en retour.

Maintenant que nous avons exploré comment une plateforme MOOCs génère de la valeur en facilitant l'interaction et les bénéfices mutuels des différents groupes d'acteurs, nous passons en revue cinq modèles d'affaires. Ces modèles décrivent les différentes façons dont une plateforme MOOCs peut essayer de tirer profit de la valeur qu'elle crée. Nous nous posons également des questions sur la durabilité de ces modèles.

1. Le modèle de certification :

Le modèle de certification des MOOC vise à reproduire le système éducatif traditionnel en offrant des certificats ou des diplômes à l'issue des cours. Cependant, les MOOC sont confrontés à des défis dans ce modèle en raison des taux élevés d'abandon et de la crédibilité des certifications. Contrairement aux organismes de certification spécialisés, les MOOC se concentrent principalement sur l'enseignement plutôt que sur la certification. L'absence d'accréditation et de contrôle de la qualité fait douter les employeurs de la valeur des certifications des MOOC. Pour résoudre ces problèmes et créer de la valeur, un système d'accréditation rigoureux basé sur une évaluation continue de la qualité est nécessaire. [32]

2. Le modèle publicitaire :

Le modèle publicitaire est largement utilisé sur l'internet comme forme de financement. Au lieu de vendre du contenu aux consommateurs, ce modèle cherche à vendre l'accès à ces consommateurs aux annonceurs. Les annonceurs cherchent à capter l'attention des utilisateurs et sont plus satisfaits s'ils disposent d'informations sur leur public cible. Les plateformes de MOOC sont attrayantes pour les annonceurs en raison de la richesse des données qu'elles peuvent collecter sur leurs utilisateurs, ainsi que du temps que les apprenants passent sur la plateforme, ce qui offre des possibilités d'affichage publicitaire. Cependant, il est essentiel d'évaluer comment ces publicités peuvent affecter l'expérience d'apprentissage et si elles peuvent décourager les étudiants et les enseignants. [33]

3. Le modèle de partenariat :

Le modèle de partenariat dans les MOOCs se réfère à l'intégration des plateformes de cours en ligne ouverts et massifs avec les institutions d'enseignement traditionnel via une approche d'apprentissage mixte. Cette intégration repose sur la collaboration des universités avec des partenaires externes et leur capacité à financer ces plateformes de manière viable. Les MOOCs peuvent servir de substituts ou de compléments au système éducatif existant, visant à améliorer la qualité de l'enseignement, à réduire les frais de scolarité, à gérer les contraintes de capacité et à enrichir les programmes universitaires, bien que cela ne réduise pas nécessairement les coûts éducatifs globaux.[34]

1.2.4.5.7. Les principales plateformes :

Le MOOC	Definition
Alison	Fondée en Irlande en 2007, Alison est aujourd'hui l'une des plus grandes plateformes gratuites d'apprentissage au monde pour l'éducation et la formation aux compétences. Elle propose plus de 4 000 cours gratuits et a bénéficié à plus de 30 millions d'apprenants dans près de 200 pays. Alison, une entreprise à but lucratif, génère des revenus grâce à la publicité, la vente de marchandises, ainsi qu'à la vente de certificats et de diplômes pour les diplômés qui choisissent de les acheter.[35]
Udemy	Est l'outil parfait pour acquérir des compétences pratiques dans un domaine spécifique. Udemy propose actuellement environ 20 000 cours. La principale différence ici est que n'importe qui peut créer son propre cours. Cela signifie que la qualité des cours peut varier considérablement. C'est également la raison pour laquelle vous n'obtiendrez pas de diplôme. Le prix dépend du professeur. Assurez-vous de consulter les avis sur ce cours particulier avant de payer quoi que ce soit.[36]
Canvas	Contrairement à d'autres solutions telles que EdX et Coursera, Canvas n'est pas limité aux universités prestigieuses. Il propose des

Chapitre 1 Etat de l'art

	cours issus d'une sélection beaucoup plus large d'universités moins connues. Cela rend les cours quelque peu plus accessibles et moins intimidants si vous souhaitez vous aventurer dans un domaine que vous ne connaissez pas du tout. [37]
FutureLearn	Est une plateforme d'apprentissage mondiale ayant pour mission de transformer l'accès à l'éducation en proposant des cours en ligne de premier plan provenant des universités et des marques les plus réputées au monde. Des diplômes en ligne, des micro-certifications aux cours courts, nous proposons une éducation de classe mondiale accréditée et non accréditée, entièrement en ligne, à la demande et sociale. Nos cours permettent aux apprenants de débloquer de nouvelles voies professionnelles hautement qualifiées et de se donner les moyens de changer le monde.[w5]
Khan Academy	Khan Academy est une plateforme d'apprentissage personnalisé adaptée à tous les âges, couvrant une vaste gamme de sujets tels que les mathématiques, les sciences, la programmation informatique, l'histoire, l'histoire de l'art, l'économie, et bien plus encore. Elle propose des exercices interactifs, des vidéos pédagogiques, ainsi qu'un tableau de bord personnalisé qui permet aux apprenants d'étudier à leur propre rythme, à l'intérieur ou à l'extérieur de la salle de classe, et ce, dans plus de 40 langues.[w4]
Coursera	Coursera est une plateforme d'apprentissage en ligne proposant des cours, des diplômes, des programmes de certificats et des tutoriels dans un large éventail de sujets. Plus de 300 universités et entreprises de premier plan dispensent des cours, notamment Stanford, Duke, Google et IBM. Les expériences d'apprentissage vont de projets pratiques ciblés à des certificats et des diplômes complets prêts à l'emploi [38]
edX	edX est une plateforme d'apprentissage en ligne proposant des cours et des programmes de premier plan provenant de grandes universités mondiales. Fondée en 2012 par Harvard et le MIT, elle vise à rendre l'éducation accessible à tous. Les apprenants peuvent suivre des cours interactifs, participer à des discussions et obtenir des certifications. edX offre une diversité de sujets, de la technologie aux sciences humaines.[39]

Tableau 1 : Les principales plateformes des MOOCs

1.2.5. Le principe de E-Learning :

1.2.5.1.Principes de contiguïté :

Ce principe exploite les mots et les graphiques correspondants, présentant les deux éléments en étroite proximité l'un de l'autre, et est divisé en deux sous-catégories : la contiguïté spatiale et temporelle.

- Principe de contiguïté spatiale Ce principe repose sur la relativité spatiale, en maintenant les mots expliquant un concept et leurs images associées ensemble, facilitant ainsi la capacité de l'apprenant à comprendre pleinement et à traiter l'information.
- Principe de contiguïté temporelle Inversement, ce principe présente des mots et des images correspondantes simultanément, plutôt que successivement. Les deux éléments visuels, mots et images, peuvent ensuite être traités simultanément pour faciliter la rétention et la compréhension. [40]

1.2.5.2. Principe du multimédia :

Ce principe utilise simultanément des mots et des graphiques plutôt que des mots exclusivement. Cette approche mobilise à la fois des éléments visuels et auditifs, car les apprenants ont souvent une préférence pour l'apprentissage à partir de mots et d'images plutôt que de mots seuls. Avec les nouvelles innovations telles que la modélisation et l'animation 3D, la réalité virtuelle et la réalité augmentée, les cours prennent vie comme jamais auparavant, créant un impact maximal. Ce principe est économique, impactant, et les combinaisons possibles sont pratiquement illimitées, en faisant un choix facile. [41]

1.2.5.3.Principe de Modalité :

Le principe de Modalité préconise l'utilisation de la narration audio plutôt que du texte à l'écran pour présenter les mots et les informations. L'objectif est d'éviter la surcharge cognitive en permettant aux apprenants de se concentrer sur la narration tout en décrivant également les images. Ce principe, en accord avec la théorie de la charge cognitive, favorise une meilleure compréhension en intégrant les informations par le biais de la narration. La narration, particulièrement puissante lorsqu'elle est utilisée comme témoignage réel, est mise en avant dans certains modules d'apprentissage en ligne pour prévenir des situations telles que les agressions sexuelles, la violence dans les relations, et le harcèlement sur les campus universitaires. [42]

1.2.5.4.Principe de Cohérence

Similaire au Principe de Redondance, celui-ci adopte également une approche du moins-est-plus. Les informations non pertinentes, superflues ou inapplicables sont éliminées, y compris l'audio, les visuels et les mots, pour éviter la distraction et augmenter l'apprentissage et la rétention, permettant à l'apprenant de se concentrer uniquement sur les éléments essentiels. En d'autres termes : si des mots, de l'audio ou des images peuvent être omis sans affecter l'impact du message ou de l'information, omettez-les. [42] [43]

1.2.5.5.Principe de la Voix :

Ce principe suggère que les personnes apprennent mieux lorsque la narration est prononcée d'une voix humaine intime, informelle et amicale plutôt que d'une voix robotique. Souvent, les voix artificielles sont plus économiques, mais à long terme, la rétention et l'engagement en souffrent. En fait, SafeSchools, par exemple, a remporté un prix du récit, soulignant à quel point les cours peuvent efficacement transmettre des informations en créant des scénarios auxquels vos apprenants pourront s'identifier, favorisant ainsi une connexion émotionnelle avec le contenu. [44]

1.2.5.6. Principe de Redondance :

Ce principe explique les éléments visuels avec des mots dans l'audio ou le texte, mais pas les deux en même temps. Ainsi, les deux éléments se complètent plutôt que de perturber l'apprenant avec une surcharge de tous les éléments disponibles, en adoptant une approche du moins-est-plus. Un exemple de l'application de ce principe est lorsque les apprenants ont le pouvoir de choisir s'ils veulent écouter l'audio ou lire le texte. Les sous-titres peuvent être activés ou désactivés, et l'audio peut être coupé lorsque les apprenants se trouvent dans un espace partagé. [45]

1.3. Les systèmes de recommandations dans le E-Learning :

1.3.1. Introduction

Les systèmes de recommandation dans le E-learning jouent un rôle crucial en aidant les apprenants à naviguer dans l'immense volume de ressources éducatives disponibles en ligne. Contrairement aux systèmes de recommandation utilisés dans le commerce électronique, qui cherchent principalement à maximiser les profits, les systèmes de recommandation en E-learning visent à améliorer l'expérience d'apprentissage en proposant des contenus adaptés aux besoins individuels des utilisateurs. Ces systèmes exploitent des algorithmes avancés, tels que le filtrage collaboratif, pour personnaliser les recommandations en fonction des préférences et des comportements d'apprentissage des utilisateurs. En intégrant ces systèmes dans les plateformes éducatives, on peut offrir un apprentissage plus efficace et personnalisé, répondant aux styles d'apprentissage variés et favorisant une meilleure rétention des connaissances.

1.3.2. Définition du système de recommandation :

Les systèmes de recommandation en E-learning sont des outils technologiques conçus pour personnaliser l'expérience d'apprentissage en ligne en suggérant des contenus et des ressources éducatives adaptés aux besoins et aux préférences des apprenants. Utilisant des algorithmes tels que le filtrage collaboratif, ces systèmes analysent les comportements et les interactions des utilisateurs pour fournir des recommandations pertinentes, visant à améliorer l'efficacité et la personnalisation de l'apprentissage. [46]

1.3.3. Les types systèmes de recommandation dans le E-Learning :

1.3.3.1.Recommandation basée sur le contenu :

La recommandation basée sur le contenu est une méthode de recommandation dans laquelle des systèmes analysent les caractéristiques intrinsèques des items eux-mêmes pour suggérer des éléments similaires à ceux que l'utilisateur a déjà appréciés. Cette approche repose sur l'idée que les items partageant des caractéristiques similaires avec ceux que l'utilisateur a aimés sont susceptibles de lui plaire également. Voici quelques détails sur ce type :

- **Création de profils d'items :**

Les systèmes de recommandation basés sur le contenu commencent par créer des profils détaillés pour chaque item disponible. Ces profils peuvent inclure des attributs tels que des mots-clés, des tags, des métadonnées structurées, ou même des représentations vectorielles obtenues à partir du contenu textuel, audiovisuel ou interactif des items.

- **Construction de profils d'utilisateurs :**

En parallèle, les systèmes élaborent des profils pour chaque utilisateur en fonction de leurs interactions passées avec les items. Cela peut inclure les items qu'ils ont consultés, notés, marqués comme favoris, ou consommés d'une autre manière sur la plateforme d'E-learning.

- **Calcul de similarité :**

Une fois que les profils d'items et d'utilisateurs sont établis, le système calcule la similarité entre le profil de l'utilisateur et ceux des items disponibles. Cette similarité peut être mesurée à l'aide

de diverses techniques comme la similarité cosinus pour les représentations vectorielles, ou d'autres mesures basées sur des attributs spécifiques des items.

- **Recommandation :**

Enfin, le système utilise la similarité calculée pour recommander les items les plus pertinents à l'utilisateur.]Les items recommandés sont ceux dont le profil correspond le mieux aux préférences et aux comportements d'interaction passés de l'utilisateur.[47]

1.3.3.2.Filtrage collaboratif (CF)

Les systèmes de recommandation basés sur le filtrage collaboratif analysent les préférences des utilisateurs pour recommander des éléments similaires à ceux appréciés par d'autres utilisateurs ayant des comportements et des intérêts similaires. Contrairement aux approches basées sur le contenu qui se concentrent sur les caractéristiques intrinsèques des items (comme le texte, les tags ou les métadonnées), le CF ne prend pas en compte les détails spécifiques des items eux-mêmes, mais plutôt les interactions passées des utilisateurs avec ces items.

Voici un travail sur le filtrage collaborative dans le E-Learning :

Travail:

- Titre: Item-Based Collaborative Filtering Recommendation Algorithms
- Auteurs: Sarwar, B., Karypis, G., Konstan, J., Riedl, J. Conférence:
- Proceedings of the 10th International Conference on World Wide Web (WWW '01)
Année: 2001 Pages: 285-295 Éditeur : ACM

Ce travail représente une avancée significative dans le domaine du filtrage collaboratif appliqué à l'E-learning, offrant une méthode efficace pour personnaliser l'expérience d'apprentissage en recommandant des items basés sur la similarité avec les préférences des utilisateurs. [48]

1.3.3.3. Recommandation basée sur les connaissances :

La recommandation basée sur les connaissances est une approche des systèmes de recommandation dans le domaine de l'éducation en ligne qui se distingue par l'utilisation d'ontologies ou de bases de connaissances. Ces systèmes visent à recommander des éléments éducatifs en analysant de manière approfondie le contenu et en prenant en compte le contexte éducatif spécifique à chaque utilisateur.

Fonctionnement :

Utilisation d'ontologies : Les ontologies sont des structures de données sémantiques qui modélisent les connaissances dans un domaine particulier. Dans le contexte de la recommandation basée sur les connaissances, les ontologies sont utilisées pour représenter et organiser les concepts éducatifs, les relations entre eux, ainsi que les préférences et les profils des utilisateurs.

Compréhension approfondie du contenu : Contrairement aux approches basées sur le contenu qui se concentrent sur les caractéristiques intrinsèques des éléments (par exemple, mots-clés, métadonnées), la recommandation basée sur les connaissances analyse le contenu

Chapitre 1 Etat de l'art

éducatif de manière plus complexe. Cela inclut la sémantique des concepts, les relations hiérarchiques et les dépendances contextuelles pour comprendre la pertinence d'un élément éducatif pour un utilisateur spécifique.

Contexte éducatif spécifique : Chaque utilisateur a des besoins et des objectifs d'apprentissage uniques. Les systèmes de recommandation basés sur les connaissances prennent en considération ce contexte éducatif spécifique en examinant les parcours d'apprentissage antérieurs, les compétences déjà acquises, les préférences d'apprentissage et les objectifs futurs des utilisateurs.

Exemple de travail connexe :

Article : Stamatis, D., & Choriantopoulos, K. (2016). Ontology-driven personalized e-learning recommendations using semantic similarity. *Journal of Educational Technology & Society*, 19(2), 65-77.

Dans cet article, les auteurs ont développé un système de recommandation personnalisée pour l'E-learning qui utilise la similarité sémantique et les ontologies. Ce système montre comment l'exploitation des ontologies peut améliorer la précision et la pertinence des recommandations en prenant en compte les relations sémantiques entre les concepts éducatifs et les préférences des apprenants. [49]

1.3.3.4.Recommandation hybride :

Les systèmes de recommandation hybrides dans l'E-learning intègrent diverses approches, telles que le filtrage collaboratif, le filtrage basé sur le contenu, et parfois d'autres méthodes comme le filtrage démographique ou la recommandation contextuelle. L'objectif est de tirer parti des forces de chaque méthode tout en atténuant leurs faiblesses respectives. Voici les principales caractéristiques et bénéfiques de la recommandation hybride :

Combinaison des approches : Les systèmes hybrides intègrent le filtrage collaboratif, qui se base sur les préférences des utilisateurs similaires, avec le filtrage basé sur le contenu, qui analyse les caractéristiques intrinsèques des éléments (cours, ressources, etc.). Par exemple, en utilisant le filtrage collaboratif, le système identifie des utilisateurs similaires à l'apprenant pour recommander des éléments populaires parmi ces utilisateurs. En parallèle, le filtrage basé sur le contenu recommande des éléments similaires à ceux que l'apprenant a déjà appréciés en analysant les caractéristiques des éléments.

Amélioration de la pertinence : En combinant différentes méthodes, les systèmes hybrides visent à améliorer la précision des recommandations. Par exemple, lorsque le filtrage collaboratif peut être biaisé par des profils d'utilisateurs incomplets ou peu similaires, le filtrage basé sur le contenu peut compenser en recommandant des éléments pertinents en fonction des caractéristiques spécifiques de l'utilisateur.

Adaptabilité et personnalisation : La combinaison de méthodes permet une meilleure adaptation aux préférences individuelles et aux contextes changeants des utilisateurs. Les systèmes hybrides peuvent ajuster les recommandations en temps réel en fonction des interactions de l'utilisateur avec la plateforme d'apprentissage et de l'évolution de ses besoins éducatifs.

Performance globale : En intégrant plusieurs approches, les systèmes hybrides peuvent améliorer la couverture (variété des recommandations), la précision et la satisfaction globale des utilisateurs par rapport aux approches individuelles.

➤ Exemple de travail connexe :

Article: A hybrid recommender system for personalized e-learning environments. Computers in Human Behavior.

Cet article explore le développement d'un système de recommandation hybride spécifiquement conçu pour les environnements d'apprentissage en ligne personnalisés. Les auteurs détaillent comment l'intégration de différentes techniques de recommandation a été utilisée pour améliorer l'expérience d'apprentissage des utilisateurs en optimisant la pertinence et la performance des recommandations.

En utilisant ce type de système, les plateformes d'apprentissage peuvent non seulement mieux répondre aux besoins individuels des apprenants, mais aussi augmenter l'engagement et l'efficacité des sessions d'apprentissage en ligne. [50]

1.3.4. Travaux connexes sur les systèmes de recommandation dans le domaine de l'apprentissage en ligne

1.3.4.1. Systèmes de recommandation pour l'éducation personnalisée :

Les travaux sur les systèmes de recommandation pour l'éducation personnalisée s'efforcent de créer des environnements d'apprentissage en ligne qui répondent de manière proactive aux besoins uniques de chaque apprenant, améliorant ainsi leur expérience globale et leurs résultats d'apprentissage.

Par exemple, un système de recommandation pour l'éducation personnalisée pourrait recommander des cours, des modules ou des ressources d'apprentissage spécifiques en fonction des intérêts déclarés de l'apprenant, de ses performances antérieures, et des contenus qui ont été efficaces pour des apprenants similaires. De cette manière, ces systèmes visent à optimiser l'engagement, la rétention et la réussite des apprenants en leur fournissant un contenu adapté à leurs besoins individuels. [51]

1.3.4.2. Analyse des données d'apprentissage

L'analyse des données d'apprentissage consiste à examiner et à interpréter les informations générées par les plateformes d'apprentissage en ligne. Ces données incluent souvent les traces laissées par les apprenants lors de leur navigation sur la plateforme, leurs interactions avec le contenu pédagogique, leurs résultats d'évaluation, ainsi que d'autres métriques pertinentes telles que le temps passé sur chaque activité. L'objectif principal de cette analyse est de mieux comprendre le comportement des apprenants afin de pouvoir fournir des recommandations pertinentes et des interventions personnalisées. [52]

1.3.4.3. Systèmes de recommandation adaptatifs

Les systèmes de recommandation adaptatifs visent à offrir une expérience d'apprentissage en ligne personnalisée et dynamique en s'adaptant aux évolutions des préférences et des besoins des apprenants au fil du temps. Ces systèmes utilisent des techniques d'intelligence artificielle

pour surveiller en continu le comportement des apprenants et ajuster les recommandations en fonction de leurs interactions passées avec le contenu pédagogique et les résultats d'évaluation. En anticipant les besoins futurs des apprenants, ces systèmes proposent activement du contenu pertinent, ce qui peut améliorer l'engagement et la réussite des apprenants en ligne. [53]

1.3.4.4. Évaluation des systèmes de recommandation dans l'éducation

Les travaux sur l'évaluation des systèmes de recommandation dans l'éducation mesurent l'efficacité de ces systèmes en termes de satisfaction des apprenants et de rétention. Les principales métriques utilisées incluent :

- **Précision** : Proportion de recommandations pertinentes. Rappel : Proportion d'éléments pertinents recommandés.
- Taux de satisfaction des apprenants : Mesuré par des sondages ou des questionnaires.
- **Taux de rétention** : Proportion d'apprenants qui continuent à utiliser la plateforme après les recommandations.
- **Diversité des recommandations** : Variété des recommandations fournies.

Ces métriques permettent d'évaluer et d'optimiser les systèmes de recommandation pour améliorer l'expérience d'apprentissage en ligne. [54]

1.4. L'Elicitation de préférences des apprenants dans les MOOCs

1.4.1. Introduction

Les plateformes d'apprentissage en ligne, en particulier les MOOC, sont devenues des outils essentiels dans le domaine de l'éducation, offrant un accès facile à une variété de cours et de ressources. Cependant, cette diversité de contenu s'accompagne du défi de personnaliser l'expérience d'apprentissage pour chaque utilisateur. Tout comme dans les systèmes de recommandation sociale, où la compréhension des préférences de l'utilisateur est cruciale pour fournir des recommandations pertinentes, dans les MOOC, l'obtention des préférences de l'apprenant joue un rôle essentiel. Pour que les MOOC soient efficaces et apportent une valeur ajoutée à chaque apprenant, il est essentiel de comprendre ses intérêts, ses préférences d'apprentissage et ses besoins spécifiques.

L'identification de ces préférences permet aux plateformes d'adapter le contenu des cours, les méthodes d'enseignement et les interactions avec les apprenants, améliorant ainsi l'engagement et les résultats de l'apprentissage. Dans cette étude, nous explorerons l'importance de l'obtention des préférences des apprenants dans les MOOCs en examinant les différentes méthodes et techniques utilisées pour recueillir ces informations.

Nous chercherons à comprendre comment ces préférences peuvent être utilisées pour personnaliser l'expérience d'apprentissage et améliorer l'efficacité des MOOCs.

Enfin, nous proposerons des recommandations pour intégrer efficacement l'élicitation des préférences de l'apprenant dans la conception et la diffusion des cours en ligne, afin de maximiser les avantages pour les apprenants et les formateurs.

1.4.2. Définition :

L'élicitation des préférences dans les MOOCs est un processus essentiel visant à comprendre les besoins individuels des apprenants pour améliorer leur expérience d'apprentissage en ligne. Ce processus commence par la collecte de données sur les préférences des apprenants, qui peut être effectuée à l'aide de diverses méthodes telles que les questionnaires, les entretiens et l'analyse des données d'utilisation de la plateforme. Une fois les données collectées, elles sont ensuite analysées pour identifier les tendances et les schémas qui peuvent indiquer les préférences des apprenants en matière de contenu, de méthodes d'apprentissage et d'interactions. Sur la base de ces informations, les plateformes MOOC peuvent personnaliser l'expérience d'apprentissage de chaque apprenant en adaptant le contenu des cours, en proposant des activités d'apprentissage pertinentes et en fournissant un soutien individualisé. En conséquence, cette personnalisation améliore l'engagement et la rétention des apprenants, car ils sont plus susceptibles de rester motivés et impliqués dans leurs études lorsqu'ils ont accès à un contenu qui correspond à leurs intérêts et à leurs besoins spécifiques. En résumé, l'élicitation des préférences des apprenants dans les MOOCs est un processus crucial pour optimiser l'efficacité et l'impact de l'apprentissage en ligne en rendant l'expérience plus personnalisée et adaptée à chaque apprenant.

1.4.3. Les méthodes utilisées dans l'élicitation des préférences

1.4.3.1. Le feedback des pairs

Le feedback des pairs dans les MOOCs permet aux apprenants d'évaluer et de donner des retours sur le travail de leurs pairs. Cette méthode encourage les interactions sociales et permet aux apprenants de recevoir des perspectives diverses sur leurs travaux. Les préférences des apprenants peuvent être déduites en analysant comment ils réagissent aux commentaires reçus, qu'ils soient positifs, constructifs ou critiques. Le feedback des pairs aide également à améliorer les compétences d'évaluation et de réflexion critique des apprenants, tout en favorisant un environnement d'apprentissage collaboratif. [55]

1.4.3.2. Suivi des réponses aux quiz et aux évaluations :

Le suivi des réponses aux quiz et aux évaluations consiste à analyser les réponses et les performances des apprenants dans différents types de quiz et d'évaluations au sein des MOOCs. Cette méthode permet de comprendre les préférences des apprenants en identifiant les types de questions auxquelles ils répondent le mieux, les sujets qui suscitent leur intérêt, ou encore les méthodes d'évaluation qu'ils trouvent les plus efficaces ou les plus équitables. En examinant ces données, les concepteurs de cours et les instructeurs peuvent ajuster le contenu et les modalités d'évaluation pour mieux répondre aux besoins et aux préférences des apprenants, améliorant ainsi leur expérience d'apprentissage. Cette méthode est particulièrement utile pour personnaliser les parcours d'apprentissage et optimiser les interactions avec le contenu pédagogique, en tenant compte des retours directs des apprenants à travers leurs performances évaluatives. [56]

1.4.3.3. Analyse de l'activité en ligne:

L'analyse de l'activité en ligne est une méthode d'élicitation des préférences des apprenants dans les MOOCs qui repose sur l'examen des traces numériques générées par leur interaction avec les plateformes d'apprentissage en ligne. Cette méthode consiste à collecter et analyser des données telles que les clics sur les ressources, le temps passé sur chaque activité, les interactions avec le contenu pédagogique, les contributions aux forums de discussion, et d'autres métriques comportementales. En utilisant des techniques d'analyse de données et d'apprentissage automatique, les concepteurs de cours et les chercheurs peuvent identifier des schémas significatifs dans le comportement des apprenants. Cela permet de comprendre leurs préférences quant aux types de contenu, aux méthodes d'enseignement, aux formats d'évaluation, et aux modalités d'interaction qui influencent leur engagement et leur réussite dans le MOOC. [57]

1.4.3.4. Le profilage des utilisateurs :

Dans les systèmes de recommandation pour l'apprentissage en ligne le profilage des utilisateurs implique la création de profils individuels par la collecte et l'analyse de données sur les caractéristiques des utilisateurs, leurs préférences, leur comportement passé et leurs performances d'apprentissage. Ces données sont obtenues à partir de sources telles que des questionnaires, l'historique de navigation et les interactions avec le contenu des cours. Les techniques d'analyse des données, y compris les algorithmes d'apprentissage automatique, permettent d'identifier des modèles afin de déterminer les préférences des utilisateurs. Une fois les profils établis, les systèmes de recommandation peuvent adapter le contenu et les recommandations

d'activités aux intérêts, aux compétences, aux objectifs d'apprentissage et aux préférences de chaque utilisateur. Par exemple, un utilisateur intéressé par la programmation informatique peut recevoir des recommandations spécifiques pour des cours ou des projets dans ce domaine. Les profils des utilisateurs sont continuellement mis à jour en fonction des nouvelles données collectées et des interactions ultérieures, ce qui garantit que les recommandations sont continuellement personnalisées en fonction de l'évolution des intérêts et des besoins de l'utilisateur. [58]

1.4.3.5. La rétroaction implicite et explicite :

La rétroaction implicite et explicite sont deux formes de retour d'information fournies par les utilisateurs dans les systèmes de recommandation et d'apprentissage en ligne. **Rétroaction implicite :** Il s'agit de données recueillies de manière indirecte à partir du comportement des utilisateurs lors de leur interaction avec la plateforme. Ces données peuvent inclure des actions telles que le temps passé sur une page, les clics sur des liens, les mouvements de la souris, les évaluations de contenu, etc. La rétroaction implicite est souvent utilisée pour mesurer l'engagement des utilisateurs et pour déduire leurs préférences et leurs intérêts à partir de leurs actions. **Rétroaction explicite :** Contrairement à la rétroaction implicite, la rétroaction explicite est fournie directement par les utilisateurs sous forme de retours volontaires ou délibérés. Cela peut inclure des évaluations de contenu, des commentaires, des réponses à des questionnaires, des annotations sur des éléments de cours, des préférences de contenu déclarées, etc. La rétroaction explicite fournit des informations directes sur les préférences, les opinions et les besoins des utilisateurs. [59]

1.4.3.6. Les enquêtes et les questionnaires :

Sont des outils essentiels dans les systèmes de recommandation et d'apprentissage en ligne pour collecter des données sur les préférences, les opinions et les besoins des utilisateurs. [34] Les enquêtes impliquent généralement des questions structurées administrées de manière formelle ou informelle, tandis que les questionnaires se concentrent souvent sur des sujets spécifiques de l'expérience utilisateur. Les réponses à ces outils fournissent des données quantitatives et qualitatives précieuses pour comprendre les tendances et les comportements des utilisateurs, ce qui peut être utilisé pour améliorer les recommandations de contenu et l'expérience globale de l'utilisateur. [60]

1.4.4. Les Difficultés et les Problèmes de l'Élicitation des Préférences des apprenants dans les moocs

L'élicitation des préférences dans les MOOCs présente plusieurs difficultés et problèmes, on cite :

1.4.4.1. Problème de démarrage à froid :

Le problème du démarrage à froid dans les systèmes de recommandation E-Learning se produit lorsque de nouveaux apprenants s'inscrivent, ce qui entraîne des difficultés pour fournir des recommandations précises et personnalisées en raison de l'absence d'historique d'interaction. Il est essentiel de résoudre ce problème pour améliorer l'expérience d'accueil et garantir un contenu pertinent aux nouveaux utilisateurs. [61]

1.4.4.2.Le problème de la nouveauté :

Il manifeste lorsque les systèmes de recommandation ont une propension à suggérer régulièrement des contenus qui ressemblent à ceux que l'utilisateur a déjà explorés. Cette répétition limite alors la possibilité pour l'utilisateur de découvrir de nouvelles idées, concepts ou domaines d'apprentissage qui pourraient lui être bénéfiques. En d'autres termes, cela restreint la diversité des recommandations et empêche l'utilisateur d'explorer des horizons d'apprentissage plus variés et enrichissants. [62]

1.4.4.3.Problème de sérendipité :

Le problème de la sérendipité se réfère au défi de fournir aux apprenants des moocs sérendipides ou inattendues. Ce problème survient lorsque les algorithmes du système se concentrent sur la prédiction du contenu en fonction des préférences passées de l'utilisateur, ce qui limite potentiellement l'introduction de suggestions diverses ou surprenantes. Une dépendance excessive à l'égard de l'historique de l'utilisateur peut conduire à un manque de variété dans les recommandations, empêchant la découverte de matériel pédagogique nouveau et non conventionnel qui pourrait enrichir l'expérience de l'apprenant. [63]

1.4.4.4.Problème de la quantité de données limitée :

Ce problème se pose lorsque l'utilisateur a interagi avec un nombre limité de ressources, ce qui entraîne des difficultés pour générer des recommandations précises et personnalisées. Les algorithmes du système s'appuient fortement sur l'historique des interactions de l'utilisateur pour formuler des prédictions, et le manque de données peut empêcher une compréhension approfondie des préférences de l'apprenant. [64]

1.4.4.5.Le problème de la qualité des données :

Le problème de la qualité des données se pose lorsque des erreurs, des biais ou une mauvaise qualité des données compromettent l'exactitude et la pertinence des recommandations. Les défauts des données, tels que les inexactitudes ou les biais, peuvent donner lieu à des suggestions erronées, ce qui a un impact négatif sur l'expérience d'apprentissage globale des utilisateurs. Pour résoudre ce problème, il faut mettre en œuvre des mesures rigoureuses d'assurance de la qualité des données, identifier et rectifier les erreurs, et établir des mécanismes pour améliorer la fiabilité des algorithmes de recommandation. L'objectif est de s'assurer que les recommandations fournies sont basées sur des données fiables et de haute qualité, ce qui améliore en fin de compte l'efficacité de la plateforme d'apprentissage en ligne. [65].

1.4.4.6. Problème de la Diversité :

Le problème de la diversité dans l'élicitation des préférences des apprenants dans les MOOCs consiste à équilibrer la pertinence et la variété des recommandations pour offrir une expérience d'apprentissage enrichissante. La diversité, dans ce contexte, fait référence à la variété et à la nouveauté des contenus recommandés aux apprenants. Un manque de diversité peut entraîner des recommandations trop spécifiques, limitant l'exposition des apprenants à de nouveaux sujets et restreignant leur apprentissage global. À l'inverse, une trop grande diversité peut diminuer la pertinence des recommandations, rendant celles-ci moins utiles pour répondre aux

besoins immédiats des apprenants. Les biais dans les données d'interaction des utilisateurs compliquent également la génération de recommandations à la fois variées et pertinentes, car ces biais peuvent refléter des préférences préexistantes plutôt que d'encourager la découverte de nouveaux domaines d'intérêt. [66]

1.4.5. L'élicitation des préférences à partir des commentaires

L'élicitation des préférences dans les systèmes de recommandation e-learning à partir des commentaires des utilisateurs est un processus précieux qui peut contribuer à améliorer la personnalisation de l'expérience d'apprentissage en ligne et à répondre aux besoins individuels des apprenants. En exploitant ces commentaires de manière efficace, les plateformes d'apprentissage en ligne peuvent offrir des recommandations plus pertinentes et améliorer ainsi l'engagement et la satisfaction des utilisateurs. [67]

Une approche pour obtenir des préférences à partir des commentaires dans les systèmes de recommandation e-learning consiste à utiliser des techniques de traitement du langage naturel (NLP) pour extraire des caractéristiques du texte. Ces techniques peuvent être appliquées pour analyser le contenu des commentaires laissés par les apprenants, tels que les critiques de cours, les retours d'expérience et les discussions sur les forums en ligne, afin d'extraire les préférences et les opinions des utilisateurs. Cette analyse permet au système de recommandation de comprendre le sens sémantique du texte, ce qui est essentiel pour saisir les nuances des préférences des utilisateurs et proposer des recommandations de contenu pertinentes et personnalisées.

Par exemple l'analyse des sentiments peut être appliquée pour extraire les émotions exprimées dans un commentaire, permettant de déterminer si un utilisateur apprécie ou non un élément spécifique. Ces informations sont ensuite utilisées comme entrée dans le système de recommandation, permettant ainsi une personnalisation plus fine des recommandations de contenu en fonction des préférences individuelles des utilisateurs. [68]

1.4.6. Travaux existants sur l'élicitation des préférences à partir des commentaires :

Dans le domaine de l'élicitation des préférences à partir des commentaires plusieurs travaux récents ont été réalisés pour comprendre les choix et les préférences des utilisateurs, Voici quelques exemples de travaux récents sur l'élicitation des préférences à partir des commentaires dans le tableau 2 :

TITRE DU TRAVAIL	Tache	Approche
User Preference Elicitation from User-Generated Content in Online Learning Platforms. [69]	Utilisation de l'analyse des commentaires des utilisateurs pour extraire leurs préférences.	Élicitation des préférences des utilisateurs à partir du contenu généré par les utilisateurs dans les plateformes d'apprentissage en ligne.
Preference Elicitation from Student Feedback for Personalized Course Recommendation in MOOCs.[70]	Utilisation des retours d'expérience des étudiants pour la recommandation personnalisée de cours.	Élicitation des préférences des étudiants à partir de leurs commentaires dans les Massive Open Online Courses (MOOCs) pour proposer des

Chapitre 1 Etat de l'art

		recommandations de cours personnalisées.
A Hybrid Approach for Preference Elicitation in E-learning Recommender Systems. [71]	Approche hybride combinant différentes techniques pour l'élicitation des préférences.	Élicitation des préférences des utilisateurs dans les systèmes de recommandation e-learning en utilisant une combinaison de méthodes.
Effective Preference Elicitation Techniques for Content Recommendation in Online Learning Environments. [72]	Effective Preference Elicitation Techniques for Content Recommendation in Online Learning Environments" Approche : Développement et évaluation de techniques efficaces pour l'élicitation des préférences.	Identification et évaluation des techniques les plus efficaces pour éliciter les préférences des utilisateurs dans les environnements d'apprentissage en ligne.

Tableau 2 : Tableau récapitulatif des travaux sur l'élicitation des préférences à partir des commentaires

Ce tableau résume différentes approches pour l'élicitation des préférences des apprenants dans les environnements d'apprentissage en ligne. Les méthodes varient de l'analyse des commentaires des utilisateurs à l'utilisation des retours d'expérience des étudiants, en passant par des approches hybrides et des techniques spécifiques. Chaque approche vise à extraire et à comprendre les préférences des utilisateurs afin de personnaliser les recommandations de contenu et améliorer l'efficacité des systèmes de recommandation en e-learning. Ces travaux mettent en évidence l'importance d'adapter les méthodes d'élicitation pour répondre aux besoins variés des apprenants.

1.4.7. Le modèle Bert :

1.4.7.1. Définition :

BERT, abréviation de "Bidirectional Encoder Representations from Transformers", est un modèle de langage développé par Google. Il est conçu pour capturer les relations contextuelles entre les mots dans un texte en utilisant une architecture de transformers bidirectionnels. Contrairement aux modèles de langage précédents qui traitent les mots dans un texte de manière séquentielle, BERT prend en compte le contexte global en analysant le texte dans les deux directions.

BERT a été conçu pour pré-entraîner des représentations profondes bidirectionnelles des mots à partir de texte non étiqueté en conditionnant conjointement sur les contextes gauche et droit. Dans une approche de transfert d'apprentissage, le modèle pré-entraîné peut-être ensuite affiné pour une large gamme de tâches de traitement du langage naturel (NLP). En exploitant les caractéristiques linguistiques apprises lors de la première étape, l'affinage donne lieu à une procédure beaucoup plus légère.

L'utilisation de masques de tokens dans l'entraînement de BERT, également connue sous le nom de Masked Language Model (MLM), lui permet d'apprendre à prédire les mots manquants

dans un texte, ce qui renforce encore sa capacité à comprendre le contexte global du langage. [73]

1.4.7.2. Entrées du modèle :

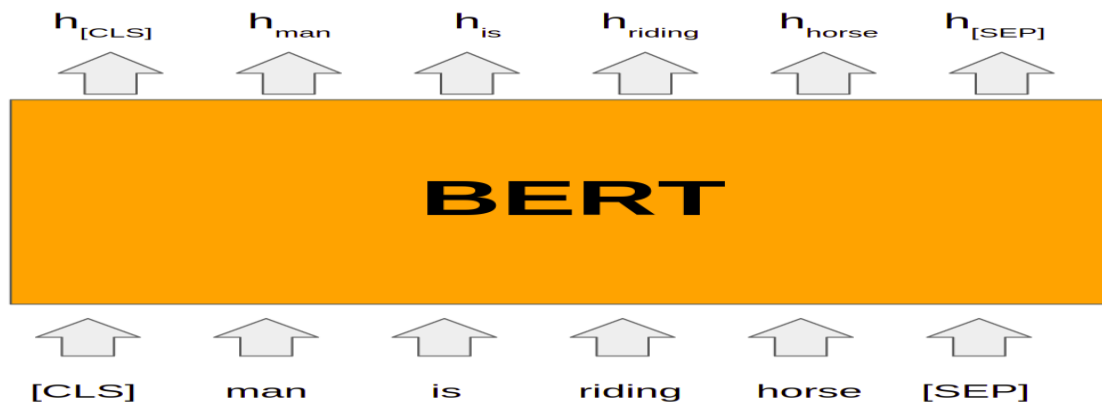


Figure 1.2. Entrées du modèle (W6):

Le modèle transformateur se compose d'une architecture codeur-décodeur. Lorsque nous introduisons une phrase d'entrée (source) dans le modèle, elle est traitée par l'encodeur. L'encodeur apprend alors une représentation de la phrase d'entrée et transmet cette représentation au décodeur. À son tour, le décodeur utilise cette représentation pour générer la phrase de sortie (cible). Pour initialiser ce processus, le premier jeton d'entrée est marqué par un jeton spécial [CLS], qui dans ce contexte signifie "Classification".

1.4.7.3 Encodeur de bert

BERT (Bidirectional Encoder Representations from Transformers) ne possède pas deux encodeurs distincts mais utilise une architecture Transformer qui comprend à la fois des couches d'encodeur et de décodeur. Cependant, dans le contexte de BERT, il y a deux variantes principales : BERT base et BERT large, qui se différencient principalement par le nombre de couches d'encodeur.

BERT base : BERT base utilise 12 couches d'encodeur Transformer. Chaque couche d'encodeur est conçue pour effectuer des opérations de self-attention et de feed-forward. Ces couches d'encodeur permettent à BERT base d'apprendre des représentations bidirectionnelles de mots et de phrases dans le texte en utilisant un modèle pré-entraîné sur un large corpus de texte.

BERT large : BERT large est une version étendue de BERT base avec 24 couches d'encodeur Transformer. Cette augmentation du nombre de couches permet à BERT large de capturer des dépendances plus complexes et de mieux modéliser les relations à longue distance dans le texte. [74]

La figure suivante explique les deux encodeurs de bert :

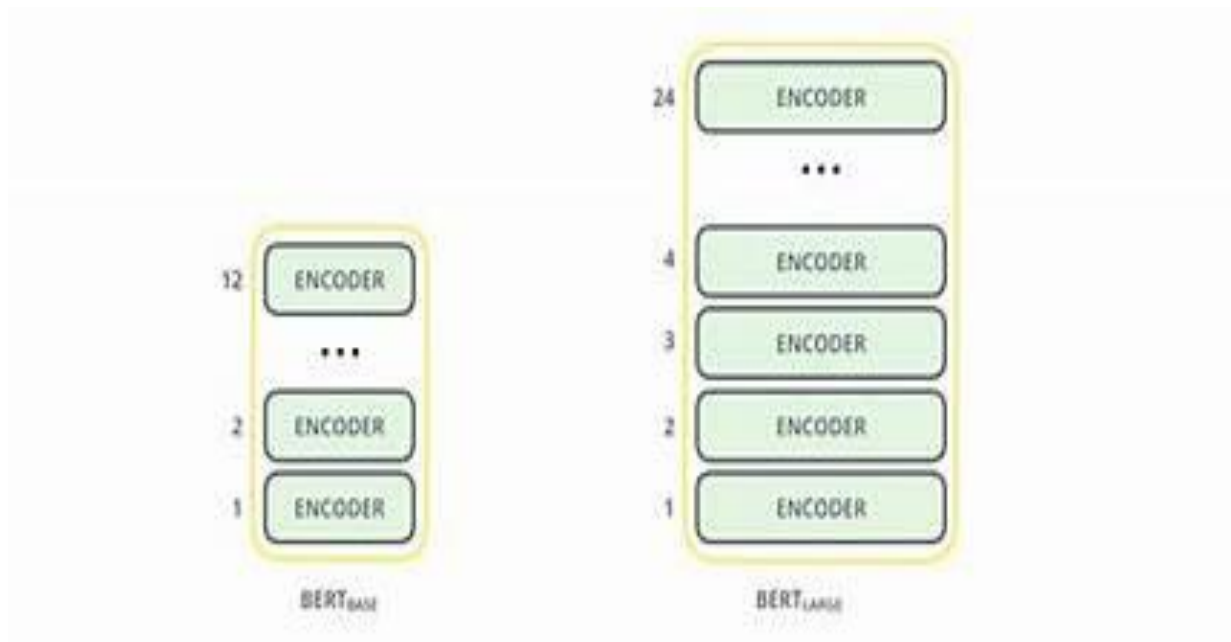


Figure 1.3 explique les deux encodeurs de bert (w7)

1.4.7.3. Architecture de BERT base uncased

BERT (Bidirectional Encoder Representations from Transformers) est un modèle de langage naturel préentraîné développé par Google. "Uncased" fait référence à une version de BERT où le texte est converti en minuscules avant l'entraînement, ce qui permet au modèle de ne pas faire de distinction entre majuscules et minuscules. BERT est particulièrement remarquable car il peut capturer la sémantique bidirectionnelle des mots et des phrases, en tenant compte du contexte à la fois à gauche et à droite de chaque mot dans une phrase. Ce modèle est entraîné sur un large corpus de texte non étiqueté, ce qui lui permet d'apprendre des représentations de mots et de phrases très riches. Il est capable de générer des représentations contextuelles de mots qui sont sensibles au contexte dans lequel ils apparaissent. BERT a été prouvé efficace dans une large gamme de tâches de traitement du langage naturel, telles que la classification de texte, la compréhension de texte et la traduction.

BERT a été créé à partir de 12 modules encodeurs empilés de l'architecture originale du Transformer. Comme le montre la figure 4.6, l'architecture de BERT est composée de différents sous-systèmes. Une couche d'incorporation qui prépare la séquence d'entrée à être traitée. La couche d'auto-attention, si importante, qui calcule la relation entre différents mots de la séquence, ainsi qu'une couche feed-forward. En plus de cela, il dispose également de connexions

résiduelles autour des deux couches ainsi que de deux couches de normalisation de couche.

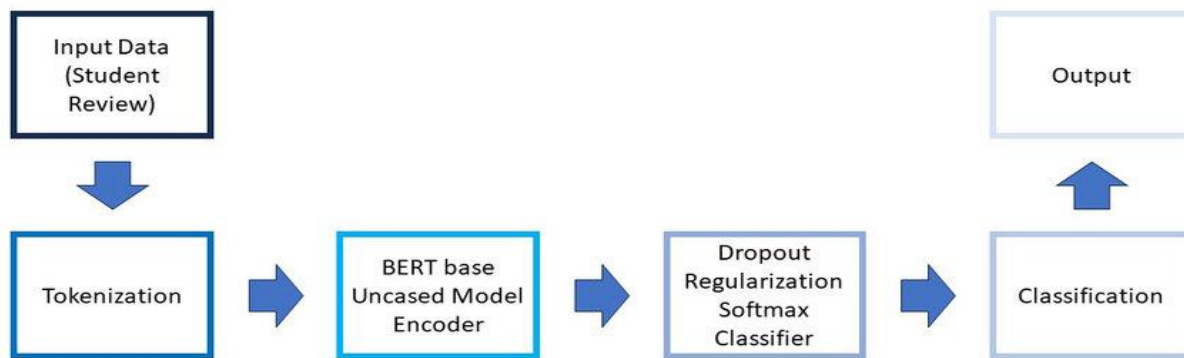


Figure1.4. BERT base uncased -model-architecture [75]

1.5. Conclusion

Ce premier chapitre a permis de réaliser une évaluation approfondie de l'état de l'art des systèmes de recommandation dans le domaine de l'e-learning. Nous avons commencé par définir les plateformes d'e-learning et les tendances actuelles, notamment les MOOC (Massive Open Online Courses). Nous avons détaillé le fonctionnement des MOOC et exploré leur relation avec l'e-learning en général.

Nous avons ensuite abordé les systèmes de recommandation en e-learning, en décrivant les diverses techniques employées pour fournir des recommandations personnalisées aux apprenants. Une attention particulière a été accordée à l'élicitation des préférences des apprenants basés sur les commentaires, en examinant les défis associés et en soulignant l'importance de cette approche. Nous avons illustré cela en présentant un aperçu des travaux connexes déjà réalisés dans ce domaine.

Pour conclure, nous avons introduit le modèle BERT et son architecture, en expliquant comment ce modèle de traitement du langage naturel peut être utilisé pour améliorer les systèmes de recommandation en e-learning. Ce chapitre a donc fourni les bases nécessaires pour la conception de notre propre système de recommandation en utilisant le modèle BERT base uncased.

Dans le prochain chapitre, nous détaillerons la conception de notre système et expliquerons comment nous exploitons les capacités du modèle BERT base uncased pour offrir des recommandations encore plus précises et personnalisées aux apprenants.

Chapitre 2 : Conception et implémentation

2.1. Introduction :

Dans cette section, nous présentons une approche avancée pour l'analyse des sentiments dans les MOOCs. Notre méthode utilise le modèle BERT Base Uncased, reconnu pour ses performances dans le traitement automatique du langage naturel en anglais. L'objectif principal est d'extraire les préférences des apprenants à partir de ces commentaires

2.2. Contexte :

Les MOOCs (massive open online courses) sont des plateformes éducatives numériques qui offrent des cours en ligne ouverts à un grand nombre d'apprenants, sans contrainte de localisation géographique ou de coût. Ils proposent une diversité de sujets et ils sont généralement dispensés par des institutions académiques ou des experts dans leur domaine.

Les systèmes de recommandation sont devenus indispensables pour les plateformes d'apprentissage, y compris les MOOCs, en raison de leur capacité à améliorer l'expérience utilisateur, à optimiser la rétention et l'engagement, et à personnaliser l'apprentissage pour chaque individu. Cependant, l'apprenant en général, est intéressé par divers axes en même temps et engage dans plusieurs cours.

L'élicitation des préférences des utilisateurs est en effet une méthode essentielle pour générer des recommandations personnalisées dans les systèmes de recommandation.

2.3. Problématique :

La variabilité des préférences représente un défi significatif pour les systèmes de recommandation, surtout dans le contexte de l'apprentissage afin de cerner tous les besoins informationnels de l'apprenant, ce qui est le cas pour les MOOCs dont elle offre également une opportunité d'innovation et d'amélioration continue. En adoptant des approches centrées sur les apprenants et en mettant en œuvre des solutions adaptatives, il est possible de créer des environnements d'apprentissage en ligne qui sont véritablement enrichissants, engageants et propices à la réussite pour tous les participants.

Cependant dans un environnement où les préférences sont variables, il faut avoir une stratégie pour les éliciter afin de répondre à tous les besoins de l'apprenant en générant des recommandations pertinentes.

D'où la problématique qui se pose est la suivante :

Comment peut-on éliciter les préférences variables des apprenants dans les MOOCs ?

2.4. Solution proposée :

Pour répondre à la problématique posée, l'étude présentée dans ce manuscrit vise à explorer différentes méthodes pour appréhender la variabilité des préférences, qui peuvent être utilisées pour comprendre les préférences des apprenants dans les MOOCs.

Ensuite, ces préférences seront utilisées dans le processus de recommandation en recourant aux méthodes de filtrage collaboratif et de filtrage basé sur le contenu, qui sont des approches traditionnelles visant à fournir des recommandations pertinentes et personnalisées.

Chapitre 2 : Conception et implémentation

Dans ce travail on utilise le modèle bert base uncased pour le traitement de langage naturel des commentaires en anglais. Donc en fait l'analyse des sentiments des commentaires et extraire les préférences des apprenants pour faire une recommandation selon ces préférences. Cette recommandation divisée sur deux catégories une recommandation traductionnelle et recommandation diversifier selon les commentaires des apprenants.

Tout d'abord, il est essentiel de collecter des données sur les interactions des apprenants avec les cours, telles que les évaluations, les commentaires et les notes. Ensuite, ces données sont bien prétraitées pour faire un entraînement de modèle de traitement de langage naturel et extraire les préférences variables des apprenants

Des techniques d'apprentissage automatique sont alors utilisées pour développer un modèle prédictif capable de recommander des cours personnalisés en fonction des préférences individuelles de chaque apprenant. Ce modèle est ensuite validé et ajusté en continu en intégrant le feedback des apprenants sur les recommandations fournies. Enfin le système est amélioré et évalué dans des conditions réelles pour faire une analyse des sentiments.

2.5. Motivation et objectifs :

En exploitant les avancées dans la recherche sur l'élicitation des préférences et les méthodes de traitement de données, ce projet vise à développer des approches novatrices pour capturer et interpréter efficacement les préférences variables des apprenants dans les MOOCs. En comprenant mieux les attentes et les intérêts individuels des apprenants, nous pourrions concevoir des systèmes de recommandation plus précis et adaptés, contribuant ainsi à améliorer l'expérience d'apprentissage en ligne et à favoriser l'engagement et la réussite des apprenants.

Ce travail développe des méthodes de traitement de langage naturel dans la langue anglaise pour mieux répondre aux besoins diversifiés des apprenants dans les MOOCs. En mettant l'accent sur la variabilité des préférences, nous visons à ouvrir de nouvelles perspectives pour une éducation en ligne plus inclusive et efficace, offrant ainsi des opportunités d'apprentissage enrichissantes pour tous les participants.

2.6. Conception du système :

Le système "**bertrec**" est conçu pour l'élicitation des préférences des apprenants et la génération de recommandations à partir des résultats de l'apprentissage.

Au début en fait l'apprentissage de modèle Bert base Uncased sur l'analyse des sentiments puis faire l'évaluation de modèle.

Puis à partir de ce modèle (Bert base Uncased) en fait prédiction de notre dataset et extraire les sentiments des commentaires pour faire la recommandation.

La conception et l'implémentation d'un système utilisant BERT base Uncased pour l'élicitation des préférences et la génération de recommandation à partir des commentaires peuvent être décrites en deux phases principales :

1. Phase d'élicitation des préférences à partir des commentaires :

Chapitre 2 : Conception et implémentation

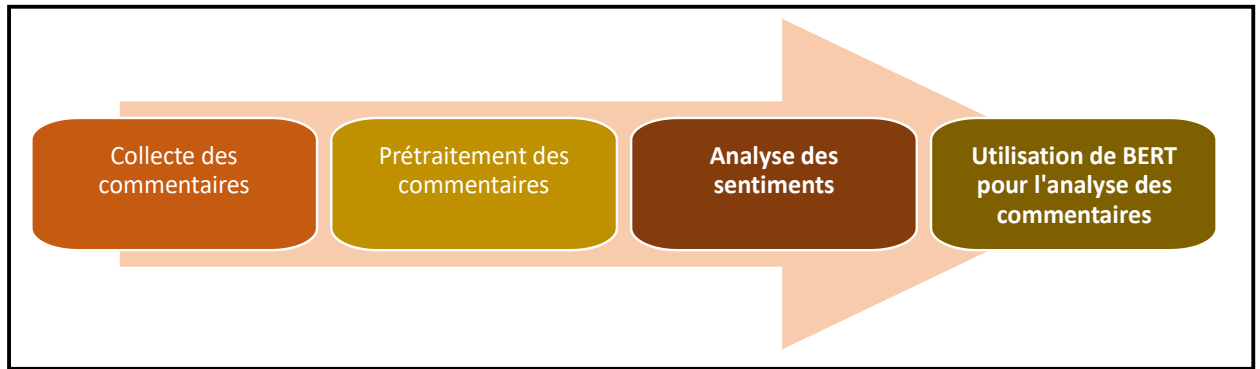


Figure 2.1. Elicitation des préférences à partir des commentaires

2. Phase de génération de la recommandation basée préférences diversifiées :

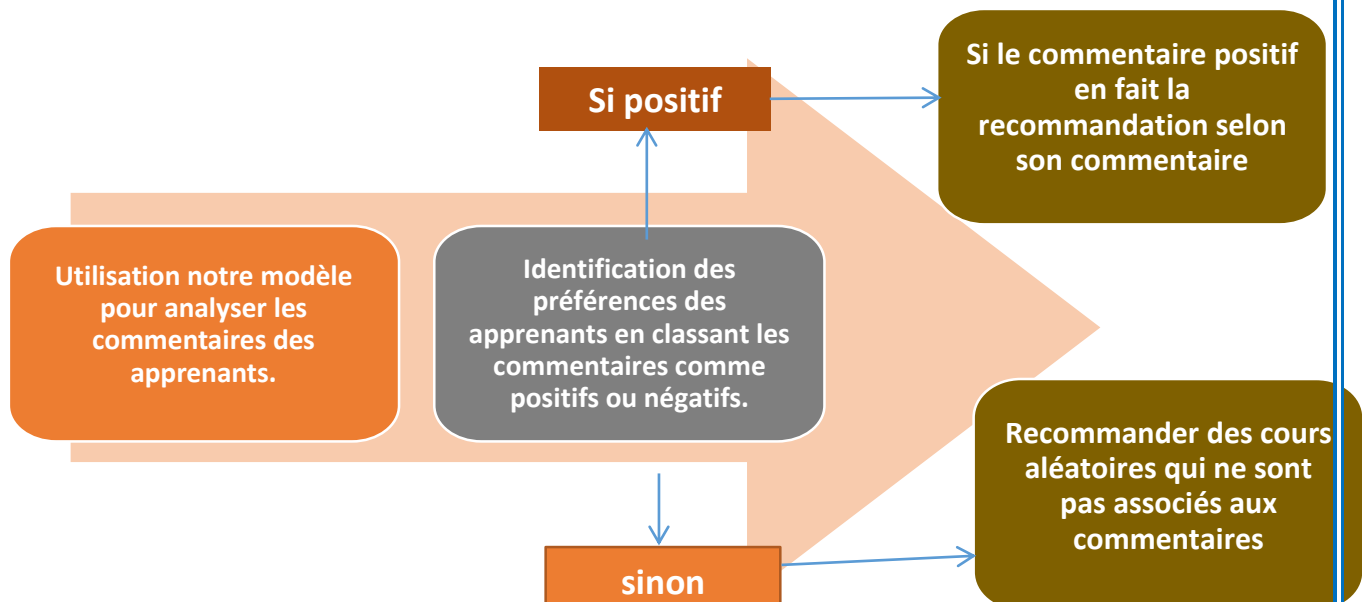


Figure 2.2. Génération de la recommandation basée sur les commentaires

Dans ce qui suit, chaque étape du processus sera bien expliquée et détaillée.

2.6.1. PHASE : L'Elicitation Des Préférences :

Dans cette phase, notre objectif est d'extraire avec précision les préférences des apprenants à partir de leurs commentaires sur les cours disponibles, plus précisément, leurs sentiments qu'ils soient positifs ou négatifs. Cette étape est cruciale car ces informations serviront de fondement à la génération de la recommandation dans la phase suivante.

Pour assurer l'efficacité et la précision de notre système, plusieurs étapes doivent être suivies lors de cette phase, comme illustré dans la Figure 5 ci-dessous :

Chapitre 2 : Conception et implémentation

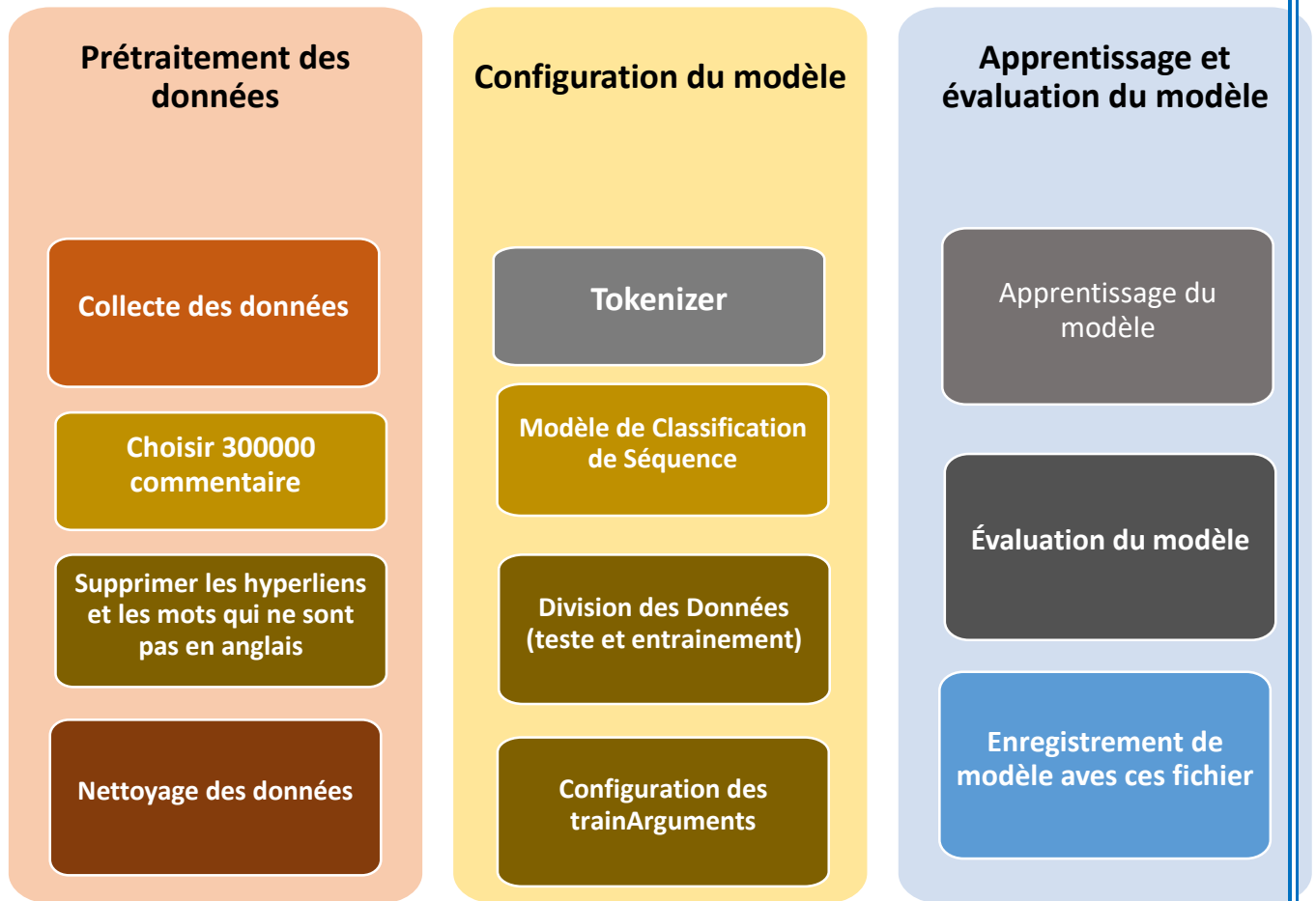


Figure 2.3. Les étapes de l'élucation des préférences dans le système bert base uncased

Nous détaillons dans ce qui suit les différentes étapes citées ci-dessus :

2.6.1.1. Etape1 Préparation Des Données :

Tout d'abord, on commence par tout ce qui concerne le nettoyage des données et leur préparation pour qu'elles soient prêtes à être utilisées par le système afin d'appliquer l'apprentissage du modèle.

➤ Chargement de Dataset :

Nous avons utilisé dans notre travail deux datasets (le premier contient les cours et le deuxième les commentaires sur ces cours).

Dans cette étape, on commence par l'importation de la bibliothèque *Pandas* et le chargement du fichier *CSV* contenant notre deux Dataset. Ensuite, on utilise la méthode "*head()*" pour afficher les premières lignes des deux Datasets et s'assurer que le chargement s'est bien déroulé.

- *Exemple de code source :*

❖ Dataset N°01 :

Chapitre 2 : Conception et implémentation

```
courses_data = pd.read_csv('/kaggle/input/udemy-courses/Course_info.csv')
courses_data.head()
```

[6]:

	id	title	is_paid	price	headline	num_subscribers	avg_rating	num_reviews	num_comments	num_lectures	content_length_min	published_time	last_update_date	category	subcategory	topic	language
0	4715.0	Online Vegan Vegetarian Cooking School	True	24.99	Learn to cook delicious vegan recipes. Filmed ...	2231.0	3.75	134.0	42.0	37.0	1268.0	2010-08-05T22:06:13Z	2020-11-06	Lifestyle	Food & Beverage	Vegan Cooking	English
1	1769.0	The Lean Startup Talk at Stanford E-Corner	False	0.00	Debunking Myths of Entrepreneurship A startup ...	26474.0	4.50	709.0	112.0	9.0	86.0	2010-01-12T18:09:46Z	NaN	Business	Entrepreneurship	Lean Startup	English
2	5664.0	How To Become a Vegan, Vegetarian, or Flexitarian	True	19.99	Get the tools you need for a lifestyle change ...	1713.0	4.40	41.0	13.0	14.0	82.0	2010-10-13T18:07:17Z	2019-10-09	Lifestyle	Other Lifestyle	Vegan Cooking	English
3	7723.0	How to Train a Puppy	True	199.99	Train your puppy the right way with Dr. Ian Du...	4988.0	4.80	395.0	88.0	36.0	1511.0	2011-06-20T20:08:38Z	2016-01-13	Lifestyle	Pet Care & Training	Pet Training	English
4	8157.0	Web Design from the Ground Up	True	159.99	Learn web design online: Everything you need to...	1266.0	4.75	38.0	12.0	38.0	569.0	2011-06-23T18:31:20Z	NaN	Design	Web Design	Web Design	English

Figure 2.4. Chargement et lecture du fichier CSV « course_info ».

la 1ere dataset contient des

❖ Dataset n°02 :

```
import pandas as pd

# Charger les données de commentaires depuis le fichier CSV
comments_data = pd.read_csv('/kaggle/input/udemy-courses/Comments.csv')

# Échantillonner aléatoirement 20 000 commentaires
sampled_comments_data = comments_data.sample(20000)

# Enregistrer l'échantillon dans un nouveau fichier CSV
sampled_comments_data.to_csv('sampled_comments.csv', index=False)
sampled_comments_data.head()
```

[9]:

	id	course_id	rate	date	display_name	comment
472066	376720	234704	5.0	2015-05-25T01:15:29-07:00	Dũng	This course is awesome! Highly recommend!
7081883	120491476	1672410	5.0	2022-08-05T09:33:09-07:00	Miran	The course is very detailed and of high quality
2137005	107327114	3907162	5.0	2022-02-11T11:15:30-08:00	Jonas	Der Dozent macht einen sehr ehrlichen Eindruck
2770230	18602616	1290784	5.0	2018-07-19T11:18:46-07:00	Daniel	Sim, achei interessante
8865576	7604094	543600	5.0	2017-06-17T14:18:21-07:00	Bill	Great course. His presentation is clear and f...

Figure 2.5. Chargement et lecture du fichier CSV « comments.csv »

Cette dataset contient des informations détaillées sur tous les cours disponibles sur Udemy le 10 octobre 2022. Ces données sont fournies dans le fichier "Course_info.csv". De plus, plus de 9 millions de commentaires ont été collectés et sont fournis dans le fichier "Comments.csv". Les informations sur plus de 209 000 cours ont été collectées en utilisant le scraping web du site Udemy. Udemy propose 209 734 cours et 73 514 instructeurs enseignant des cours dans 79 langues et dans 13 catégories différentes.

Remarque : la 2^{ème} dataset contient plus de 9 millions commentaires mais nous avons pris juste un échantillon aléatoire de 300000 commentaires et enregistré l'échantillon dans un nouveau fichier CSV.

Chapitre 2 : Conception et implémentation

➤ Prétraitement et nettoyage des données :

Dans le domaine de l'analyse de texte, l'une des étapes essentielles consiste à prétraiter les données textuelles. Cela comprend un ensemble d'opérations visant à nettoyer et à préparer les données avant de les utiliser dans un modèle. Ces opérations incluent généralement le nettoyage des données brutes, l'élimination des mots vides, la lemmatisation ou la racinisation, ainsi que d'autres techniques visant à normaliser et à simplifier le texte.

D'abord, je fusionne les commentaires et les cours pour une analyse plus approfondie. Cela comprend la suppression des lignes ne contenant pas la langue anglaise dans le premier jeu de données et des commentaires qui ne sont pas en anglais dans le deuxième jeu de données.

Ensuite, les expressions régulières nous permettent de détecter et de supprimer les hyperliens et les mots qui ne sont pas en anglais avec la bibliothèque "*langdetect*" qui pourraient apparaître dans les commentaires, ce qui simplifie et nettoie les données textuelles en vue de l'analyse.

Puis À partir des notes de commentaire (on a un tableau *rate* dans la dataset), les sentiments peuvent être extraits en appliquant une règle simple : si la note est inférieure à 3, le sentiment est considéré comme négatif ; sinon, il est considéré comme positif.

Enfin, je procède au nettoyage des données pour assurer leur qualité et leur cohérence avant toute analyse ultérieure. En commence le nettoyage des données par l'élimination des émojis inutiles du texte. Les émojis peuvent contenir des informations non pertinentes, ce qui complique l'analyse des données, quelle que soit la langue utilisée à l'aide de la bibliothèque « *re* » et « *spacy* » et des expressions régulières puis en fait la suppression des caractères spéciale et les points et les virgules etc....

Les figures suivantes expliquent ces opérations :

- *Voici quelque exemple de code :*

```
import pandas as pd
from langdetect import detect

# Charger les données de commentaires depuis le fichier CSV
sampled_comments_data = pd.read_csv('/kaggle/working/preprocessed_comments.csv')

# Définir une fonction pour détecter la langue
def detect_langue(comment):
    try:
        return detect(comment)
    except:
        return 'unknown'

# Ajouter une colonne de langue au DataFrame
sampled_comments_data['langue'] = sampled_comments_data['comment'].apply(detect_langue)

# Filtrer les commentaires qui sont en anglais
english_comments = sampled_comments_data[sampled_comments_data['langue'] == 'en'].copy()

# Supprimer la colonne de langue temporaire
english_comments.drop(columns=['langue'], inplace=True)

# Enregistrer les commentaires en anglais dans un nouveau fichier CSV
english_comments.to_csv('english_comments.csv', index=False)
```

Figure 2.6. Suppression des commentaires qui ne sont pas en anglais avec la bibliothèque langdetect.

Remarque : Après la suppression des commentaires qui ne sont pas en anglais, nous avons **185518** échantillons.

Chapitre 2 : Conception et implémentation

La Figure 2.7 explique l'opération de La création d'un tableau sentiments à partir de note de commentaire :

```
# Convertir les valeurs de la colonne 'rate' en nombres flottants
comments_df['rate'] = comments_df['rate'].astype(float)

# Créer une nouvelle colonne 'Sentiment' basée sur la colonne 'rate'
comments_df['Sentiment'] = comments_df['rate'].apply(lambda x: 'negatif' if x < 3 else 'positif')

# Transformer 'positif' en 1 et 'negatif' en 0 dans la colonne 'Sentiment'
comments_df['Sentiment'] = comments_df['Sentiment'].replace({'positif': 1, 'negatif': 0})
```

+ Code + Markdown

```
# Transformer 'positif' en 1 et 'negatif' en 0 dans la colonne 'Sentiment'
comments_df['Sentiment'] = comments_df['Sentiment'].replace({'positif': 1, 'negatif': 0})
```

```
# Afficher le nombre de positifs et de négatifs dans la colonne 'Sentiment'
sentiment_counts = comments_df['Sentiment'].value_counts()
print("Nombre de positifs :", sentiment_counts[1])
print("Nombre de négatifs :", sentiment_counts[0])
```

Nombre de positifs : 175511
Nombre de négatifs : 10007

Figure 2.7. La création d'un tableau sentiments à partir de rate

Après on a choisi deux sous-ensembles de sentiments différents ayant la même taille (10007 commentaires positifs et 10007 commentaires négatifs). Ensuite, le nettoyage de données est effectué.

Les Figures 2.8 et 2.9 contiennent les codes sources qui expliquent le processus de nettoyage des commentaires.

Chapitre 2 : Conception et implémentation

```
stop_words = set(stopwords.words('english'))

def remove_punctuation_and_stopwords(comment):
    # Supprimer la ponctuation
    comment = re.sub(r'^\w\s', '', comment)
    # Tokenizer et supprimer les stopwords
    word_tokens = word_tokenize(comment)
    filtered_comment = [word for word in word_tokens if word not in stop_words]
    return ' '.join(filtered_comment)

combined_comments['comment'] = combined_comments['comment'].apply(remove_punctuation_and_stopwords)

+ Code + Markdown

stemmer = PorterStemmer()

def stem_comment(comment):
    word_tokens = word_tokenize(comment)
    stemmed_comment = [stemmer.stem(word) for word in word_tokens]
    return ' '.join(stemmed_comment)

combined_comments['comment'] = combined_comments['comment'].apply(stem_comment)

import pandas as pd
import spacy

# Charger le modèle de langue anglaise de spaCy
nlp = spacy.load('en_core_web_sm')

for col in eng_data.columns:
    combined_comments[col] = combined_comments[col].astype(str)

# Lemmatisation avec spaCy
def lemmatize_comment(comment):
    doc = nlp(comment)
    lemmatized_comment = [token.lemma_ for token in doc]
    return ' '.join(lemmatized_comment)

combined_comments['comment'] = combined_comments['comment'].apply(lemmatize_comment)
```

Figure 2.8. Le nettoyage des données

```
def handle_special_characters(comment):
    comment = re.sub(r'^a-zA-Z\s', '', comment)
    comment = re.sub(r'\s+', ' ', comment).strip()
    return comment

combined_comments['comment'] = combined_comments['comment'].apply(handle_special_characters)

+ Code + Markdown

import pandas as pd
import spacy
import re

# Charger le modèle de langue anglaise de spaCy
nlp = spacy.load('en_core_web_sm')

# Assurez-vous que toutes les colonnes sont de type string
for col in combined_comments.columns:
    combined_comments[col] = combined_comments[col].astype(str)

# Fonction de prétraitement pour éliminer les chiffres et lemmatiser
def preprocess_comment(comment):
    # Éliminer les chiffres
    comment = re.sub(r'\d+', '', comment)
    # Lemmatisation avec spaCy
    doc = nlp(comment)
    lemmatized_comment = [token.lemma_ for token in doc]
    return ' '.join(lemmatized_comment)

# Appliquer la fonction de prétraitement aux commentaires
combined_comments['comment'] = combined_comments['comment'].apply(preprocess_comment)
```

Figure 2.9. Nettoyage des données en utilisant la bibliothèque « re » et « spacy »

2.6.1.2. Étape 2 : Configuration Du Modèle ber-base-Uncased :

Les étapes représentent le flux de travail standard pour configurer et entraîner un modèle BERT pour la classification de séquences. Chaque étape est cruciale pour assurer que le modèle est correctement préparé, entraîné et évalué, en utilisant les meilleures pratiques de traitement du langage naturel et d'apprentissage automatique.

- **En commence la configuration de notre par l'importation les importations des Bibliothèques nécessaires**

Pour faire l'entraînement de notre modèle, l'utilisation de NumPy, PyTorch et Hugging Face Transformers est essentielle. NumPy est crucial pour le traitement efficace des données, telles que la gestion des matrices de tokens BERT. PyTorch offre les outils nécessaires à la construction, l'entraînement et l'optimisation du modèle BERT, y compris la gestion des gradients et le support GPU. Hugging Face Transformers simplifie l'intégration de modèles NLP pré-entraînés comme BERT, facilitant ainsi leur adaptation à des tâches spécifiques comme la classification de texte. Ensemble, ces bibliothèques permettent le déploiement de solutions avancées et performantes pour l'analyse et le traitement automatique du langage naturel.

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, recall_score, precision_score, f1_score
import torch
from transformers import TrainingArguments, Trainer
from transformers import BertTokenizer, BertForSequenceClassification
```

Figure 2.10. L'importation des bibliothèques nécessaire

- 2eme étape est le chargement du Tokenizer et du Modèle Bert base Uncased pour la Classification de Séquences.

Le chargement du tokenizer et du modèle Bert base Uncased implique l'utilisation de **BertTokenizer** et **BertForSequenceClassification** de la bibliothèque Hugging Face Transformers. Le tokenizer BERT, chargé via `BertTokenizer.from_pretrained('Bert-base-Uncased')`, convertit les textes bruts en séquences de tokens compréhensibles par le modèle BERT. Ce processus inclut la segmentation des mots en sous-mots ou tokens, l'ajout de tokens spéciaux pour indiquer les débuts et fins de séquences, et la conversion des tokens en identifiants numériques.

Le modèle BERT pour la classification de séquences est chargé avec **BertForSequenceClassification.from_pretrained ('Bert-base-Uncased', num_labels=2)**, ajoutant une couche de classification au modèle BERT pré-entraîné pour permettre la prédiction des classes de texte, ici pour une tâche de classification binaire (par exemple, sentiments positifs ou négatifs). Ensemble, ces étapes préparent le texte et configurent le modèle BERT pour traiter des tâches spécifiques de traitement du langage naturel, en utilisant des représentations linguistiques puissantes apprises à partir de grands corpus de texte.

```
from transformers import BertTokenizer, BertForSequenceClassification
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertForSequenceClassification.from_pretrained('bert-base-uncased', num_labels=2)
```

Figure 2.11. Chargement du Tokenizer et du Modèle Bert base Uncased

Chapitre 2 : Conception et implémentation

➤ *Voici les commentaires avant et après la tokenization :*

```
# Charger le tokenizer BERT
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')

# Fonction de tokenization
def encode_comments(comments, tokenizer):
    return tokenizer(comments, truncation=True, padding=True, return_tensors='pt')

# Liste de commentaires
comments = ["C'est un exemple de commentaire.", "Un autre commentaire à traiter."]

# Tokenization des commentaires
encodings = encode_comments(comments, tokenizer)

# Affichage des commentaires tokenisés
for i, comment in enumerate(comments):
    print(f"Commentaire {i+1}: {comment}")
    print(f"Tokens: {tokenizer.convert_ids_to_tokens(encodings['input_ids'][i])}")

Commentaire 1: C'est un exemple de commentaire.
Tokens: ['[CLS]', 'c', "'", 'est', 'un', 'ex', '##em', '##ple', 'de', 'comment', '##aire', '.', '[SEP]']
Commentaire 2: Un autre commentaire à traiter.
Tokens: ['[CLS]', 'un', 'au', '##tre', 'comment', '##aire', 'a', 'trait', '##er', '.', '[SEP]', '[PAD]', '[PAD]']
```

Figure 2.12. La tokenization des commentaires

➤ *Exemple illustratif :*

- Commentaries Original: **basic course company modeled sap**
- Tokens: **['[CLS]', 'basic', 'course', 'company', 'modeled', 'sap']**
- IDs de Token : **tensor ([101, 3937, 2607, 2194, 14440, 20066, 102,**

❖ La 3eme étape est la Préparation des Données pour l'Entraînement et la Validation

La préparation des données commence par la conversion des commentaires (\mathbf{X}) et des sentiments associés (\mathbf{y}) en listes pour une manipulation plus facile. Ensuite, ces données sont divisées en ensembles d'entraînement et de validation en utilisant la fonction `train_test_split` de `sklearn`, avec 80 % des données allouées à l'entraînement et 20 % à la validation. Cette division est stratifiée (`stratify=y`) pour maintenir la même proportion de classes positives et négatives dans les deux ensembles, garantissant ainsi que la distribution des sentiments est équilibrée entre l'entraînement et la validation.

```
X = list(df["comment"])
y = list(df["Sentiment"])
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, stratify=y)
```

Figure 2.13. Préparation des Données pour l'Entraînement et la Validation

❖ La 4eme étape est la tokenization des Données pour BERT :

La tokenization des données commence par l'utilisation du tokenizer BERT pré-entraîné pour convertir chaque commentaire en une séquence de tokens. Le processus de tokenization segmente les mots en sous-mots ou tokens et ajoute des tokens spéciaux comme `[CLS]` au début et `[SEP]` à la fin de chaque séquence. Pour uniformiser la longueur des séquences, le padding est appliqué, ajoutant des zéros aux séquences plus courtes, et la troncature réduit les séquences trop longues pour qu'elles respectent une longueur maximale prédéfinie. Cela optimise l'utilisation de la mémoire et améliore l'efficacité du traitement en permettant de traiter des séquences de longueurs différentes de manière cohérente dans des batches.

```
X_train_tokenized = tokenizer(X_train, padding=True, truncation=True, max_length=512)|
X_val_tokenized = tokenizer(X_val, padding=True, truncation=True, max_length=512)
```

Figure 2.14. tokenization des Données pour BERT

❖ La 5eme étape de la configuration est la création de la classe Dataset pour bert

La création de classe est important pour préparer les données tokenisées afin qu'elles puissent être utilisées efficacement par un modèle BERT lors de l'entraînement. Cette classe encapsule les données tokenisées et leurs étiquettes, le cas échéant, dans un format compatible avec PyTorch.

La méthode `__init__` initialise la classe en prenant les encodages des données (tels que les `input_ids` et `attention_mask` générés par le tokenizer BERT) et les étiquettes associées (par exemple, les sentiments positifs ou négatifs). La méthode `__getitem__` est implémentée pour permettre l'accès aux éléments du dataset par index. Elle retourne un dictionnaire contenant les encodages des données et, si spécifié, les étiquettes converties en tensors PyTorch. Enfin, la méthode `__len__` renvoie la taille du dataset, ce qui est essentiel pour l'itération sur le dataset pendant l'entraînement du modèle.

Cette approche garantit que les données sont chargées de manière optimale en mémoire et fournies au modèle BERT par lots (batching), ce qui améliore l'efficacité du processus d'entraînement tout en respectant les exigences de format et de gestion des données propres à PyTorch.

```
# Create torch dataset
class Dataset(torch.utils.data.Dataset):
    def __init__(self, encodings, labels=None):
        self.encodings = encodings
        self.labels = labels

    def __getitem__(self, idx):
        item = {key: torch.tensor(val[idx]) for key, val in self.encodings.items()}
        if self.labels:
            item["labels"] = torch.tensor(self.labels[idx])
        return item

    def __len__(self):
        return len(self.encodings["input_ids"])
```

Figure 2.15. Création de la Classe Dataset pour BERT

❖ La dernière étape de la configuration de modèle est la Définition des Métriques pour l'Évaluation du Modèle

Une fonction `compute_metrics` est définie pour évaluer les performances du modèle. Cette fonction calcule des métriques telles que la précision, le rappel, le F1-score et l'exactitude en comparant les prédictions du modèle avec les étiquettes réelles des données de validation.

```
def compute_metrics(pred):
    labels = pred.label_ids
    preds = pred.predictions.argmax(-1)
    precision, recall, f1, _ = precision_recall_fscore_support(labels, preds, average='macro')
    acc = accuracy_score(labels, preds)
    return {'accuracy': acc, 'f1': f1, 'precision': precision, 'recall': recall}
```

Figure 2.16. Définition des Métriques pour l'Évaluation du Modèle

2.6.1.3. Etape 2 Apprentissage du modèle Bert base Uncased :

Nous avons entraîné notre modèle en utilisant le module « *Trainer* » de **PyTorch**. Nous avons configuré notre entraînement en spécifiant les hyper-paramètres tels que le nombre d'époques, la taille du batch, le taux d'entraînement (learning rate). Voici l'explication :

En commence l'apprentissage de modele par Les TrainingArguments

Les TrainingArguments sont cruciaux pour définir les paramètres d'entraînement du modèle BERT. Dans votre configuration, `output_dir` est spécifié comme "output", indiquant que les modèles entraînés et les résultats d'évaluation seront sauvegardés dans ce répertoire. `load_best_model_at_end` est activé avec une valeur de True, ce qui signifie que le meilleur modèle basé sur la métrique spécifiée sera chargé à la fin de l'entraînement.

Chapitre 2 : Conception et implémentation

La **metric_for_best_model** est définie comme "accuracy", indiquant que le modèle sera évalué en fonction de son exactitude. Le **learning_rate** est fixé à $2e-5$, déterminant la vitesse à laquelle le modèle ajuste ses poids pendant l'entraînement. Les tailles de lot par périphérique (**per_device_train_batch_size** et **per_device_eval_batch_size**) sont fixées à 16, ce qui signifie que chaque itération d'entraînement ou d'évaluation traite 16 exemples sur chaque périphérique. **num_train_epochs** est configuré à (3,10,15), spécifiant que l'ensemble des données d'entraînement est parcouru 15 fois pendant l'entraînement du modèle. **logging_steps** est fixé à 100, ce qui permet de journaliser les informations d'entraînement toutes les 100 étapes. Enfin, **eval_strategy** est défini sur "epoch", indiquant que l'évaluation des performances du modèle sur l'ensemble de validation sera effectuée à chaque fin d'époque, tandis que **save_strategy** est également "epoch", ce qui signifie que le modèle sera sauvegardé à la fin de chaque époque d'entraînement.

Ces paramètres sont cruciaux pour optimiser l'entraînement du modèle Bert base Uncased, garantissant une gestion efficace des ressources et une évaluation régulière des performances pour assurer une convergence optimale vers un modèle de haute précision.

Figure 2.17 présente le code source qui permet le réglage des paramètres du modèle dans la phase d'entraînement.

```
# Configuration des arguments d'entraînement
args = TrainingArguments(
    output_dir="/kaggle/working/output",
    logging_dir="/kaggle/working/logs",
    load_best_model_at_end=True,
    metric_for_best_model="accuracy",
    learning_rate=3e-5,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    num_train_epochs=10,
    logging_steps=100,
    evaluation_strategy="epoch",
    save_strategy="epoch",
    report_to="none" # Désactiver la journalisation wandb
)

# Initialisation du Trainer
trainer = Trainer(
    1 model=model,
    2 args=args,
    3 train_dataset=train_dataset,
    4 eval_dataset=val_dataset,
    5 compute_metrics=compute_metrics
)

trainer.train()
```

Figure 2.17. Réglage des paramètres d'entraînement.

Pendant l'entraînement, le modèle est mis à jour en fonction des prédictions et l'exactitude calculées par la fonction de perte. L'entraînement est effectué sur différentes époques jusqu'à ce que la valeur de perte diminue.

Chapitre 2 : Conception et implémentation

La Figure 2.18 présente les différentes valeurs d'évolution de la performance du modèle.

Epoch	Training Loss	Validation Loss	Accuracy
1	0.537900	0.381124	0.851361
2	0.570100	0.427663	0.856857
3	0.244600	0.434392	0.869598
4	0.408100	0.439251	0.855109
5	0.263900	0.421878	0.871097
6	0.384100	0.465461	0.853360
7	0.407400	0.453306	0.850362
8	0.680600	0.711681	0.500125
9	0.192000	0.488448	0.854359
10	0.295200	0.422023	0.867849
11	0.307600	0.458051	0.847864
12	0.342800	0.465096	0.866600
13	0.303200	0.462149	0.868599
14	0.162400	0.465540	0.869348
15	0.215200	0.465808	0.871346

Figure 2.18: Évolution des performances du système.

2.6.1.4. Etape 3 Evaluation du modèle bert-base-uncased :

On présente dans cette section, les résultats obtenus lors de l'entraînement du modèle, en général, et on va les détailler dans le prochain chapitre.

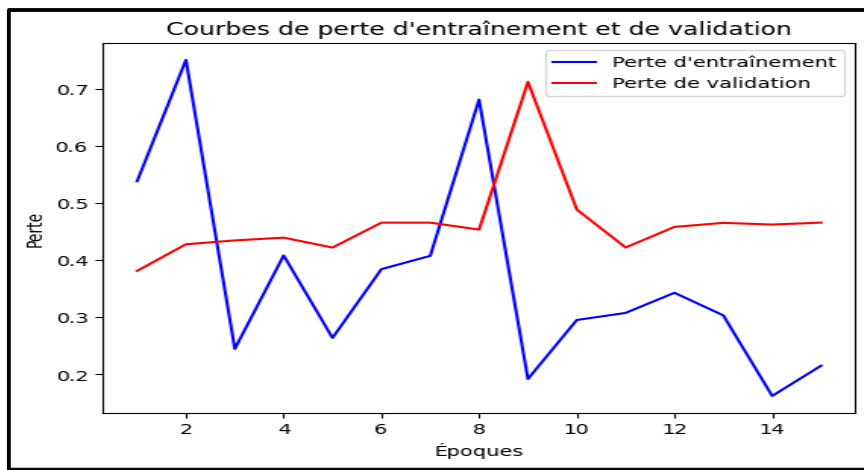


Figure 2.19. Performance d'entraînement du système

Les résultats d'entraînement montrent la performance de notre modèle sur plusieurs époques, évaluée à travers la perte d'entraînement, la perte de validation et l'exactitude (accuracy). Voici une discussion détaillée sur ces résultats :

Perte d'Entraînement et Perte de Validation : La perte d'entraînement diminue progressivement de 0.5379 à 0.2152, indiquant que le modèle ajuste efficacement ses paramètres pour mieux correspondre aux données d'entraînement au fil du temps. Cependant, des fluctuations peuvent être observées, atteignant un minimum local de 0.1624 à l'époque 14 avant de remonter

Chapitre 2 : Conception et implémentation

légèrement. La perte de validation suit une tendance similaire, bien que légèrement plus élevée que la perte d'entraînement, avec des valeurs allant de 0.3811 à 0.8713. Cela suggère que le modèle généralise raisonnablement bien sur les données de validation, bien que les pertes les plus basses se produisent souvent après plusieurs époques d'entraînement.

Exactitude (Accuracy) : L'exactitude commence à 0.8514 et augmente jusqu'à un pic de 0.8711 à l'époque 5, ce qui indique une bonne capacité du modèle à prédire correctement les sentiments sur les données d'entraînement. Cependant, l'exactitude montre des variations importantes au fil des époques, chutant à des valeurs plus basses telles que 0.1624 à l'époque 14 avant de remonter. Cela suggère des défis dans la capacité du modèle à maintenir une performance constante sur l'ensemble des données.

2.6.2. PHASE 2 : Génération Des Recommandations basées Préférences diversifiées

Dans cette section. On propose une nouvelle approche qui combine entre Filtrage Basé sur les Sentiments et Filtrage Basé sur la Catégorie et la Sous-catégorie

Donc L'algorithme de recommandation utilisé ici est basé sur les sentiments exprimés dans les commentaires des utilisateurs et exploite les informations de catégorie et de sous-catégorie pour fournir des recommandations pertinentes. En filtrant les commentaires positifs et en recommandant des cours de catégories similaires, l'algorithme vise à proposer des recommandations à la fois pertinentes et diversifiées.

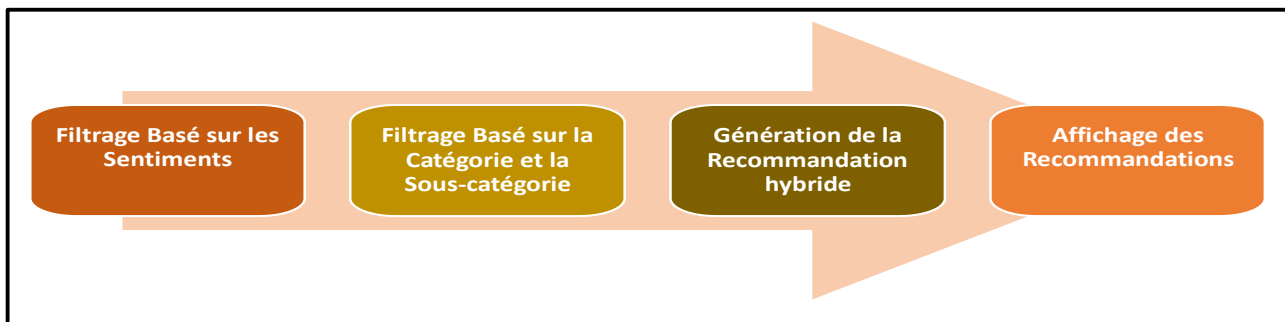


Figure 2.20. Les étapes de la recommandation

Dans ce qui suit, nous détaillons les différentes étapes du processus de la recommandation hybride basée sur les préférences diversifiées.

2.6.2.1. Le Filtrage Basé sur les Sentiments :

Dans cette première étape, les commentaires des apprenants et les données des cours sont importés à partir de fichiers CSV. Ensuite, les noms de colonnes sont nettoyés pour éliminer tout espace indésirable, garantissant ainsi la cohérence et la qualité des données.

Ensuite, les deux ensembles de données sont fusionnés en associant chaque commentaire à son cours correspondant. Cela est réalisé en fusionnant les datasets sur des clés communes, à savoir `course_id` pour les commentaires et `id` pour les cours. Cette fusion permet d'établir une relation directe entre les commentaires des apprenants et les cours auxquels ils se rapportent, facilitant ainsi l'analyse des sentiments et la recommandation de cours pertinents.

Chapitre 2 : Conception et implémentation

Processus de Recommandation Basé sur les Préférence :

Extraction des Préférences : À partir des résultats de notre modèle d'analyse de sentiments (bert base uncased), les commentaires sont classés en deux catégories : positifs et négatifs. Les commentaires sont ensuite segmentés en deux phases : recommandation traditionnelle et recommandation diversifiée.

Recommandation Traditionnelle : Pour les commentaires positifs classés dans la sous-catégorie de recommandation traditionnelle, les cours recommandés appartiennent tous à la même sous-catégorie que celle du commentaire positif. Cela garantit une recommandation directe et ciblée basée sur les préférences exprimées spécifiquement pour cette sous-catégorie.

Recommandation Diversifiée : Pour les commentaires positifs classés dans la sous-catégorie de recommandation diversifiée, un paramètre d'ajustement α est utilisé pour pondérer la recommandation entre les cours des sous-catégories positive et ceux des sous-catégories négatives. La formule est : $\text{Recommandation} = \alpha \cdot C_{\text{positifs}} + (1 - \alpha) \cdot C_{\text{négatifs}}$ où C_{positifs} représente les cours de la même sous-catégorie que le commentaire positif, et $C_{\text{négatifs}}$ représente les cours des autres sous-catégories.

Traitement des Différentes Sous-Catégories ; Lorsque les commentaires positifs sont classés dans différentes sous-catégories de cours, la recommandation favorise initialement les cours de la sous-catégorie positive. Toutefois, une partie des recommandations peut également provenir d'autres sous-catégories positives pour diversifier l'offre. Cela se fait en sélectionnant de 2 à 5 catégories différentes et en classant les recommandations selon la base de préférence d'apprenant.

Classement et Affichage : Les cours sont classés selon le degré de préférence attribué par le modèle. Les meilleurs cours sont alors affichés en fonction de leur classement, souvent présentés en top 10, 15 ou 20 pour offrir une sélection variée mais pertinente à l'utilisateur.

2.7. Conclusion

Dans ce chapitre, nous avons exposé les étapes de conception de notre système de recommandation basé sur l'élicitation des préférences. Notre approche, décomposée en deux phases, permet de générer des recommandations personnalisées dans un système de recommandation E-Learning, en se fondant sur les commentaires fournis par les apprenants. Grâce à l'application de techniques d'apprentissage automatique, nous avons pu comprendre les préférences des apprenants et proposer des recommandations adaptées à leurs besoins.

Les approches de recommandation hybride basé sur les sentiments et la similarité de contenu, pour générer des recommandations pertinentes.

Notre proposition a pour objectif d'améliorer l'expérience des utilisateurs au sein des plateformes de recommandation E-Learning.

Afin d'évaluer l'efficacité de notre solution, nous avons entrepris une série d'essais et de comparaison, dont les conclusions seront exposées dans le prochain chapitre.

3.1. Introduction :

Dans ce chapitre, nous présenterons les divers outils et plateformes utilisés pour la mise en œuvre de notre système, ainsi que les métriques d'évaluation employées dans nos expériences. Ces éléments nous permettront de répondre aux questions suivantes.

Question de recherche 1 – Le nouveau modèle BERT base uncased est-il efficace pour l'analyse des sentiments et l'enrichissement l'éducation de l'apprenant ?

Question de recherche 2 – Comment réaliser une recommandation pertinente et diversifier à L'apprenant pour personnaliser la processus d'apprentissage ?

3.2. Base de données (DataSet) :

Dans le but d'étudier les performances de notre modèle d'analyse des sentiments et de mieux comprendre les défis et les opportunités associés à l'élicitation des préférences des apprenants, nous avons utilisé deux ensembles de données qui contiennent des informations liées entre eux.

Le premier ensemble de données, composée de 9 millions d'échantillons, comprend 6 colonnes : "*id*", "*course_id*", "*rate*", "*date*", "*display name*", et "*comment*". Ces données sont des commentaires textuels en anglais, ce qui en fait une ressource précieuse pour l'élicitation des préférences des apprenants, ainsi que pour la formation et l'évaluation des modèles d'analyse des sentiments.

Pour des raisons d'insuffisance dans la capacité du matériel et du temps d'exécution, nous avons sélectionné **300 000** échantillons et après la suppression des commentaires qui ne sont pas en anglais, nous avons eu **185518** échantillons entre commentaires positifs et commentaires négatifs.

Le deuxième ensemble de données, que contient "*course_id*" et "*id*", contient des informations détaillées sur tous les cours disponibles sur **Udemy au 10 octobre 2022**. Ces données ont été obtenues à partir du fichier "*Course_info.csv*". De plus, plus de 9 millions de commentaires ont été collectés et fournis dans le fichier "*Comments.csv*".

Les informations de plus de **209 000** cours ont été collectées en extrayant des données du site Web **Udemy**. Ce dernier propose ainsi **209 734** cours enseignés par **73 514** instructeurs dans **79** langues différentes et répartis dans **13** catégories distinctes, aussi chaque catégorie y'a des sous-catégorie dans la dataset (subcategory).

3.3. Outils de développements :

3.3.1. Le Langage de programmation (python) :

Python est un langage de programmation polyvalent, apprécié pour sa simplicité et sa lisibilité. Sa syntaxe claire et concise en fait un choix idéal pour les débutants en programmation, mais sa puissance et sa flexibilité en font également un outil précieux pour les développeurs expérimentés.

Chapitre 3 : Expérimentations et tests

Pour ceux qui souhaitent s'initier à Python, le site officiel de Python (**W1**) propose une documentation complète ainsi que des tutoriels pour apprendre les bases du langage. Une fois familiarisé avec les fondamentaux, le livre "Automate the Boring Stuff with Python" par Al Sweigart [81] est une ressource précieuse pour apprendre à automatiser des tâches courantes avec Python. Python bénéficie d'un écosystème riche de bibliothèques et de frameworks, facilitant ainsi le développement dans une grande variété de domaines. La bibliothèque standard de Python offre de nombreux modules pour des tâches courantes, tandis que des bibliothèques tierces telles que NumPy et Pandas sont largement utilisées pour le calcul scientifique et l'analyse de données. Pour découvrir et installer des packages Python, le site web PyPI (**w2**) est une ressource incontournable.

La communauté Python est également active et engagée, offrant de nombreuses ressources pour poser des questions et trouver des réponses à travers des plateformes telles que Stack Overflow(**w3**). Enfin, Python est largement adopté dans l'industrie, utilisé pour le développement Web avec des frameworks comme Django et Flask, le développement d'applications scientifiques avec des outils comme SciPy et Matplotlib, et le développement d'applications d'apprentissage automatique avec des bibliothèques comme TensorFlow et PyTorch. En explorant ces ressources, vous découvrirez rapidement pourquoi Python est devenu l'un des langages de programmation les plus populaires au monde, et vous serez bien équipé pour maîtriser ses nombreuses applications.

3.3.2. Les bibliothèques utilisées :

La bibliothèque essentielle utilisée dans notre étude est *Transformers*. Elle est également connue sous le nom de *Hugging Face Transformers*, qui est un ensemble d'outils logiciels open-source développé par *Hugging Face*. Elle offre une interface simplifiée pour utiliser et affiner des modèles de transformateurs pré-entraînés dans le domaine du Traitement du Langage Naturel (Natural Language Processing NLP) avec une large gamme de modèles, tels que **BERT**, **GPT**, **RoBERTa** et **XLN**. Cette bibliothèque est adaptée à diverses tâches telles que la classification de texte, la reconnaissance d'entités et la génération de texte.

Elle prend en charge à la fois **PyTorch** et **TensorFlow** pour l'apprentissage en profondeur, et propose une API de haut niveau pour simplifier le chargement, l'ajustement et l'évaluation des modèles. Grâce à l'apprentissage par transfert, elle permet également un ajustement précis des modèles pré-entraînés sur des tâches spécifiques. En utilisant l'architecture des transformateurs basée sur des mécanismes d'auto-attention, la bibliothèque **Transformers** capture efficacement les informations contextuelles dans les données textuelles, ce qui en fait une ressource incontournable pour de nombreuses applications de NLP. [82]

Nous avons également utilisé d'autres bibliothèques telles que :

- **Pandas** : Une bibliothèque Python pour la manipulation et l'analyse de données tabulaires

Note : Pour télécharger
Transformers : <https://pypi.org/project/transformers/#files>

Chapitre 3 : Expérimentations et tests

- **Langdetect** : Un outil Python pour détecter automatiquement la langue d'un texte
- **Spacy** : Une bibliothèque Python pour le traitement avancé du langage naturel (NLP).
- **re**: Le module Python pour les expressions régulières, utilisé pour rechercher et manipuler des motifs de texte.
- **numpy**: Une bibliothèque Python pour le calcul numérique, offrant des tableaux multidimensionnels et des fonctions mathématiques.
- **os**: Le module Python pour les fonctionnalités liées au système d'exploitation, permettant d'interagir avec les fichiers et les répertoires.
- **sklearn**: Scikit-learn est une bibliothèque Python pour l'apprentissage automatique, offrant des outils pour la classification, la régression, le clustering, etc.
- **matplotlib** : Une bibliothèque Python pour la visualisation de données, permettant de créer des graphiques et des tracés de qualité publication.

3.3.3. Le matériel utilisé :

- **Modèle de l'ordinateur** : Hp Probook 450 15.6 inch G9 notebook pc
- **Mémoire RAM** : 8 Go de RAM.
- **Stockage**: SSD de 256 Go.
- **Processeur**: 12th Gen Intel(R) Core(TM) i5-1235U 1.30 GHz
- **Architecture** : Système d'exploitation 64 bits, processeur x64.
- **ID de périphérique** 9F37B251-4E12-4F6D-A4DB-F71A99B24A08
- **ID de produit** 00355-63032-07394-AAOEM

3.4. Les métriques d'évaluation :

Pour évaluer la performance de notre système, nous avons employé plusieurs métriques d'évaluation.

3.4.1. Accuracy :

Mesure le ratio des prédictions correctes par rapport aux prédictions totales, calculé en divisant la somme des vrais positifs (TP) et des vrais négatifs (TN) par le nombre total d'échantillons, incluant les vrais positifs, les vrais négatifs, les faux positifs (FP) et les faux négatifs (FN). En d'autres termes, l'accuracy représente la proportion d'échantillons correctement classés par rapport à l'ensemble total d'échantillons.

$$\text{Formule 1 : Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

- **TP (True Positives)** : correspond au nombre de prédictions positives correctes,
- **TN (True Negatives)** : correspond au nombre de prédictions négatives correctes,

- **FP (False Positives)** : correspond au nombre de prédictions positives incorrectes,
- **FN (False Negatives)** : correspond au nombre de prédictions négatives incorrectes.

3.4.2. Matrice de confusion:

La matrice de confusion est une table qui permet de visualiser les performances d'un modèle de classification en comparant les prédictions du modèle aux valeurs réelles. Elle présente le nombre de prédictions correctes et incorrectes pour chaque classe, notamment les vrais positifs (TP), les vrais négatifs (TN), les faux positifs (FP) et les faux négatifs (FN).

3.5. Evaluation du système bertec :

Dans cette section, nous présenterons en détail les résultats d'apprentissage de notre système et évaluerons notre modèle en le comparant à d'autres modèles testés sur le même jeu de données. Ensuite, nous fournirons une évaluation des recommandations générées par notre système.

3.5.1. Evaluation du modèle bert-base-uncased :

Avant de lancer l'apprentissage du système, on a ajusté plusieurs paramètres et les modifier après chaque entraînement jusqu'à avoir des résultats performants. Les paramètres optimaux sont présentés dans le **Tableau** :

Paramètre	Valeur
Nombre d'époques d'entraînement	3/10/15
Nombre de couches cachées utilisées	12
Taille du batch par périphérique Batch_size (training/evaluation)	16/8
Taux d'apprentissage (Learning_rate)	2e-5

Tableau 3. Les paramètres optimaux d'entraînement

3.5.1.1. Performance du système pour 3 époques :

On a commencé par entraîner le modèle sur 3 époques, les résultats obtenus sont présentés dans la Figure 3.1

Epoch	Training Loss	Validation Loss	Accuracy	F1	Precision	Recall
1	0.341300	0.315296	0.873345	0.873345	0.873345	0.873345
2	0.257000	0.285782	0.894829	0.894827	0.894858	0.894828
3	0.192800	0.336491	0.899825	0.899808	0.900087	0.899822

Figure 3.1. Performance du système en 3 époques.

Les informations indiquées montrent les performances du système dans les 3 époques. Les mesures d'exactitude augmentent régulièrement à chaque époque, ce qui démontre une amélioration des performances du modèle en termes de classification correcte des commentaires et de précision des prédictions des sentiments.

De plus, les valeurs de perte pour l'apprentissage et la validation diminuent au fil des époques, indiquant que le système apprend et réduit les erreurs. Les résultats sont présentés dans les **Figures 21** :

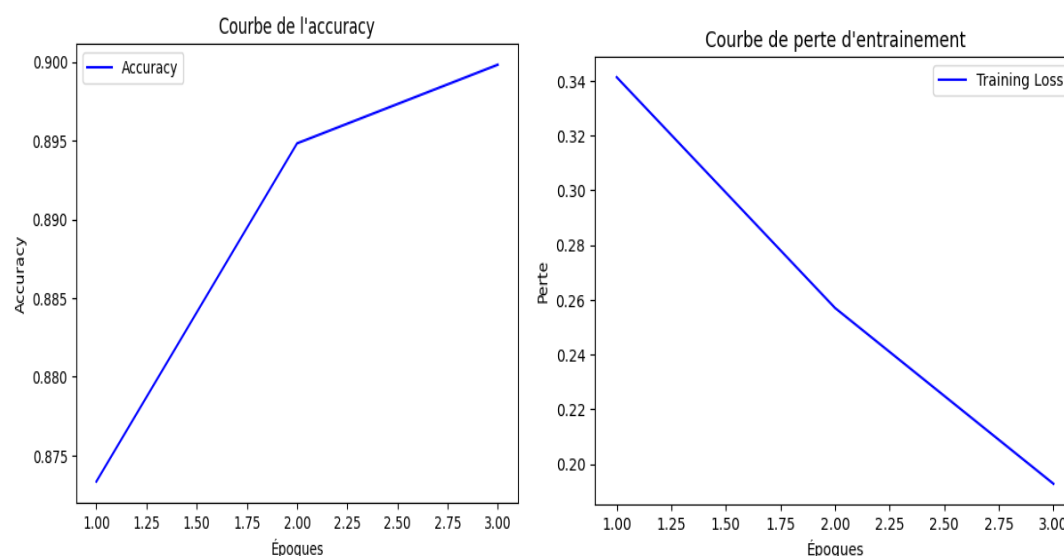


Figure 3.2. Perte Vs. Précision du système en 3 époques.

3.5.1.2. Performance du système pour 10 époques :

En observant le schéma d'amélioration des mesures à travers les époques, il est raisonnable de suggérer que l'augmentation du nombre d'époques pourrait conduire à des meilleurs résultats.

Le système semble capable d'apprendre plus à mesure des époques, ce qui suggère qu'une période d'entraînement plus longue pourrait permettre au système de continuer à s'améliorer.

Chapitre 3 : Expérimentations et tests

Toutefois, il est également important d'être vigilant quant au sur-apprentissage possible, le système risque de trop s'adapter aux données spécifiques d'apprentissage et de ne pas se généraliser de manière optimale à de nouvelles données.

On a donc lancé l'entraînement sur 10 époques, les résultats obtenus sont présentés dans la **Figure 3.3** :

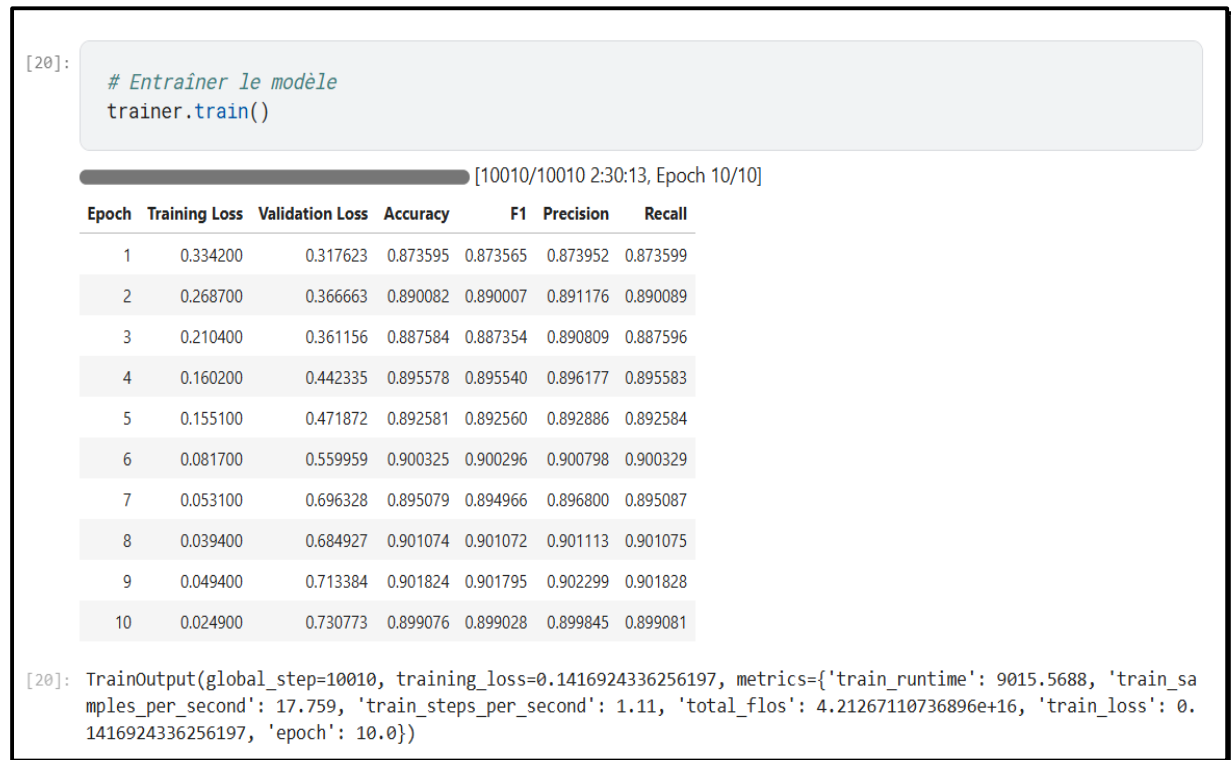


Figure 3.3. Performance du système en 10 époques.

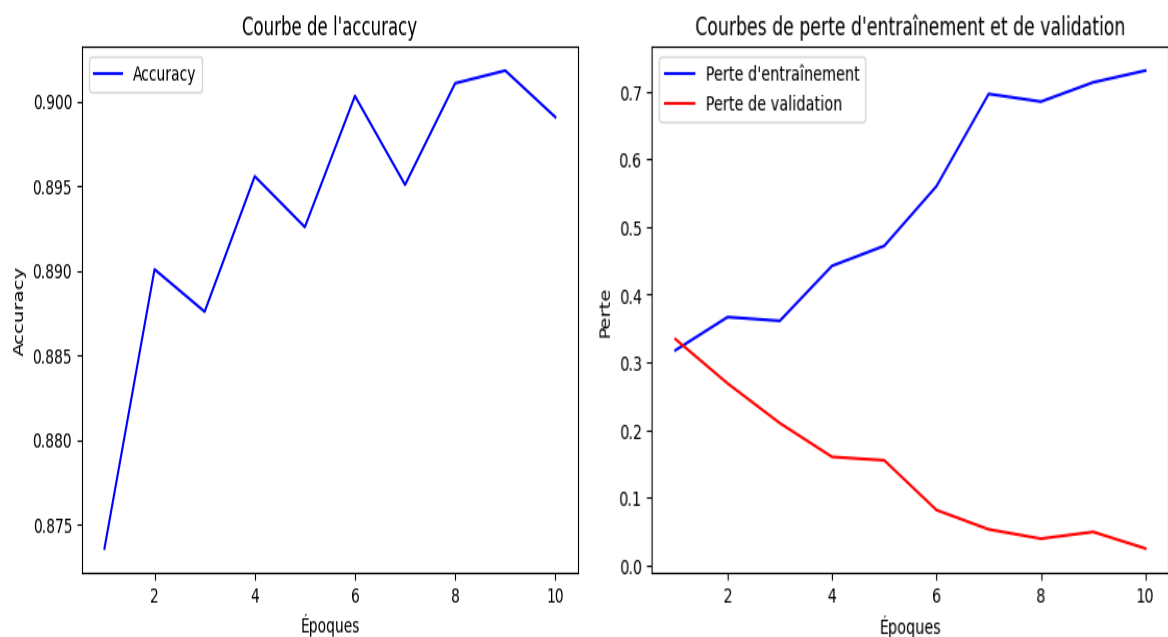


Figure 3.4. Précision Vs. Perte en 10 époques

Chapitre 3 : Expérimentations et tests

Les résultats illustrés dans la Figure ci-dessus montrent que la perte d'apprentissage diminue progressivement au fil des époques, ce qui suggère que le système s'adapte de mieux en mieux aux données d'apprentissage. Cependant, la perte de validation suit un schéma plus complexe, avec des augmentations à certaines époques, ce qui peut indiquer des difficultés du système à généraliser.

3.5.1.3. Performance du système dans 15 époques :

Les résultats de performance du système pour 15 époques sont présentés dans la Figure 3.5 :

Epoch	Training Loss	Validation Loss	Accuracy
1	0.537900	0.381124	0.851361
2	0.570100	0.427663	0.856857
3	0.244600	0.434392	0.869598
4	0.408100	0.439251	0.855109
5	0.263900	0.421878	0.871097
6	0.384100	0.465461	0.853360
7	0.407400	0.453306	0.850362
8	0.680600	0.711681	0.500125
9	0.192000	0.488448	0.854359
10	0.295200	0.422023	0.867849
11	0.307600	0.458051	0.847864
12	0.342800	0.465096	0.866600
13	0.303200	0.462149	0.868599
14	0.162400	0.465540	0.869348
15	0.215200	0.465808	0.871346

Figure 3.5. Performance du système en 15 époques.

Selon la Figure 24, on observe une amélioration continue des performances du système au fil du temps. La perte d'apprentissage diminue progressivement, indiquant que le système s'adapte bien aux données d'apprentissage. Toutefois, la perte de validation montre une certaine variation, ce qui peut indiquer que le système rencontre des difficultés de généralisation. De plus, on constate une légère baisse de l'exactitude (accuracy). Les différents résultats sont illustrés dans la Figure 3.6.

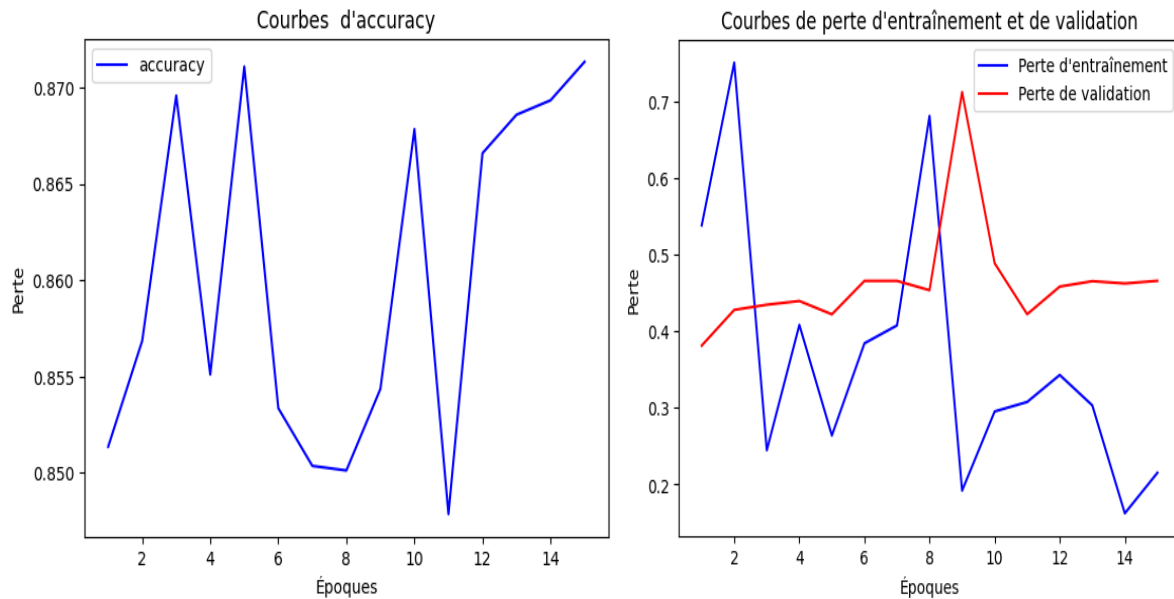


Figure 3.6. Précision Vs. Perte en 15 époques.

3.5.1.3. Discussion des résultats obtenus :

En comparant les résultats des trois premières époques avec ceux obtenus après dix et quinze époques, on observe une amélioration significative des mesures d'exactitude. Cela suggère que le système a continué à apprendre et à se perfectionner au fil des époques supplémentaires. Cependant, il est important de noter que la perte de validation a augmenté au cours de certaines époques supplémentaires, ce qui pourrait indiquer le début d'un sur-apprentissage. Cela souligne l'importance de surveiller attentivement les mesures de validation et d'appliquer des techniques de régularisation pour éviter le sur-apprentissage et améliorer la capacité du modèle à se généraliser correctement.

En conclusion, bien que l'entraînement du système sur un plus grand nombre d'époques ait permis d'améliorer les performances globales, il est essentiel de prendre en compte la perte de validation lors de l'évaluation de la capacité du modèle à bien se généraliser.

3.5.2. Teste de modèle bert base Uncased :

Exemples réels :

Exemple 1 :

Commentaire: instructor concept clearli explain excel wualiti video

Tokens:

[CLS]: 101

instructor: 9450

concept: 4145

clear: 3154

##li: 3669

explain: 4863

excel: 24970

wu: 8814

##ali: 11475

##ti: 3775

video: 2678

[SEP]: 102

Classe prédite: 1

Dans cet exemple le modèle transforme chaque mot en un identifiant unique appelé token. Par exemple, "instructor" devient 9450.

Les sous-mots sont également tokenisés, comme "clearli" devient "clear" (3154) et "##li" (3669).

"[CLS]" et "[SEP]" sont des tokens spéciaux utilisés par BERT pour marquer le début et la fin de la séquence.

La classe prédite "1" indique que ce commentaire est classé comme positif.

Exemple 2 :

Commentaire: Teaching quality may vary between instructors, affecting the consistency of learning

Tokens:

[CLS]: 101

teaching: 4252

quality: 3737

may: 2089

vary: 8137

between: 2090

instructors: 19922

,: 1010

affecting: 12473

the: 1996

consistency: 18700

of: 1997

learning: 4083

[SEP]: 102

Classe prédite: 0

Dans cet exemple Chaque mot et ponctuation est converti en tokens. Par exemple, "teaching" devient 4252 et "," devient 1010. La classe prédite "0" indique que ce commentaire est classé comme négatif.

Exemple 3:

Commentaire: The model encounters difficulty in correctly identifying the language of certain texts, leading to unreliable language classifications.

Tokens:

[CLS]: 101

the: 1996

Chapitre 3 : Expérimentations et tests

model: 2944
encounters: 11340
difficulty: 7669
in: 1999
correctly: 11178
identifying: 12151
the: 1996
language: 2653
of: 1997
certain: 3056
texts: 6981
,: 1010
leading: 2877
to: 2000
unreliable: 23579
language: 2653
classifications: 26739
.: 1012
[SEP]: 102
Classe prédite: 0

Dans cette exemple

Le modèle identifie les tokens pour chaque mot et les transforme en identifiants uniques.

La répétition de mots, comme "the" (1996) et "language" (2653), montre comment le modèle gère le contexte.

La classe prédite "0" indique que le commentaire est perçu comme négatif.

La figure suivante explique le teste de notre modèle :

```
from transformers import BertTokenizer, BertForSequenceClassification
import torch
import os

# Définir le chemin correct
model_path = "/content/drive/MyDrive/newmodel/bert_base_uncased_train"
tokenizer = BertTokenizer.from_pretrained(model_path)
model = BertForSequenceClassification.from_pretrained(model_path)

# Tester le modèle avec un exemple de texte
test_text = "The model faces challenges in accurately determining its classification type, often resulting in inconsistent predictions."

# Affichage du texte original
print(f"Commentaire: {test_text}")

# Tokenization du texte
inputs = tokenizer(test_text, return_tensors="pt")

# Affichage des tokens à partir des input_ids
print("Tokens:")
for token_id in inputs['input_ids'][0]:
    token = tokenizer.convert_ids_to_tokens(token_id.item())
    print(f"{token}: {token_id}")

# Désactiver le calcul des gradients pour éviter d'utiliser trop de mémoire
with torch.no_grad():
    # Faire la prédiction
    outputs = model(**inputs)
    logits = outputs.logits
    predicted_class = torch.argmax(logits, dim=1).item()

# Affichage de la classe prédite
print(f"Classe prédite: {predicted_class}")
```

Figure 3.7. Le teste de modèle bert base uncased

En analysant plusieurs commentaires, notre modèle est capable de faire des prédictions précises. Chaque commentaire contribue à affiner sa prédiction, offrant ainsi une évaluation détaillée de l'expérience utilisateur. Cette capacité à comprendre différents signaux textuels aide à prévoir les réactions des utilisateurs et à identifier les domaines où des améliorations sont nécessaires.

3.5.3. Comparaison des résultats :

Pour démontrer la performance de notre modèle, nous avons sélectionné deux modèles parmi les modèles du deep learning les plus renommés dans le domaine du traitement automatique du langage naturel et de la classification : le modèle **LSTM** et le modèle **BiLSTM**.

Le modèle **LSTM** (*Long Short-Term Memory*) comprend **une couche d'embedding** pour convertir les entiers en vecteurs denses, suivie de couches **LSTM** pour capturer les dépendances séquentielles dans les données.

Les couches **LSTM** traitent la séquence d'entrée tout en maintenant la mémoire à long terme, ce qui est essentiel pour comprendre le contexte des commentaires des utilisateurs. Des couches de **dropout** sont ajoutées pour éviter le **surapprentissage** en désactivant aléatoirement des neurones pendant l'entraînement. Enfin, une couche **dense** avec une **activation sigmoid** génère les probabilités de classe, permettant la classification binaire des sentiments exprimés dans les commentaires.

Le modèle **BiLSTM** (*Bidirectional Long Short-Term Memory*) est constitué **d'une couche d'embedding** qui convertit les entiers en **vecteurs denses**. Ce modèle exploite des couches **LSTM bidirectionnelles**, permettant de capturer les dépendances séquentielles dans les données en analysant les séquences d'entrée dans les deux directions (avant et arrière).

Chapitre 3 : Expérimentations et tests

Cette approche améliore la compréhension contextuelle par rapport aux modèles **LSTM** unidirectionnels classiques. Des couches de **dropout** sont intégrées après chaque couche **BiLSTM** pour prévenir le **surapprentissage**. Enfin, une couche **dense** avec une **activation sigmoid** est utilisée pour générer les probabilités de classe, facilitant ainsi la classification binaire des sentiments exprimés dans les commentaires des utilisateurs.

Avant de comparer les résultats obtenus par notre modèle avec ceux des deux modèles existants, nous allons d'abord présenter brièvement leurs performances en apprentissage et en test. Ensuite, nous allons comparer ces résultats avec ceux de notre système.

Les sections suivantes détaillent les résultats de chaque modèle de manière distincte

3.5.3.1 Modèle 1 : Modèle LSTM :

Architecture	LSTM (128 unités) x2, dropout x2, dense sigmoid
Entraînement	15 époques
Précision d'apprentissage	0.9616
Précision sur les tests	0.8728

Tableau 4. Résultats du modèle avec architecture LSTM et dropout

Le tableau décrit un modèle de classification basé sur deux couches LSTM de 128 unités chacune, avec des couches de dropout pour prévenir le surapprentissage. Le modèle utilise une couche dense avec une activation sigmoïde pour la classification binaire. Entraîné sur 15 époques, il a atteint une précision de 96.16% sur les données d'apprentissage et de 87.28% sur les données de test, montrant une bonne capacité de généralisation tout en maintenant des performances élevées. Les résultats sont présentés dans la Figure 27 :

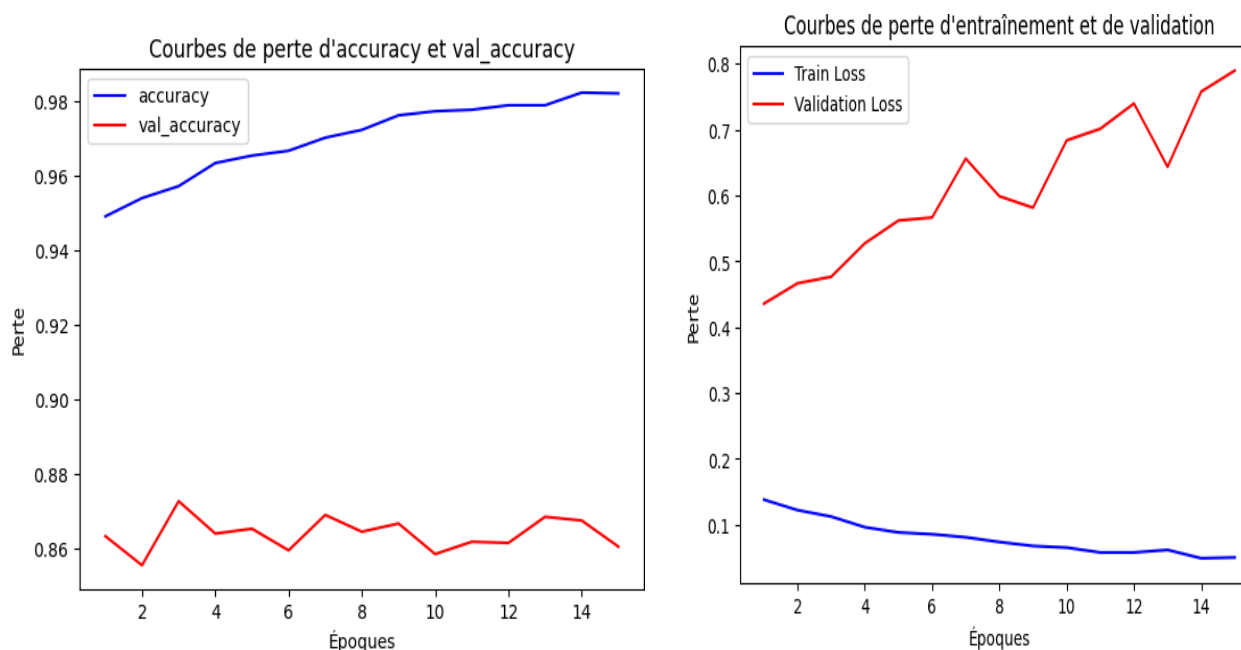


Figure 3.8. Performance du modèle LSTM

Chapitre 3 : Expérimentations et tests

D'après les résultats obtenus après l'entraînement du modèle pendant 15 époques, on observe une augmentation progressive de la précision d'entraînement, atteignant jusqu'à 0,9822. Cependant, les résultats de validation montrent une tendance différente. La perte de validation fluctue et augmente légèrement vers la fin, tandis que la précision de validation stagne autour de 0,8606. Cela suggère que le modèle pourrait être en train de sur-ajuster les données d'entraînement, ce qui entraîne une généralisation insuffisante sur les nouvelles données.

3.5.3.1. Modèle 2 : Le modèle BiLSTM:

Architecture	Couche intégration, BiLSTM (64 unités) x2, dropout x2, dense sigmoid
Entraînement	15 époques
Précision d'apprentissage	0.96
Précision sur les tests	0.86

Tableau 5. Résultats du modèle avec architecture Bidirectionnelle LSTM et dropout

Le tableau présente un modèle de classification utilisant une architecture comprenant une couche d'intégration suivie de deux couches BiLSTM (64 unités chacune), et des couches de dropout pour éviter le surapprentissage. La couche dense avec activation sigmoïde effectue la classification binaire. Après 15 époques d'entraînement, le modèle a atteint une précision de 96% sur les données d'apprentissage et 86% sur les données de test, indiquant de solides capacités de généralisation tout en maintenant des performances globalement élevées

Les résultats obtenus avec une architecture composée d'une couche d'intégration sont présentés dans la **Figure 3.9**

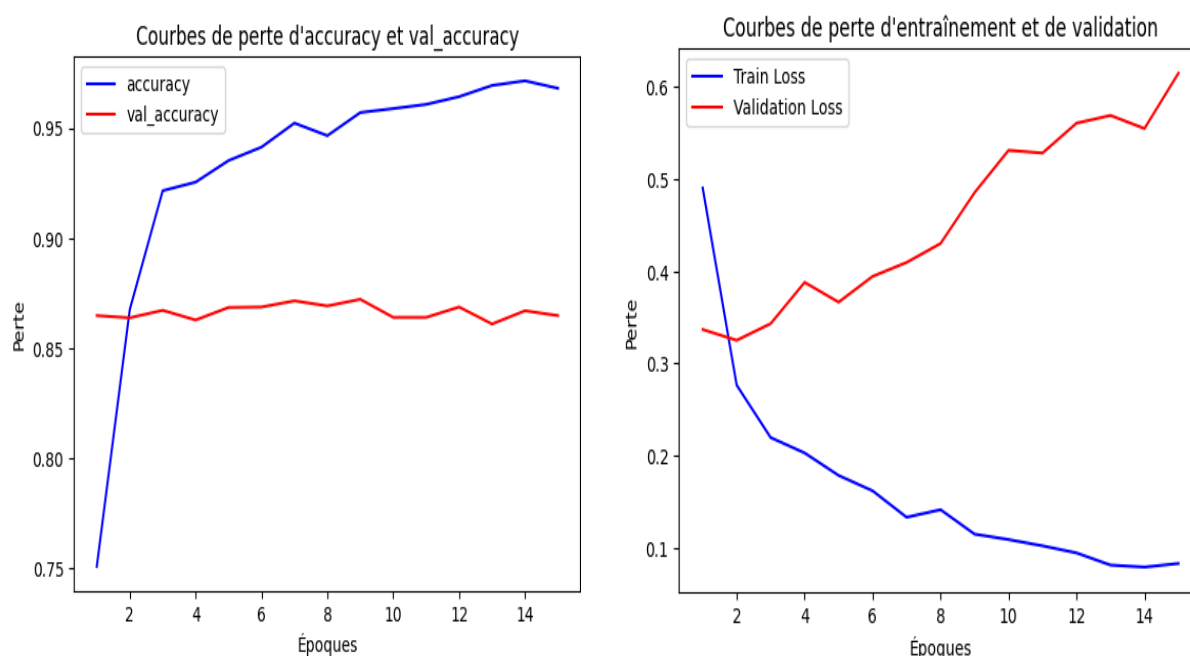


Figure 3.9. Performance du modèle BiLSTM.

Après 15 époques d'entraînement, le modèle a montré une amélioration significative de ses performances sur les données d'entraînement, avec une diminution progressive de la perte

d'entraînement. Cependant, les résultats sur les données de validation ont été moins satisfaisants, avec une légère augmentation de la perte de validation. Malgré cela, le modèle a atteint une exactitude de 0.96 sur les données d'entraînement et une exactitude de validation de 0.86, indiquant une capacité de généralisation raisonnable.

3.5.3.3. Comparaison entre les trois modèles :

La Figure 3.10 présente les résultats de performance des trois modèles afin d'avoir une vue claire sur la différence entre eux et d'exprimer une comparaison entre eux.

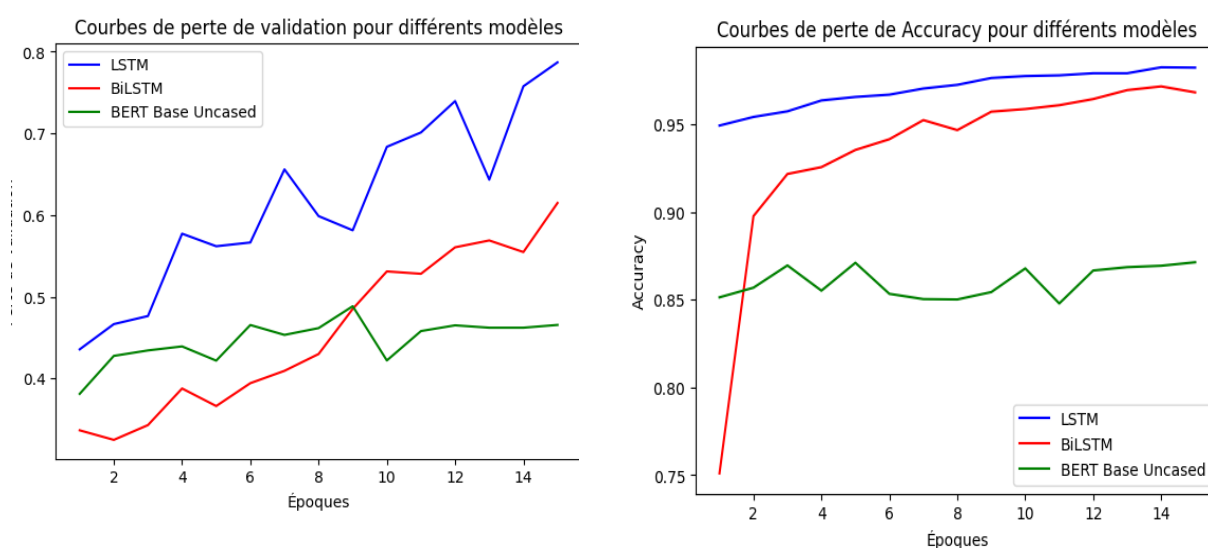


Figure 3.10. Comparaison entre les trois modèles.

Modèle	LSTM	base_uncased	Bilstm
Train Accuracy	0.9616	0.8713	0.9681
Test Accuracy	0.8728	0.8890	0.8723

Tableau 6. Les résultats des 3 modèle

3.6. Génération Des Recommandations basées Préférences diversifiées et normale:

Avant d'évaluer les recommandations générées, il est d'abord nécessaire d'expliquer le choix des valeurs optimales pour les deux paramètres utilisés dans la recommandation :

- Le paramètre **k** (nombre de cours recommandées)
- Alpha (α) est un paramètre numérique qui ajuste le poids relatif d'une méthode de recommandation par rapport à une autre lors de la combinaison des recommandations.

3.6.1. Réglage des paramètres :

Pour évaluer cette recommandation, nous appliquerons trois valeurs distinctes pour K (5, 8 et 10) ainsi que trois valeurs pour alpha (0.6, 0.5 et 0.8), nous calculons la précision et le rappel pour chaque utilisateur, puis en fait la moyenne pour obtenir les scores globaux pour tous les utilisateurs.

Les résultats de calculs présenter dans les courbes suivantes :

➤ **Pour $\alpha=0.5$:**

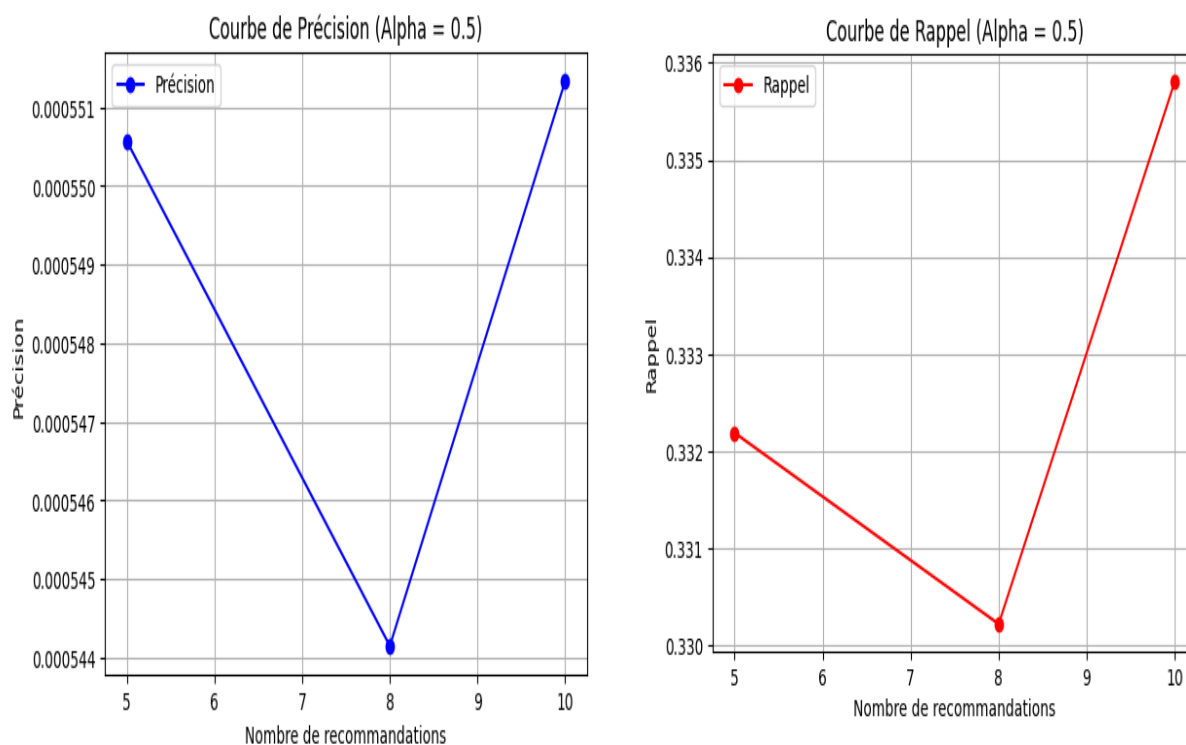


Figure 3.11. Courbe de précision rappel pour $\alpha= 0.5$

Pour $\alpha = 0.5$, nous avons les valeurs suivantes pour la précision et le rappel avec différents nombres de cours recommandés (K)

K = 5 : Précision : 0.0005505733147015178 , Rappel : 0.33219145354769075

K = 8 : Précision : 0.0005441504131154473 , Rappel : 0.3302231027770269

K = 10 : Précision : 0.0005513384951274691 , Rappel : 0.33580932719567536

3.6.1.1. Analyse des résultats :

Les résultats montrent que pour $\alpha = 0.5$, augmenter K de 5 à 10 n'affecte pas beaucoup la précision. Cependant, le rappel est un peu meilleur avec un K plus élevé. Cela signifie que pour un meilleur rappel, K = 10 pourrait être préféré, même si la différence est petite. Comme la précision ne change pas beaucoup, le choix de K pourrait dépendre d'autres facteurs, comme le nombre de recommandations et les besoins des utilisateurs.

➤ **Pour alpha 0.6 :**

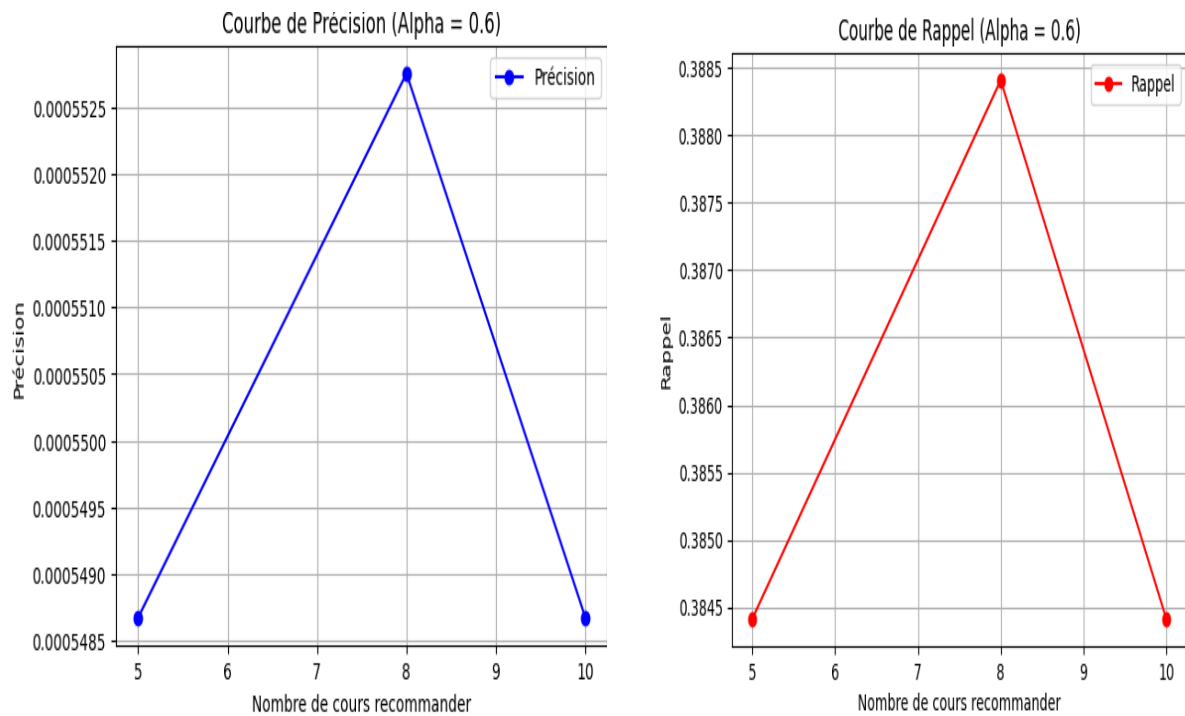


Figure 3.12. Courbe de précision rappel pour alpha= 0.6

Pour $\alpha = 0.6$, nous avons les valeurs suivantes pour la précision et le rappel avec différents nombres de cours recommandés (K) :

K = 5 : Précision : 0.0005486741066603667	Rappel : 0.3844139394771743
K = 8 : Précision : 0.0005527520943349455	Rappel : 0.38840935428909557
K = 10 : Précision : 0.0005486741066603667	Rappel : 0.3844139394771743

Analyse des résultats :

Pour $\alpha = 0.6$, augmenter K à 8 offre un léger avantage en termes de précision et de rappel. Les valeurs de K = 5 et K = 10 donnent des résultats très similaires, suggérant que K = 8 pourrait être le choix optimal pour équilibrer précision et rappel.

Pour $\alpha = 0.6$, lorsque nous utilisons K = 8, nous constatons une légère amélioration de la précision et du rappel par rapport à K = 5 et K = 10. Cela indique que K = 8 semble trouver un bon équilibre entre obtenir des recommandations précises et maximiser le rappel. Bien que K = 5 et K = 10 produisent des résultats similaires, K = 8 offre un léger avantage, en en faisant un choix avisé pour une performance optimisée.

➤ Pour $\alpha = 0.8$:

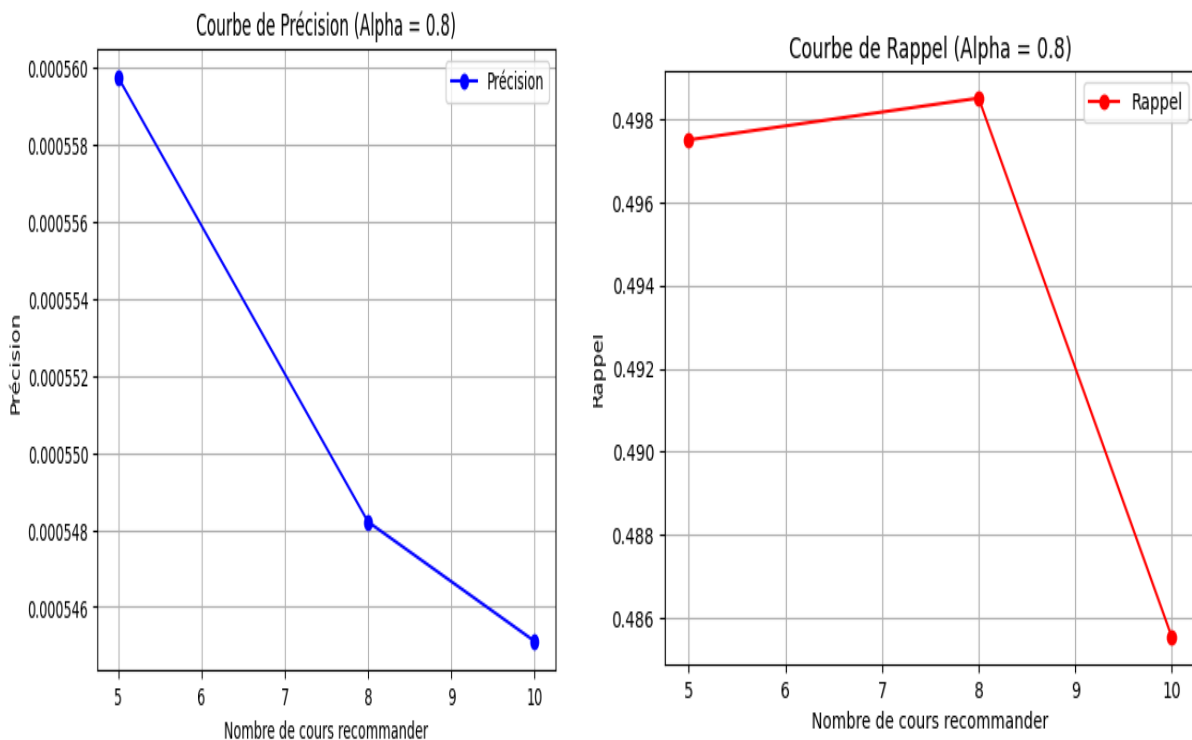


Figure 3.13. Courbe de précision rappel pour $\alpha = 0.8$

Précision :

$K=5$, la précision est 0.0005597489966453284. Le rappel est 0.4975052362599318

$K = 8$, la précision est 0.0005482069464953453 le rappel est 0.4985099193370231

$K = 10$ la précision est 0.000545107334593504 le rappel est 0.48553710972390046

On observe une légère diminution de la précision lorsque K augmente de 5 à 10.

Le rappel semble relativement stable pour les valeurs de K testées, avec une légère variation entre $K = 5$ et $K = 10$.

En général, une précision plus élevée indique une meilleure capacité du modèle à recommander des cours pertinents parmi ceux proposés. Un rappel élevé, quant à lui, suggère que le modèle parvient à capturer une grande partie des cours pertinents disponibles pour chaque utilisateur. Dans ce cas, bien que la précision diminue légèrement avec $K = 8$ et $K = 10$, le rappel reste relativement stable, ce qui pourrait indiquer que ces valeurs de K capturent efficacement les cours pertinents tout en maintenant une bonne couverture des recommandations.

Donc

Pour $\alpha = 0.5$, la précision reste relativement constante autour de 0.00055 pour toutes les valeurs de K testées (5, 8, 10). Cela suggère que la pondération des commentaires par rapport

à d'autres critères n'influence pas significativement la précision des recommandations. En revanche, le rappel montre une légère amélioration avec un K plus élevé, passant de 0.330 à 0.336. Ainsi, un K plus élevé pourrait légèrement augmenter la capacité du système à rappeler des cours pertinents pour les utilisateurs.

Pour $\alpha = 0.6$, les résultats montrent une stabilité similaire en termes de précision autour de 0.00055 pour $K = 5, 8$ et 10. Cependant, le rappel présente une légère augmentation avec K plus élevé, passant de 0.384 à 0.388. Cela indique que pour $\alpha = 0.6$, un K plus élevé pourrait légèrement améliorer la capacité du système à rappeler davantage de cours pertinents, tout en maintenant une précision stable.

Enfin, pour $\alpha = 0.8$, la précision reste stable autour de 0.00055 pour toutes les valeurs de K testées. Le rappel montre une légère diminution avec K plus élevé, passant de 0.485 à 0.486. Cela suggère que pour $\alpha = 0.8$, un K plus faible pourrait être optimal pour maintenir un bon rappel tout en conservant une précision stable.

Conclusion : En se basant sur les résultats de rappel qui montrent une légère amélioration pour $\alpha = 0.6$ avec $K = 8$, il semble que cette valeur d' α pourrait être légèrement préférable pour optimiser les recommandations. Cependant, la différence entre les valeurs d' α est assez minime. Le choix final pourrait dépendre d'autres considérations spécifiques aux besoins et aux priorités du système de recommandation, ainsi que des compromis entre précision et rappel.

3.8. Conclusion

Dans ce chapitre, nous avons présenté les outils et bibliothèques utilisés pour mettre en œuvre notre système. Nous avons également conduit une série de tests pour évaluer ses performances. Ensuite, nous avons comparé notre système aux modèles les plus couramment utilisés, tels que LSTM et BiLSTM, pour évaluer son efficacité dans l'élicitation des préférences et dans l'approche de recommandation hybride proposée.

Les résultats obtenus ont été généralement satisfaisants, mettant en lumière les avantages et les limitations de chaque approche. Ces résultats ouvrent de nouvelles perspectives de recherche qui seront explorées dans la conclusion.

Conclusion générale :

Ce mémoire porte sur le domaine du Traitement du Langage Naturel et plus spécifiquement de l'élicitation des préférences des apprenants pour personnaliser le processus d'apprentissage dans les MOOC. Notre étude se focalise sur l'analyse des sentiments en utilisant le modèle BERT Base Uncased, récemment développé par Google, pour optimiser les requêtes, la prédiction et la génération de textes en anglais. De plus, nous utilisons ces analyses de sentiments pour améliorer la qualité des recommandations fournies.

Ce travail développe des méthodes de traitement du langage naturel en anglais pour mieux répondre aux divers besoins des apprenants dans les MOOCs. En mettant l'accent sur la diversité des préférences, nous cherchons à ouvrir de nouvelles perspectives pour une éducation en ligne plus inclusive et efficace, offrant ainsi des opportunités d'apprentissage enrichissantes pour tous les participants.

Par la suite, nous avons exposé notre méthode qui implique la création d'une solution pour analyser les sentiments (positifs/négatifs) à partir des commentaires des apprenants anglophones. Pour ce faire, nous avons employé l'algorithme BERT base uncased, un modèle d'apprentissage profond, afin d'extraire les préférences (sentiments) à partir des commentaires fournis par les apprenants.

Ensuite, les sentiments et les préférences prédits ont été intégrés dans une nouvelle méthode de recommandation basée sur les préférences des utilisateurs, utilisant un algorithme basé sur les sous-catégories des cours. L'accent est mis sur l'exploitation des sentiments exprimés dans les commentaires des apprenants pour recommander des cours diversifiés et pertinents. L'algorithme intègre également les sous-catégories des cours pour diversifier les recommandations tout en assurant leur pertinence.

Puis, une série de tests et d'expérimentations a été réalisée pour illustrer nos propositions et démontrer les performances du système. Les résultats de notre modèle basé sur l'analyse de sentiments montrent des performances de classification relativement bonnes, surpassant celles des modèles LSTM et BiLSTM. De plus, la recommandation combinant le Filtrage Basé sur les Sentiments et le Filtrage Basé sur la Catégorie, utilisant le modèle BERT base uncased, s'est avérée plus précise et pertinente par rapport à celle générée en utilisant le simple Filtrage Basé sur la Catégorie.

Pour améliorer la performance de notre système de recommandation, plusieurs axes de développement sont envisagés comme perspectives, sachant :

- L'adaptation du modèle à d'autres langues permettrait de répondre aux besoins diversifiés des utilisateurs à travers différentes régions.
- Intégration des techniques de recommandation inter-domaines pour résoudre le problème du démarrage à froid améliorerait l'efficacité des recommandations, même pour les nouveaux apprenants.
- Tester le système sur des ensembles de données plus vastes permettrait d'évaluer sa performance dans des environnements variés et avec une diversité accrue de données. Ces initiatives visent à renforcer la précision, la pertinence et l'adaptabilité du système,

contribuant ainsi à améliorer de manière continue l'expérience d'apprentissage en ligne pour les apprenants.

A. Références Bibliographiques :

- [1] Bates, A. W., & Poole, G. (2003). *Effective teaching with technology in higher education: Foundations for success*. Wiley.
- [2] [Garrison, D. R., & Vaughan, N. D. (2013). *E-Learning in the 21st century: A framework for research and practice*.
- [3] J, Annie. « Créer de la présence à distance en e-learning. Cadre théorique, définition, et dimensions clés », *Distances et savoirs*, vol. 8, no. 2, 2010, pp. 257-274. consulté le 28/04/2024
- [4] Bates, A. W., & Poole, G. (2003). *Effective teaching with technology in higher education: Foundations for success*. Wiley.
- [5] Garrison, D. R., & Vaughan, N. D. (2013). *E-Learning in the 21st century: A framework for research and practice*. Routledge.
- [6] Siemens, G. (2005). *Connectivism: A learning theory for the digital age*. *International journal of instructional technology and distance learning*, 2(1), 3-10.
- [9] Anderson, T. (2003). Getting the mix right again: An updated and theoretical rationale for interaction. *The International Review of Research in Open and Distributed Learning*, 4(2).
- [10] Bel F Le e-learning et l'expression écrite : cas des étudiants de première année licence langue française de l'université 8 Mai 1945-Guelma
- [11] Daniel, J. (2012). Making sense of MOOCs: Musings in a maze of myth, paradox and possibility. *Journal of interactive Media in education*, 3(18), 2012-18.
- [12] Wang, Y., Chen, Z., & Li, N. (2019). Challenges and Opportunities in Preference Elicitation for MOOCs. *Proceedings of the 12th International Conference on Educational Data Mining (EDM)*, 123-130
- [13] Daniel, J. (2012). Making sense of MOOCs: Musings in a maze of myth, paradox and possibility. *Journal of interactive Media in education*, 3(18), 2012-18.
- [14] Milligan, C., Littlejohn, A. and Margaryan, A. (2013) Patterns of engagement in connectivist MOOCs. *Journal of Online Learning and Teaching*, 9(2), pp. 149-159.
- [15] Christensen, G., Steinmetz, A., Alcorn, B., Bennett, A., Woods, D., & Emanuel, E. J. (2013). *The MOOC Phenomenon: Who Takes Massive Open Online Courses and Why?* Retrieved from https://www.academia.edu/36415503/The_MOOC_Phenomenon_Who_Takes_Massive_Open_Online_Courses_and_Why

- [16] Siemens, G. (2005). Connectivism: A Learning Theory for the Digital Age. *International Journal of Instructional Technology and Distance Learning*, 2(1), 3-10.
- [17] Siemens, G. (2005). Connectivism: A Learning Theory for the Digital Age. *International Journal of Instructional Technology and Distance Learning*, 2(1), 3-10.
- [18] Bozkurt, A., Akgün-Özbek, E., & Zawacki-Richter, O. (2017). Trends and Patterns in Massive Open Online Courses: Review and Content Analysis of Research on MOOCs (2008-2015). *The International Review of Research in Open and Distributed Learning*, 18(5).
- [19] McAuley, A., Stewart, B., Siemens, G., & Cormier, D. (2010). The MOOC Model for Digital Practice.
- [20] Baggaley, J. (2013). MOOC rampant: Massive Open Online Courses. *Distance Education*, 34(2), 251-264.
- [21] Liyana gunawardena, T. R., Adams, A. A., & Williams, S. A. (2013). MOOCs: A Systematic Study of the Published Literature 2008-2012
- [22] Jacqmin, Julien, belleflamme, paul Les plateformes MOOCs. Menaces et opportunités pour l'enseignement universitaire 2014 • In *Regards Economiques*, 110
- [23] Paul Belleflamme Julien Jacqmin Les plateformes MOOCs. Menaces et opportunités pour l'enseignement universitaire.
- [24] Roediger et Karpicke (2006a et b), Karpicke et Roediger (2008) ou Karpicke et Blunt (2011)
- [25] Markoff, J. (2013). Essay-grading software offers professors a break. *The New York Times*, 20(3), 1-4.
- [26] Jacqmin, J. (2014). Les plateformes MOOCs. Menaces et opportunités pour l'enseignement universitaire. *Regards économiques*, 110.
- [27] Carrier, C. (2009). L'enseignement de l'entrepreneuriat : au delà des cours magistraux, des études de cas et du plan d'affaires 1. *Revue de l'Entrepreneuriat*, 8(2), 17-33.
- [29] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.
- [29] Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist*, 38(1), 43-52.
- [30] Mayer, R. E. (2001). *Multimedia learning*. Cambridge University Press.
- [31] OSTERWALDER, Alexander et PIGNEUR, Yves. *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons, 2010.

- [32] Moreno, R., & Mayer, R. E. (2007). Interactive multimodal learning environments. *Educational Psychology Review*, 19(3), 309-326.
- [33] Sweller, J., van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3), 251-296.
- [34] HOLLANDS, Fiona M. et TIRTHALI, Devayani. MOOCs: Expectations and Reality. Full Report. *Online Submission*, 2014.
- [36] CETINA, Iuliana, GOLDBACH, Dumitru, et MANEA, Natalia. Udey: a case study in online education and training. *Revista Economică*, 2018, vol. 70, no 3, p. 46-54.
- [37] Omandji Lokonde, Paul, et François Durrieu. « Business Model Canvas et pérennité des PME en environnement hostile : Une méta-analyse en contexte congolais », *Management & Sciences Sociales*, vol. 35, no. 2, 2023, pp. 74-91.
- [38] SAADATDOOST, Robab, SIM, Alex Tze Hiang, JAFARKARIMI, Hosein, *et al.* Understanding the Setting of a MOOC: A Journey into Coursera. *International Journal of Information and Communication Technology Education (IJICTE)*, 2016, vol. 12, no 1, p. 77-98.
- [39] Hichem Mendac, massinissa Mohellebi. Evaluation de formative et modelisation de l'apprenant dans un mooc. 30/06/2016
- [40] Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist*, 38(1), 43-52.
- [41] Mayer, R. E. (2001). *Multimedia learning*. Cambridge University Press.
- [42] Moreno, R., & Mayer, R. E. (2007). Interactive multimodal learning environments. *Educational Psychology Review*, 19(3), 309-326.
- [43] Sweller, J., van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3), 251-296.
- [44] Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist*, 38(1), 43-52.
- [45] Clark, R. C., & Mayer, R. E. (2016). *E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning*. John Wiley & Sons.
- [46] MANOUSELIS, Nikos, DRACHSLER, Hendrik, VERBERT, Katrien, *et al.* *Recommender systems for learning*. Springer Science & Business Media, 2012.
- [47] ADOMAVICIUS, Gediminas et TUZHILIN, Alexander. Toward the next of generation recommender systems: A survey of the state-of-the-art and possible

extensions. *IEEE transactions on knowledge and data engineering*, 2005, vol. 17, no 6, p. 734-749.

[48] SARWAR, Badrul, KARYPIS, George, KONSTAN, Joseph, *et al.* Item-based collaborative filtering recommendation algorithms. In : *Proceedings of the 10th international conference on World Wide Web*. 2001. p. 285-295.

[49] Stamatis, D., & Chorianopoulos, K. (2016). Ontology-driven personalized e-learning recommendations using semantic similarity. *Journal of Educational Technology & Society*, 19(2), 65-77.

[50] DEN HENGST, Floris, GRUA, Eoin Martino, EL HASSOUNI, Ali, *et al.* Reinforcement learning for personalization: A systematic literature review. *Data Science*, 2020, vol. 3, no 2, p. 107-147.

[47] Georges-Louis Baron. Applications et usages éducatifs des technologies de l'information et de la communication : éléments pour une analyse de la conjoncture scientifique. 2003. <edutice-00869506>

[48] Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3), 56-58.

[49] Ricci, F., Rokach, L., & Shapira, B. (2015). *Introduction to recommender systems handbook*. Springer.

[50] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.

[51] HERLOCKER, Jon, KONSTAN, Joseph A., *et* RIEDL, John. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 2002, vol. 5, p. 287-310.

[52] ROMERO, Cristobal *et* VENTURA, Sebastian. Educational data mining: A survey from 1995 to 2005. *Expert systems with applications*, 2007, vol. 33, no 1, p. 135-146.

[53] BRUSILOVSKY, Peter, SOSNOVSKY, Sergey A., YUDELSON, Michael, *et al.* Interactive Authoring Support for Adaptive Educational Systems. In : *AIED*. 2005. p. 96-103.

[54] RICCI, Francesco, ROKACH, Lior, *et* SHAPIRA, Bracha. *Introduction to recommender systems handbook*. In : *Recommender systems handbook*. Boston, MA : springer US, 2010. p. 1-35.

[55] MACCANN, Carolyn, JIANG, Yixin, BROWN, Luke ER, *et al.* Emotional intelligence predicts academic performance: A meta-analysis. *Psychological bulletin*, 2020, vol. 146, no 2, p. 150.

[56] CARVALHO JUNIOR, Giovani Lemos De, CEBRIÁN-ROBLES, Daniel, CEBRIÁN-DE-LA-SERNA, Manuel, *et al.* Comparative Study SPOC vs. MOOC for Socio-Technical Contents from Usability and User Satisfaction. 2019.

[58] Hui Han, Wenjie Li, Jian Cao, and Zhiwen Yu User Profiling for Personalized Learning Systems: A Survey Auteurs : Journal : IEEE Access Année : 2018 Pages : 14040-14055 DOI : 10.1109/ACCESS.2018.2819462

[59] Herlocker, J.L., Konstan, J.A., Terveen, L.G., & Riedl, J.T. (2004). "Evaluating collaborative filtering recommender systems." *ACM Transactions on Information Systems (TOIS)*, 22(1), 5-53.

[60] NURHIDAYATI, Anita. The Use of ALISON in ESP Classroom. *for Specific Pusposes using Alison Platform*, 2021, p. 11.

[61] H. Sobhanam et A. K. Mariappan, « Addressing cold start problem in recommendersystems using association rules and clustering technique », in 2013 International Conference on Computer Communication and Informatics, Coimbatore, Tamil Nadu, India: IEEE, janv. 2013, p. 1-5. doi: 10.1109/ICCCI.2013.6466121.

[62] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5-53.

[63] Knijnenburg, B. P., Willemsen, M. C., Gantner, Z., Soncu, H., & Newell, C. (2012). Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction*, 22(4-5), 441-504

[64] Chen, L., Pu, P., & Liu, Q. (2012). A study of information retrieval weighting schemes for collaborative filtering based recommendation. In *Proceedings of the Sixth ACM Conference on Recommender Systems* (pp. 249-252).

[65] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8), 30-37.

[66]KAMINSKAS, Marius et RICCI, Francesco. Contextual music information retrieval and recommendation: State of the art and challenges. *Computer Science Review*, 2012, vol. 6, no 2-3, p. 89-119.

[67] Clark, R. C., & Mayer, R. E. (2016). *E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning*. John Wiley & Sons.

- [68] Blake T Motion in instructional media: Some subject-display mode interactions. *Perceptual and Motor Skills* 1977;44:975-985
- [69] Cheng, J., Li, S., & Huang, J. (2021). User Preference Elicitation from User-Generated Content in Online Learning Platforms. *International Journal of Human-Computer Interaction*, 37(7), 673-689.
- [70] Liu, Y., Zhao, Z., & Liu, L. (2020). Preference Elicitation from Student Feedback for Personalized Course Recommendation in MOOCs. *Proceedings of the 15th International Conference on Computer Science & Education (ICCSE)*, 235-239.
- [71] Patel, N., Singh, R., & Patel, R. (2020). A Hybrid Approach for Preference Elicitation in E-learning Recommender Systems. *Journal of Intelligent & Fuzzy Systems*, 39(2), 2223-2235.
- [72] Wang, Y., Chen, Y., & Zhang, L. (2019). Effective Preference Elicitation Techniques for Content Recommendation in Online Learning Environments. *IEEE Transactions on Learning Technologies*, 12(4), 499-510.
- [73] DEVLIN, Jacob, CHANG, Ming-Wei, LEE, Kenton, *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [74] VASWANI, Ashish, SHAZEER, Noam, PARMAR, Niki, *et al.* Attention is all you need. *Advances in neural information processing systems*, 2017, vol. 30.
- [75] Meem, Raisa & Hasan, Khandaker. (2023). Improving Sentiment Analysis in Online Course Reviews with BERT and Transformer Attention Mechanism. 10.21203/rs.3.rs-3741963/v1.

B. Références Web (Techniques) :

(W1) : <https://www.python.org/>

(w2) : <https://pypi.org/>

(w3) <https://stackoverflow.com/>

(w5) <https://fr.khanacademy.org/about>

(W6) <https://arxiv.org/abs/1810.04805>

(w7] [http ://www.jalanmar.github.io](http://www.jalanmar.github.io)

Résumé :

Dans ce manuscrit, nous avons mené une analyse et une étude approfondie des domaines du e-learning, plus particulièrement les MOOCs, et des systèmes de recommandation en mettant l'accent sur la fusion entre eux afin d'améliorer la satisfaction des apprenants en se concentrant sur l'élicitation de leurs préférences dans le processus de la recommandation. Pour cette étude, nous avons utilisé le modèle BERT base uncased de la bibliothèque Transformers de Hugging Face, développé par Google et dédié pour le traitement des requêtes, des commentaires, la prédiction et la génération des textes en anglais.

L'objectif de ce travail est de proposer une solution au problème de la diversification des préférences des apprenants, en exploitant les avantages de BERT pour extraire les préférences à partir des commentaires fournis par les apprenants. Ensuite, ces préférences seront incorporées dans le processus de la recommandation afin d'améliorer la pertinence des recommandations de cours et d'augmenter la satisfaction des apprenants. Le système a été implémenté en utilisant le langage de programmation Python.

Une série de tests a été menée afin de démontrer les performances de notre système, suivie d'une comparaison avec les modèles BiLSTM et LSTM. Les résultats obtenus étaient satisfaisants en termes de classification des sentiments et des recommandations générées.

Mots clés : Élicitation des préférences, MOOCs, Systèmes de recommandation, E-learning, BERT base uncased, Pertinence, diversité.

Abstract:

In this manuscript, we conducted an in-depth analysis and study of the fields of e-learning, particularly MOOCs, and recommendation systems, focusing on the integration between them to enhance learner satisfaction by eliciting their preferences in the recommendation process. For this study, we utilized the BERT base uncased model from the Hugging Face Transformers library, developed by Google, specifically designed for processing queries, comments, prediction, and text generation in English.

The objective of this work is to propose a solution to the issue of diversifying learner preferences by leveraging BERT's capabilities to extract preferences from learner comments. These preferences are then incorporated into the recommendation process to improve the relevance of course recommendations and increase learner satisfaction. The system was implemented using the Python programming language.

A series of tests was conducted to demonstrate the performance of our system, followed by a comparison with the BiLSTM and LSTM models. The results obtained were satisfactory in terms of sentiment classification and the recommendations generated.

Keywords: Preference elicitation, MOOCs, Recommendation systems, E-learning, BERT base uncased, Relevance, Diversity.

ملخص

في هذه المخطوطة، أجرينا تحليلاً ودراسة متعمقة لمجالات التعلم الإلكتروني، وتحديدًا المساقات التعليمية الإلكترونية المفتوحة المصادر، وأنظمة التوصية مع التركيز على الدمج بينهما لتحسين رضا المعلمين من خلال التركيز على استنباط تفضيلاتهم في عملية التوصية. واستخدمنا في هذه الدراسة النموذج الأساسي غير المدمج من مكتبة المحولات الخاصة بـ "عناق الوجه" التي طورها جوجل والمخصصة لمعالجة الاستعلام والتعليق والتنبؤ وتوليد النصوص باللغة الإنجليزية.

والهدف من هذا العمل هو اقتراح حل لمشكلة تنوع تفضيلات المعلمين، من خلال استغلال مزايا BERT لاستخراج التفضيلات من التعليقات التي يقدمها المعلمون. سيتم بعد ذلك دمج هذه التفضيلات في عملية التوصيات من أجل تحسين ملاءمة توصيات الدورة التدريبية وزيادة رضا المعلمين. تم تنفيذ النظام باستخدام لغة برمجة بايثون.

وقد تم إجراء سلسلة من الاختبارات لإثبات أداء نظامنا، تليها مقارنة مع نموذجي BiLSTM و LSTM كانت النتائج التي تم الحصول عليها فرضية من حيث تصنيف المشاعر والتوصيات التي تم إنشاؤها.

الكلمات المفتاحية: استنباط التفضيلات، MOOCs، أنظمة التوصيات، التعلم الإلكتروني، قاعدة BERT غير المبينة على أساس غير الموصى بها، وجه المعانقة، أداء أنظمة التوصية.