

Les éléments à télécharger sont disponibles à l'adresse suivante :

<http://www.editions-eni.fr>

Saisissez la référence ENI de l'ouvrage **RI28PHP** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Chapitre 1

Préambule

1. Objectif de l'ouvrage 11
2. Bref historique de PHP 12
3. Où se procurer PHP ? 13
4. Conventions d'écriture 14
5. À propos des exercices 15

Chapitre 2

Introduction à PHP

1. Qu'est-ce que PHP ? 17
2. Structure de base d'une page PHP 19
 - 2.1 Les balises PHP 19
 - 2.2 La fonction echo 19
 - 2.3 Séparateur d'instructions 21
 - 2.4 Commentaires 21
 - 2.5 Mixer du PHP et du HTML 22
 - 2.6 Règles de nommage 25
 - 2.7 Exercice 1 : mon premier script PHP 26
3. Configuration de PHP 27
 - 3.1 Le fichier de configuration php.ini 27
 - 3.2 Informations sur la configuration 28
 - 3.3 Jeu de caractères 31
4. Utiliser PHP en ligne de commande 31

5. Les bases du langage PHP	32
5.1 Constantes.....	32
5.1.1 Définition.....	32
5.1.2 Portée	35
5.2 Variables	35
5.2.1 Initialisation et affectation	35
5.2.2 Portée et durée de vie.....	37
5.2.3 Variables dynamiques (ou variables variables).....	38
5.3 Types de données	38
5.3.1 Types de données disponibles.....	38
5.3.2 Types de données scalaires	39
5.3.3 Types de données spéciaux	47
5.3.4 Déclaration de type	48
5.4 Tableaux	50
5.4.1 Définition.....	50
5.4.2 Création	51
5.4.3 Manipulation	57
5.4.4 Décomposer un tableau	62
5.4.5 Portée	64
5.5 Opérateurs.....	64
5.5.1 L'opérateur d'affectation par valeur	64
5.5.2 L'opérateur d'affectation par référence	66
5.5.3 Les opérateurs arithmétiques	67
5.5.4 L'opérateur de chaîne	68
5.5.5 Les opérateurs combinés	68
5.5.6 Les opérateurs de comparaison	69
5.5.7 Les opérateurs logiques	70
5.5.8 L'opérateur ternaire	71
5.5.9 L'opérateur de fusion NULL	72
5.5.10 L'opérateur d'affectation de fusion NULL	73
5.5.11 L'opérateur de comparaison combinée.....	74
5.5.12 Précédence des opérateurs	75
5.6 Structures de contrôle.....	76
5.6.1 La structure if.....	76
5.6.2 La structure switch	78
5.6.3 La structure while	81
5.6.4 La structure do ... while.....	83

5.6.5	La structure for	84
5.6.6	Les instructions continue et break	87
5.6.7	L'expression match	88
5.7	Inclure un fichier.	90
5.7.1	Fonctionnement	90
5.7.2	Utilisation	93
5.8	Interrompre le script.	94
5.9	Exercice 2 : variables et structures de contrôle	95

Chapitre 3

Utiliser les fonctions PHP

1.	Préambule.	101
2.	Manipuler les constantes, les variables et les types de données	102
2.1	Constantes	102
2.2	Variables	103
2.3	Types de données	109
2.3.1	Conversions	109
2.3.2	Fonctions utiles	114
3.	Manipuler les tableaux	120
4.	Manipuler les nombres	134
5.	Manipuler les chaînes de caractères	140
6.	Utiliser les expressions rationnelles	159
6.1	Introduction	159
6.2	Structure d'une expression rationnelle	159
6.3	Fonctions.	169
7.	Manipuler les dates	174
8.	Générer un identifiant unique	194
9.	Manipuler les fichiers sur le serveur.	196
9.1	Fonctions utiles.	196
9.2	Exemples d'utilisation.	204
10.	Manipuler les en-têtes HTTP	205

11. Exercices	207
11.1 Exercice 3 : manipuler les données.	207
11.2 Exercice 4 : écrire et lire un fichier sur le serveur	211

Chapitre 4

Écrire des fonctions et des classes PHP

1. Fonctions	215
1.1 Introduction	215
1.2 Déclaration et appel	215
1.3 Paramètres	225
1.3.1 Syntaxe	225
1.3.2 Valeur par défaut	227
1.3.3 Déclaration du type de données	229
1.3.4 Passage par référence	233
1.3.5 Liste variable de paramètres	234
1.3.6 Utilisation du nom du paramètre dans l'appel.	237
1.4 Considérations sur les variables utilisées dans les fonctions	239
1.4.1 Variables locales/globales	239
1.4.2 Variables statiques	241
1.5 Les constantes et les fonctions	243
1.6 Récursivité	244
1.7 Fonction anonyme	245
1.8 Fonction fléchée	247
1.9 Fonction générateur	248
1.10 Exercice 5 : écrire des fonctions	251
2. Classes	253
2.1 Concept	253
2.2 Définir une classe	253
2.3 Instancier une classe	259
2.4 Héritage	263
2.5 Autres fonctionnalités sur les classes	268
2.5.1 Classes ou méthodes abstraites	268
2.5.2 Classes ou méthodes finales	269
2.5.3 Interfaces	270
2.5.4 Propriétés ou méthodes statiques - Constantes de classes ...	272

2.5.5 Traits	276
2.5.6 Classes anonymes	278
2.6 Exceptions	279
2.7 Énumérations	283
2.8 Exercice 6 : écrire une classe	289
3. Espaces de noms	293

Chapitre 5

Gérer les erreurs dans un script PHP

1. Vue d'ensemble	299
2. Les messages d'erreur PHP	300
3. Les fonctions de gestion des erreurs	304
4. Exercice 7 : gérer les erreurs	319

Chapitre 6

Gérer les formulaires et les liens

1. Vue d'ensemble	323
1.1 Introduction	323
1.2 Les liens	323
1.3 Les formulaires	326
1.3.1 Petit rappel sur les formulaires	326
1.3.2 Construire un formulaire dynamiquement	328
1.3.3 Traiter un formulaire à l'aide d'un script PHP	333
1.4 Récupérer les données d'une URL ou d'un formulaire	337
2. Récupérer les données passées par l'URL	340
2.1 Considérations	340
2.1.1 Que se passe-t-il si deux paramètres portent le même nom ?	340
2.1.2 Utiliser un tableau pour passer des données dans l'URL	340
2.2 Transmettre des caractères spéciaux	341
2.3 Exercice 8 : récupérer des données passées par l'URL	344

3.	Récupérer les données saisies dans le formulaire	347
3.1	Considérations	347
3.1.1	Que se passe-t-il si deux zones portent le même nom ?	347
3.1.2	Que se passe-t-il s'il y a deux formulaires dans la page HTML ?	347
3.1.3	Utiliser un tableau pour récupérer les données saisies.	347
3.1.4	Passer des informations dans une zone de formulaire cachée. . . .	349
3.2	Les différents types de zones	350
3.2.1	Vue d'ensemble	350
3.2.2	Zones contenant du texte	353
3.2.3	Groupes de boutons radio	353
3.2.4	Cases à cocher	354
3.2.5	Listes à sélection unique	356
3.2.6	Listes à sélection multiple	357
3.2.7	Boutons de validation	359
3.2.8	Boutons image	361
3.2.9	Boutons "reset" ou "button".	361
3.3	Synthèse.	362
3.4	Exercice 9 : récupérer des données saisies dans un formulaire	364
4.	Contrôler les données récupérées	367
4.1	Vue d'ensemble	367
4.2	Vérifications classiques.	367
4.2.1	Nettoyage des espaces indésirables	367
4.2.2	Données obligatoires	368
4.2.3	Longueur maximum d'une chaîne.	368
4.2.4	Caractères autorisés pour une chaîne - Format	368
4.2.5	Validité d'une date - Plage de valeurs	369
4.2.6	Validité d'un nombre - Plage de valeurs	371
4.2.7	Validité d'une adresse e-mail	371
5.	Problèmes sur les données récupérées.	373
6.	Utilisation des filtres	380
6.1	Principes.	380
6.2	Application aux formulaires.	389

6.3 Exercices	391
6.3.1 Exercice 10 : contrôler des données passées par l'URL	391
6.3.2 Exercice 11 : contrôler des données saisies dans un formulaire.	392
7. Aller sur une autre page	396
8. Échanger un fichier entre le client et le serveur.	402
8.1 Vue d'ensemble.	402
8.2 Envoyer un fichier depuis le client (upload)	402
8.3 Télécharger un fichier à partir du serveur (download).	407

Chapitre 7 Accéder aux bases de données

1. Introduction.	415
1.1 Vue d'ensemble	415
1.2 La notion de fetch.	416
2. Utilisation de MySQL.	417
2.1 Préambule	417
2.2 Connexion et déconnexion	418
2.2.1 Connexion	418
2.2.2 Déconnexion	419
2.2.3 Obtenir des informations sur le serveur MySQL.	420
2.2.4 Définir le jeu de caractères du client.	420
2.2.5 Obtenir des informations en cas d'erreur de connexion	420
2.2.6 Mode de rapport d'erreur	421
2.2.7 Exemple	422
2.3 Sélectionner une base de données	423
2.4 Utiliser des requêtes non préparées	424
2.4.1 Vue d'ensemble	424
2.4.2 Exécuter une requête	425
2.4.3 Connaître le nombre de lignes dans le résultat d'une requête de lecture	426
2.4.4 Extraire le résultat d'une requête de lecture	427
2.4.5 Obtenir des informations sur le résultat d'une requête de mise à jour.	437
2.4.6 Gérer les erreurs	439

2.5	Utiliser des requêtes préparées	441
2.5.1	Vue d'ensemble	441
2.5.2	Préparer une requête	443
2.5.3	Lier des variables PHP aux paramètres de la requête	444
2.5.4	Exécuter la requête préparée	446
2.5.5	Lier des variables PHP aux colonnes du résultat d'une requête de lecture	447
2.5.6	Extraire le résultat d'une requête de lecture	449
2.5.7	Utiliser un résultat stocké	451
2.5.8	Obtenir des informations sur le résultat d'une requête de mise à jour	453
2.5.9	Gérer les erreurs	456
2.5.10	Fermer une requête préparée	458
2.6	Gérer les transactions	458
2.7	Appeler un programme stocké	461
2.7.1	Procédure stockée	461
2.7.2	Fonction stockée	465
2.8	Exercice 12 : utiliser MySQL	467
3.	Utilisation d'Oracle	474
3.1	Préambule	474
3.2	Environnement NLS	475
3.3	Connexion et déconnexion.	475
3.3.1	Connexion	475
3.3.2	Déconnexion	477
3.3.3	Obtenir des informations sur le serveur Oracle	478
3.3.4	Obtenir des informations en cas d'erreur de connexion	478
3.3.5	Exemple	479
3.4	Exécuter une requête	480
3.4.1	Vue d'ensemble	480
3.4.2	Analyser une requête	481
3.4.3	Lier des variables PHP aux paramètres de la requête	482
3.4.4	Exécuter une requête	485
3.4.5	Extraire le résultat de la requête de lecture	487
3.4.6	Mettre à jour des données et gérer les transactions.	501
3.4.7	Fermer un curseur	506
3.5	Appeler une procédure stockée.	507
3.6	Illustration des problèmes liés à l'environnement NLS	512

3.7	Gérer les erreurs	515
3.8	Exercice 13 : utiliser Oracle	517
4.	PHP Data Objects (PDO)	524
5.	Gestion des apostrophes dans le texte des requêtes	527
6.	Exemples d'intégration dans des formulaires	532
6.1	Vue d'ensemble	532
6.2	Construction d'une liste de sélection dans un formulaire	543
6.3	Affichage d'une liste	545
6.4	Formulaire de saisie en liste	549
6.5	Formulaire de recherche et de saisie	553

Chapitre 8 Gérer les sessions

1.	Description du problème	557
2.	Authentification	558
2.1	Vue d'ensemble	558
2.2	Saisie de l'identification	558
2.2.1	Identification par formulaire	558
2.2.2	Identification par authentification HTTP	561
2.3	Vérifier l'identification saisie	563
3.	Utiliser des cookies	564
3.1	Principe	564
3.2	Application à la gestion des sessions	571
4.	Utiliser la gestion des sessions de PHP	572
4.1	Principes	572
4.2	Mise en œuvre	572
4.3	Gérer soi-même la transmission de l'identifiant de session	586
4.3.1	Description du problème	586
4.3.2	Solution	590
4.4	Quelques directives de configuration supplémentaires	592
4.5	Exemples d'application	593
4.5.1	Principes	593
4.5.2	Avec authentification des utilisateurs	596
4.6	Remarques et conclusion	599

4.7 Exercice 14 : gérer les sessions	603
5. Conserver des informations d'une visite à une autre.	607
6. Petite synthèse sur les variables Get/Post/Cookie/Session.	612

Chapitre 9

Envoyer un courrier électronique

1. Vue d'ensemble	617
2. Envoyer un message texte sans pièce jointe.	617
3. Envoyer un message au format MIME	621
3.1 Préambule	621
3.2 Message au format HTML	621
3.3 Message avec pièce jointe	624
4. Exercice 15 : envoyer un courrier électronique	627

Annexe

1. Variables PHP prédéfinies	631
2. Constantes PHP prédéfinies	633
3. Exemples complémentaires	634
3.1 Introduction	634
3.2 Lire un document XML	635
3.3 Générer un document PDF	640
3.4 Générer une image	645
4. Résumé des principales nouveautés des versions 8.0, 8.1 et 8.2	650

Index	655
-----------------	-----

Les éléments à télécharger sont disponibles à l'adresse suivante :

<http://www.editions-eni.fr>

Saisissez la référence de l'ouvrage **EIIAPHP** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

1. Public concerné et prérequis. 11
2. Objectifs 12
3. Méthodologie 12
4. Remerciements 13

Chapitre 1

Introduction à l'intelligence artificielle

1. Introduction 15
2. Définition de l'intelligence artificielle 16
 - 2.1 Systèmes basés sur des règles. 16
 - 2.2 Le Machine Learning - apprentissage automatique 16
 - 2.3 Le Deep Learning - apprentissage profond 17
3. Évolution de l'intelligence artificielle. 18
4. Applications concrètes de l'IA 19

Chapitre 2

PHP et l'intelligence artificielle

1. Introduction 21
2. Place de PHP dans l'IA 22
 - 2.1 Facilité d'utilisation 22
 - 2.2 Librairies et frameworks. 22
 - 2.3 Communauté 23

2 ————— PHP et intelligence artificielle

Concepts, outils et applications

2.4	Performances	23
2.5	Écosystème de développement	24
3.	Librairies PHP pour l'IA	24
3.1	Librairies de création de modèles	24
3.1.1	PHP-ML	25
3.1.2	Rubix ML	25
3.2	Librairies de consommation de modèles	25
4.	Configurer l'environnement de développement pour PHP	26
4.1	Serveur web et PHP 7.4	26
4.1.1	Windows	26
4.1.2	Linux	26
4.1.3	Mac OS	26
4.1.4	Extensions PHP	27
4.2	Librairies PHP	27

Chapitre 3

Comprendre et préparer les données

1.	Introduction	31
2.	Types de données	32
2.1	Données numériques	32
2.2	Données catégorielles	32
2.3	Données ordinales	33
2.4	Données textuelles	33
2.5	Données temporelles	33
2.6	Données géospatiales	33
2.7	Données multimédias	34
2.8	Données logiques/binaires	34
3.	Types d'apprentissage	35
4.	Solutions de visualisation des données	36
4.1	Diagramme à bâtons	37
4.2	Diagramme en camembert	38

4.3	Diagramme nuage de points	40
4.4	Boîte à moustaches	42
4.5	Carte de chaleur	45
4.6	Diagramme de carte géographique	47
5.	Récolte, nettoyage et imputation des données	50
5.1	Récolte de données au format CSV	50
5.2	Récolte de données au format NDJSON	51
5.3	Récolte de données depuis une base de données SQL	51
5.4	Extraction/Filtrage de colonnes	51
5.4.1	Extraction de colonnes	52
5.4.2	Filtrage de colonnes	53
5.5	Les ensembles de données dans Rubix ML	54
5.5.1	Création d'un ensemble Labeled	54
5.5.2	Création d'un ensemble Unlabeled	55
5.5.3	Méthodes de description des ensembles de données	56
5.5.4	Méthodes de sélection et de manipulation des ensembles de données	58
5.6	Nettoyage et imputation des données	61
5.6.1	Suppression des doublons	61
5.6.2	Traitement en masse des données avec les transformers	61
5.6.3	Gestion des données manquantes, imputation des données	63
6.	Exploration et analyse des données	70
6.1	Analyse univariée des caractéristiques numériques continues	70
6.2	Analyse univariée des caractéristiques catégorielles	75
6.3	Analyse multivariée	77
6.4	Matrice des corrélations	81
7.	Prétraitement des données	82
7.1	Transformation des données catégorielles en données numériques	82
7.2	Transformation des données ordinales en données numériques	83
7.3	Transformer les données binaires en données numériques	84

4 _____ PHP et intelligence artificielle

Concepts, outils et applications

7.4	Transformer les données de texte en données numériques	84
7.5	Transformer les données multimédias en données numériques exploitables	86
7.6	Standardisation	87
7.7	Normalisation	87
8.	Réduction des dimensions	88
8.1	L'analyse en composantes principales	89
8.2	T-SNE - t-Distributed Stochastic Neighbor Embedding	93
9.	En résumé	94

Chapitre 4

Critères de performance et de sélection

1.	Introduction	95
2.	Défi de la création du modèle idéal	96
2.1	Capacité de généralisation	97
2.2	Sur-apprentissage	98
3.	Évaluation des performances	99
3.1	Découpage en ensembles d'entraînement et de test	99
3.2	La validation croisée en K parties (K-folds)	101
4.	Métriques d'évaluation des performances des modèles	101
4.1	Métriques d'évaluation des modèles de classification	102
4.1.1	Matrice de confusion	102
4.1.2	Précision globale	105
4.1.3	Précision par classe de prédiction	106
4.1.4	Rappel d'une classe	107
4.1.5	Score F1	107
4.1.6	Les métriques des modèles de classification avec Rubix ML	108

4.2	Métriques d'évaluation des modèles de régression	109
4.2.1	Erreur absolue moyenne (Mean Absolute Error)	111
4.2.2	Moyenne du carré des erreurs (Mean Squared Error)	112
4.2.3	La racine de la moyenne des erreurs au carré (Root Mean Squared Error)	113
4.2.4	L'erreur carrée relative (RSE) et le coefficient de détermination R^2	114
4.2.5	Les métriques des modèles de régression avec Rubix ML	115
5.	Exploitation des métriques d'évaluation dans l'optimisation des hyperparamètres.	116

Chapitre 5

Algorithmes d'apprentissage supervisé

1.	Introduction	119
2.	Algorithmes de régression	120
2.1	Régression linéaire et multilinéaire	120
2.1.1	Régression linéaire	121
2.1.2	Régression multilinéaire	121
2.1.3	Mise en pratique en PHP avec l'algorithme Ridge	122
2.2	Arbre de régression	131
2.2.1	Principes de fonctionnement d'un arbre de régression	131
2.2.2	Points forts/faibles des arbres de régression	132
2.2.3	Mise en pratique en PHP	133
2.2.4	Le gradient boosting	143
2.3	Algorithme de régression par les K plus proches voisins	146
2.3.1	Principes de fonctionnement de la régression par les K plus proches voisins (KNN)	146
2.3.2	Points forts/faibles de KNN	147
2.3.3	Notion de distances	148
2.3.4	Mise en pratique en PHP	151

6 ————— PHP et intelligence artificielle

Concepts, outils et applications

2.4	Algorithme de régression par les machines à vecteur de support (SVR)	158
2.4.1	Principes de fonctionnement	158
2.4.2	Points forts/faibles	159
2.4.3	Mise en pratique en PHP	160
2.5	Réseaux de neurones appliqués à la régression	166
2.5.1	Principes de fonctionnement	167
2.5.2	Fonctions d'activation	173
2.5.3	Points forts/faibles des réseaux de neurones	174
2.5.4	Mise en pratique en PHP	175
3.	Algorithmes de classification	183
3.1	La régression logistique (classification binaire)	184
3.1.1	Principe de fonctionnement	184
3.1.2	Points forts/faibles de la régression logistique	184
3.1.3	Application en PHP	185
3.1.4	One versus all	191
3.2	Arbre de classification	196
3.2.1	Principe de fonctionnement	196
3.2.2	Points forts/faibles des arbres de classification	196
3.2.3	Mise en pratique en PHP	198
3.3	Forêt aléatoire	203
3.3.1	Principes de fonctionnement	203
3.3.2	Avantages/inconvénients par rapport à un arbre simple	203
3.3.3	Mise en pratique en PHP	204
3.4	AdaBoost	206
3.4.1	Principes de fonctionnement	206
3.4.2	Points forts/faibles d'AdaBoost	209
3.4.3	Illustration des modèles à base d'arbres de classification	209
3.4.4	Mise en pratique en PHP	210
3.5	SVC (classificateurs à vecteur de support)	212
3.5.1	Principes de fonctionnement	212
3.5.2	Points forts/faibles du SVC	213
3.5.3	Mise en pratique en PHP	214

3.6	Classificateur par les K plus proches voisins (KNN classifier)	221
3.6.1	Principe de fonctionnement	221
3.6.2	Points forts/faibles des algorithmes de classification KNN	224
3.6.3	Mise en pratique en PHP	225
3.7	Réseaux de neurones de classification	230
3.7.1	Principe de fonctionnement des réseaux de neurones de classification	230
3.7.2	Mise en pratique des réseaux de neurones de classification	231
3.8	Naive Bayes	239
3.8.1	Points forts/faibles des modèles à base de Naive Bayes	242
3.8.2	Mise en pratique avec PHP	243
4.	En résumé	246

Chapitre 6

Algorithmes d'apprentissage non supervisé

1.	Introduction	247
2.	Algorithmes de partitionnement des données/clustering	248
2.1	Algorithme des K-Moyennes (ou K-Means)	248
2.1.1	Principe de fonctionnement de l'algorithme des K-Moyennes	248
2.1.2	Points forts/faibles du clustering avec les K-Moyennes	250
2.1.3	Mise en pratique en PHP	250
2.2	Algorithme DBSCAN	259
2.2.1	Principe de fonctionnement de DBSCAN	259
2.2.2	Points forts/faibles de DBSCAN	260
2.2.3	Mise en pratique en PHP	260

8 ————— PHP et intelligence artificielle

Concepts, outils et applications

3. Algorithmes de détection d'anomalies	265
3.1 L'algorithme One-Class SVM	265
3.1.1 Principe de fonctionnement.	265
3.1.2 Points forts/faibles de One-Class SVM.	266
3.1.3 Mise en pratique en PHP	266
3.2 Les forêts d'isolation.	271
3.2.1 Principe de fonctionnement des forêts d'isolation	271
3.2.2 Points forts/faibles des forêts d'isolation	272
3.2.3 Mise en pratique en PHP	272
3.3 Algorithme du Z-Score robuste	277
3.3.1 Principe de fonctionnement.	277
3.3.2 Points forts/faibles du Z-Score robuste.	278
3.3.3 Mise en pratique en PHP	279
4. En résumé	284

Chapitre 7

Persister et réutiliser les modèles

1. Introduction	285
2. Les objets persistants et les sérialiseurs de Rubix ML	286
3. Découverte des encodages et des persisteurs dans Rubix ML	288
4. Mise en pratique de la sauvegarde d'un modèle	289
5. Mise en pratique du chargement d'un modèle	293
6. Solutions de déploiement des modèles en production	294

Chapitre 8

Choix, optimisation et sélection des modèles

1. Introduction	301
2. Présélection des modèles les plus adaptés	302
2.1 Présélection des modèles de régression	304
2.2 Présélection des modèles de classification	306
2.3 Stratégies de choix et d'optimisation des hyperparamètres	308
2.3.1 Favoriser l'optimisation des hyperparamètres les plus critiques	309
2.3.2 Ajustements par variations de granularité	309
2.3.3 Utiliser des valeurs aléatoires	310
3. Sélection des modèles	311
3.1 Choix du modèle le plus performant	311
3.2 Choix d'une combinaison de modèles	311
3.3 Refus des modèles	312
4. En résumé	313

Chapitre 9

Utiliser des modèles tiers via des API

1. Introduction	315
2. Création des accès à l'API OpenAI	316
3. Consommation d'un LLM pour répondre à une question textuelle	319
3.1 Génération d'une réponse à une question textuelle sans contexte spécifié	319
3.2 Génération d'une réponse à une question textuelle avec contexte	324
4. Utilisation d'un modèle de vision par ordinateur couplé à un LLM pour récupérer des informations sur des images	327

10 _____ PHP et intelligence artificielle

Concepts, outils et applications

5.	Utilisation d'un modèle de transcription automatique d'un fichier audio en texte (Speech-To-Text) couplé à un LLM . . .	332
5.1	Transcription automatique de l'audio vers du texte	333
5.2	Extraction des données métier à l'aide d'un LLM	335
6.	Création d'un fichier audio à partir d'un texte à l'aide d'un modèle de génération de voix	337
7.	Utilisation d'un embedding pour vectoriser du texte.	339
7.1	Utilisation de l'embedding pour transformer du texte en vecteurs	340
7.2	Tests sur les similarités sémantiques entre vecteurs	342
	Index	345