



Guide d'écriture d'un rapport scientifique

Erwann Lesech, Charly Saugey, Hédelin Ropital

Juin 2024



Abstract

Ce rapport présente une étude sur la reconnaissance de vin en utilisant le dataset disponible sur HuggingFace [1]. Nous abordons le contexte scientifique, la problématique associée, l'état de l'art, notre contribution, ainsi que les principaux résultats obtenus. La méthode utilisée est décrite en détail, suivie des expérimentations et des résultats. Enfin, nous concluons avec les perspectives futures de notre recherche.

1 Introduction

Dans cette section, nous présenterons le contexte de notre étude, la problématique abordée, notre contribution et le plan du rapport.

1.1 Contexte

La recherche sur la reconnaissance de vin est cruciale en raison de la nécessité d'authentifier et de classer correctement les vins pour des raisons commerciales et de qualité. Les défis actuels incluent la variabilité des caractéristiques des vins et les techniques de reconnaissance peu précises.

1.2 Problématique

La principale problématique est de développer un modèle de reconnaissance de vin précis et efficace. Les solutions existantes, telles que les méthodes basées sur l'analyse chimique, présentent des avantages et des inconvénients, notamment en termes de coût et de complexité.

1.3 Contribution

Notre contribution consiste à utiliser des techniques d'apprentissage automatique pour développer un modèle de reconnaissance de vin. Cette méthode est importante car elle pourrait offrir une solution plus accessible et moins coûteuse.

1.4 Plan du rapport

Ce rapport est structuré comme suit :

- **Section 2 : État de l'art** - Un examen des méthodes existantes pour la reconnaissance de vin.
- **Section 3 : Méthode** - Une description détaillée de la méthode proposée.
- **Section 4 : Expérimentation et résultats** - Les expérimentations menées et les résultats obtenus.
- **Section 5 : Conclusions et perspectives** - Les conclusions tirées de cette étude et les perspectives pour des travaux futurs.

2 État de l'art

Dans ce chapitre, nous présentons un historique des méthodes existantes pour répondre à la problématique de la reconnaissance de vin.

2.1 Analyse chimique des composants du vin

2.1.1 Description

L'analyse chimique des composants du vin implique l'utilisation de diverses techniques de chimie analytique pour identifier et quantifier les composants chimiques présents dans le vin. Ces techniques peuvent inclure la chromatographie, la spectrométrie de masse, et d'autres méthodes analytiques.

2.1.2 Analyse

Les avantages de l'analyse chimique [2] incluent une grande précision et la capacité de détecter des composants spécifiques à des concentrations très faibles. Cependant, cette méthode peut être coûteuse et nécessite un équipement spécialisé et du personnel qualifié pour interpréter les résultats.

- **Avantages :**
 - Haute précision.
 - Détection de composants spécifiques.
- **Inconvénients :**
 - Coût élevé.
 - Nécessité d'équipements spécialisés.
 - Requiert du personnel qualifié.

2.2 Utilisation de techniques de vision par ordinateur pour la reconnaissance de l'étiquette du vin

2.2.1 Description

La reconnaissance de l'étiquette du vin par des techniques de vision par ordinateur implique l'utilisation d'algorithmes de traitement d'images et de reconnaissance de formes pour identifier les étiquettes de vin à partir d'images numériques. Ces techniques peuvent inclure la reconnaissance de texte (OCR) [3], la détection de caractéristiques et l'apprentissage automatique.

2.2.2 Analyse

Les avantages de la vision par ordinateur incluent une automatisation rapide et l'aptitude à traiter de grands volumes de données. Cependant, cette méthode peut être sensible aux variations d'éclairage, à la qualité de l'image et aux angles de prise de vue.

- **Avantages :**

- Automatisation rapide.
- Traitement de grands volumes de données.

- **Inconvénients :**

- Sensible aux variations d'éclairage.
- Dépend de la qualité de l'image.
- Influencé par les angles de prise de vue.
- Nécessite l'étiquette du vin.

2.3 Bilan et proposition de solution combinée

Les deux méthodes présentées ont leurs propres avantages et inconvénients. L'analyse chimique des composants du vin offre une grande précision et la capacité de détecter des composants spécifiques, mais elle est coûteuse et nécessite des équipements spécialisés ainsi que du personnel qualifié. D'autre part, la reconnaissance par vision par ordinateur est rapide et peut traiter de grands volumes de données, mais elle est sensible aux conditions d'éclairage, à la qualité des images, et nécessite une étiquette visible.

Une solution potentielle pour améliorer la reconnaissance de vin pourrait être de combiner ces deux approches. En utilisant le machine learning, nous pourrions intégrer les résultats de l'analyse chimique avec les techniques de vision par ordinateur pour créer un modèle plus robuste et précis. Cette combinaison permettrait d'automatiser le processus de reconnaissance en tirant parti de la précision de l'analyse chimique et de la rapidité de la vision par ordinateur. Par exemple, les caractéristiques chimiques détectées pourraient servir de données d'entrée pour un modèle d'apprentissage automatique, qui serait capable de faire des prédictions même en l'absence d'étiquettes visibles, en se basant sur les profils chimiques des vins.

Cette approche hybride pourrait offrir une solution plus complète et efficace pour la reconnaissance de vin, combinant les points forts de chaque méthode tout en atténuant leurs limitations respectives.

3 Méthode

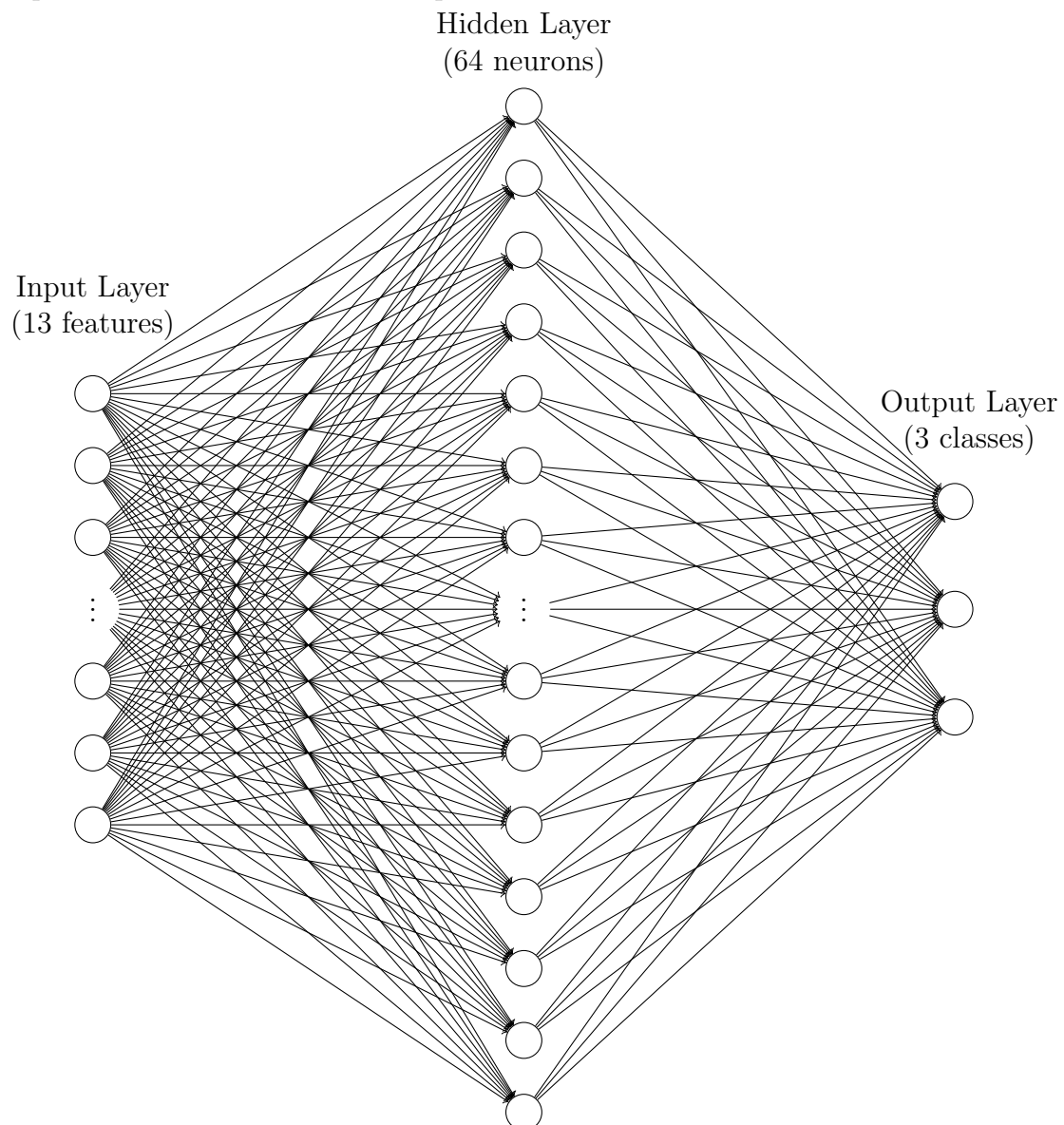
Ce chapitre décrit la méthode utilisée pour résoudre notre problématique.

3.1 Formalisation du problème

Le problème est formalisé mathématiquement par la classification des vins en fonction de leurs caractéristiques. Les variables d'entrée sont les caractéristiques chimiques comme par exemple le taux d'alcool et visuelles des vins comme par exemple l'intensité de la couleur. Les variables de sortie sont les classes de vins.

3.2 Architecture

L'architecture de notre solution est un réseau de neurones de type MLP [4], avec une couche d'entrée de 13 neurones (car le modèle comporte 13 features), une couche cachée de 64 neurones (ce choix est expliqué plus bas) et enfin une couche de sortie de 3 neurones correspondant aux 3 classes de vins que l'on doit classifier.



Le choix d'une couche cachée de 64 neurones n'est pas un hasard. Le dataset HuggingFace ne contient pas beaucoup de données, ce qui est un frein pour un apprentissage efficace.

Effectuer un grand nombre d'époch sur un petit jeu de données est une situation propice au surapprentissage.

Pour que l'apprentissage soit efficace avec un faible nombre d'époch, il faut donc un grand nombre de neurone et de poids. C'est la raison pour laquelle nous avons décidé d'utiliser cette architecture.

3.3 Fonction de coût

La fonction de coût utilisée est la cross-entropie pour la classification des vins. La formule mathématique est :

$$\mathcal{L} = - \sum_i y_i \log(\hat{y}_i) \quad (1)$$

C'est celle-ci que nous avons retenue pour plusieurs raisons.

Tout d'abord, elle fait partie des fonctions de coût les plus populaires et utilisées, donc c'est une des premières à laquelle nous avons pensé.

De plus, après plusieurs recherches sur internet [5], nous avons remarqué qu'elle était très bien adaptée pour le problème de classification multi-label ce qui est notre cas ici. Nous n'attendons pas une sortie binaire mais bel et bien un choix entre 3 classes.

Enfin, après plusieurs tests, c'est elle qui nous permettait d'augmenter notre précision le plus rapidement possible.

4 Expérimentation et résultats

Dans ce chapitre, nous détaillons les expérimentations réalisées et les résultats obtenus.

4.1 Présentation des données

Les données utilisées sont disponibles sur le site de HuggingFace [1]. La répartition des données est de 70% pour l'entraînement, 20% pour la validation, et 10% pour le test.

Le but de l'ensemble d'entraînement est d'ajuster les poids et les paramètres du modèle. Le modèle "apprend" à partir de ces données en essayant de minimiser une fonction de coût ou une perte.

Le but de l'ensemble de validation est d'évaluer la performance du modèle pendant la phase de développement. Il aide à ajuster les hyperparamètres (comme le taux d'apprentissage, la taille du réseau, etc.) et à prévenir le surapprentissage.

L'ensemble de test sert quant à lui à évaluer la performance finale du modèle après l'entraînement complet et l'ajustement des hyper-paramètres. Il n'est jamais utilisé pendant l'entraînement ou la validation.

Cette répartition équilibrée permet de maximiser l'utilisation des données disponibles pour l'apprentissage tout en assurant une évaluation rigoureuse et impartiale de la performance du modèle. En ayant des ensembles de validation et de test distincts, nous pouvons mieux détecter et prévenir le surapprentissage, sélectionner le meilleur modèle, et obtenir une estimation fiable de sa capacité à généraliser sur des données non vues.

4.2 Optimisation

Différentes méthodes d'optimisation ont été testées, incluant Adam, SGD (Stochastic Gradient Descent). Après quelques recherches [6], nous avons vu à plusieurs reprises que l'optimiseur Adam était plus performant que SGD. C'est pourquoi nous avons décidé de l'utiliser dans ce projet.

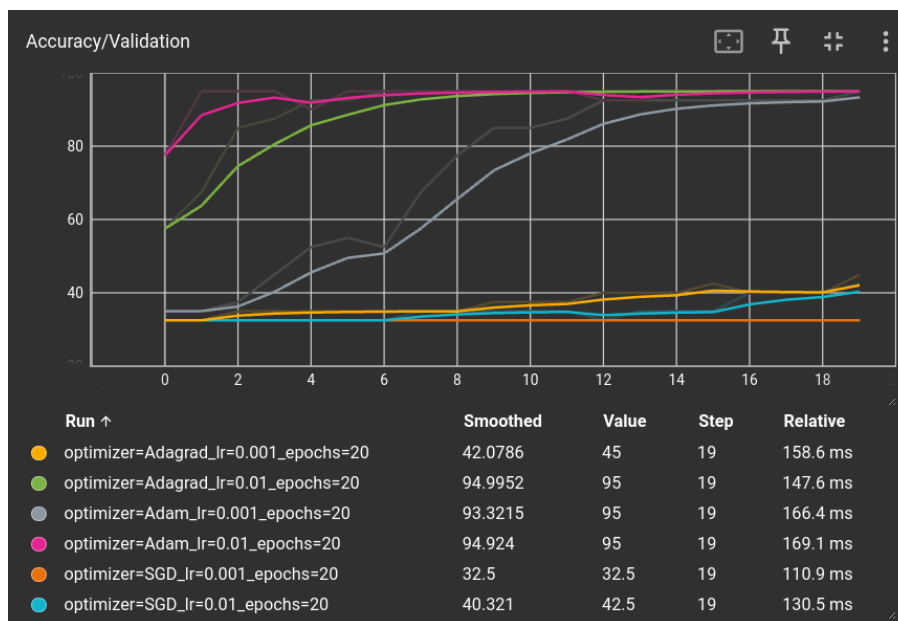


Figure 1: Accuracy dans différentes configurations

Voici ci-dessus un Tensorboard qui nous a permis de comparer les différentes méthodes d'optimisations afin de ne garder que la configuration la plus optimale pour notre réseau.

4.3 Réglage des hyperparamètres

Les performances de notre réseau ont été analysées en fonction des hyperparamètres tels que le taux d'apprentissage, le nombre de couches, le nombre de neurones par couche ainsi que le nombre d'itérations. Comme il a été dit plus haut, l'architecture du réseau de neurone n'est pas dûe au hasard mais est une conséquence de la taille du dataset. Celle-ci influe aussi les hyper-paramètres que nous avons utilisé.

4.3.1 Learning-Rate

Le learning-rate est le paramètre qui permet au réseau d'apprendre par pas plus ou moins important. Etant donnée la taille du dataset utilisé, nous ne pouvons nous permettre de faire un nombre d'epoch trop élevé pour éviter un sur-apprentissage, il a donc fallut que le learning-rate choisi ne soit pas trop faible afin que le réseau puisse apprendre avec peu d'epoch.

Comme on peut l'observer sur la figure ci-dessous qui présente la loss au fil des itérations en fonctions des différents learning rate et optimizer, nous avons conclu après plusieurs test qu'un learning-rate de 0.01 nous permettait d'obtenir les meilleurs résultats.

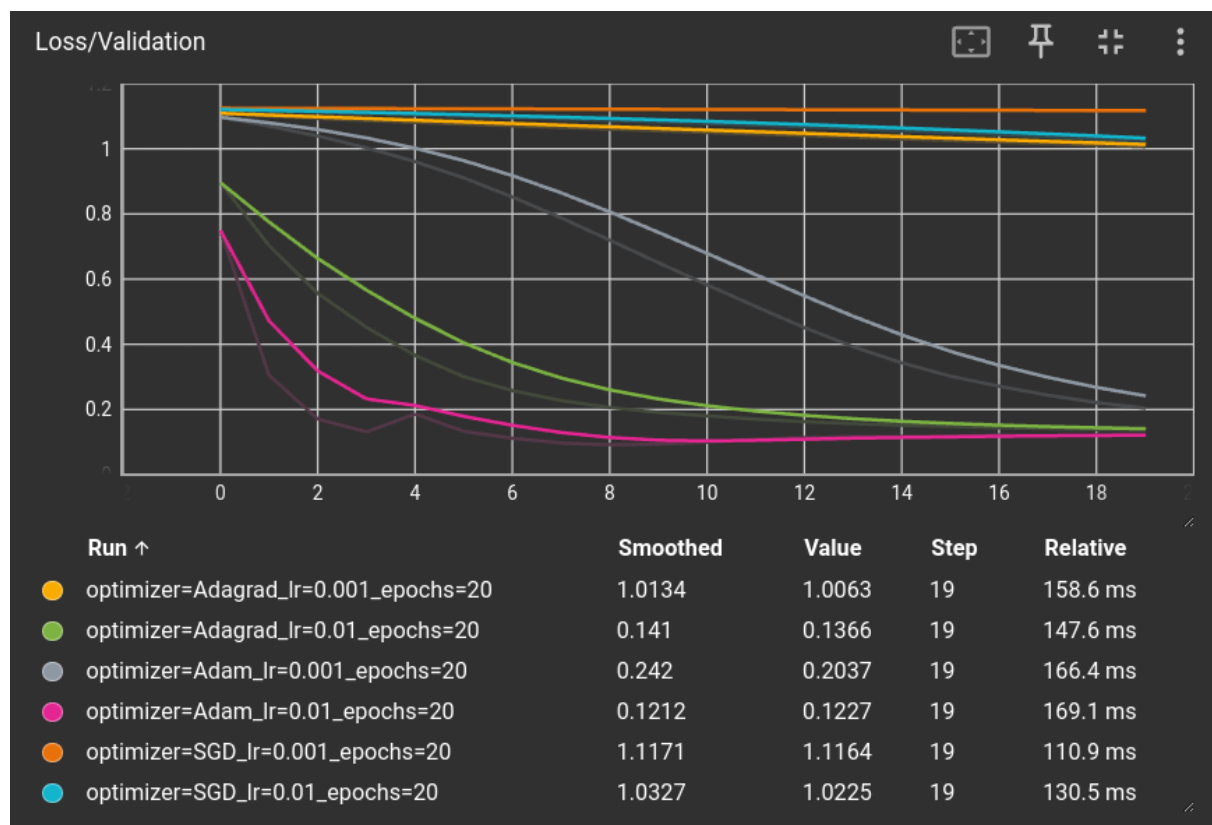


Figure 2: Loss dans différentes configurations

4.3.2 Le Batch-Size

Le batch-size, défini ici à 20, optimise l'équilibre entre la précision de l'apprentissage et la vitesse d'exécution en permettant au réseau de neurones de traiter les données en petits groupes, favorisant ainsi une convergence efficace et stable du modèle.

4.4 Résultats

Les résultats obtenus montrent une précision qui oscille entre 90% et 97.5% sur les données de test. Les figures ci-dessous illustrent ces résultats.

Epoch	training loss	validation loss	accuracy	best train loss	best validation loss	best accuracy
1	1.10608129	1.07504481	40.0	1.10608129	1.07504481	40.0
2	1.06486326	1.03539115	60.0	1.06486326	1.03539115	60.0
3	1.02803413	0.99819127	70.0	1.02803413	0.99819127	70.0
4	0.9903598	0.95917901	85.0	0.9903598	0.95917901	85.0
5	0.94946324	0.91748363	87.5	0.94946324	0.91748363	87.5
6	0.90464056	0.87031889	92.5	0.90464056	0.87031889	92.5
7	0.85475191	0.81613934	95.0	0.85475191	0.81613934	95.0
8	0.79615076	0.75585771	95.0	0.79615076	0.75585771	95.0
9	0.73117502	0.68627748	95.0	0.73117502	0.68627748	95.0
10	0.65988987	0.61284822	95.0	0.65988987	0.61284822	95.0
11	0.58509347	0.53715014	97.5	0.58509347	0.53715014	97.5
12	0.50747374	0.46205184	97.5	0.50747374	0.46205184	97.5
13	0.4333152	0.3908671	95.0	0.4333152	0.3908671	97.5
14	0.36354594	0.32712565	95.0	0.36354594	0.32712565	97.5
15	0.3028362	0.27305694	95.0	0.3028362	0.27305694	97.5
16	0.25021429	0.22819545	95.0	0.25021429	0.22819545	97.5
17	0.20904391	0.19257662	95.0	0.20904391	0.19257662	97.5
18	0.17472603	0.16418613	95.0	0.17472603	0.16418613	97.5
19	0.1484331	0.14197569	95.0	0.1484331	0.14197569	97.5
20	0.1268372	0.12575147	95.0	0.1268372	0.12575147	97.5

Figure 3: Figure des résultats

5 Conclusions/Perspectives

La conclusion résume les principales contributions et résultats de notre étude. Nous avons montré que notre modèle de reconnaissance de vin est précis et efficace. Les perspectives futures incluent l'amélioration du modèle en intégrant davantage de caractéristiques et l'application de notre méthode à d'autres types de produits alimentaires.

References

- [1] Katosky. Wine recognition dataset. *HuggingFace*, 2022.
- [2] Mon Vin Nature. Analyse chimique des vins: décryptage des secrets dans le verre. <https://monvinnature.com/fr/blogs/13-analyse-chimique-des-vins-decryptage-des-secrets-dans-le-verre>, 2023.
- [3] Neovision. Reconnaissance des bouteilles de vin. <https://neovision.fr/portfolio-item/reconnaissance-bouteilles-vin/>, 2023.
- [4] Wikipedia. Multilayer perceptron. https://en.wikipedia.org/wiki/Multilayer_perceptron, 2023.
- [5] Understanding loss functions for classification. https://medium.com/@nghihuynh_37300/understanding-loss-functions-for-classification-81c19ee72c2a#74cd, 2020.
- [6] Adam vs. sgd: An illustrative comparison. <https://ichi.pro/fr/un-guide-2021-pour-ameliorer-les-cnn-optimizers-adam-vs-sgd-80643525206024>, 2021.