

# Classification des produits

François COMBES<sup>1</sup>, and Thomas LEYSENS<sup>2</sup>

<sup>1</sup>Université Gustave Eiffel, <sup>2</sup>Université Gustave Eiffel

Il s'agit de classer les produits des enquêtes chargeurs en [catégories NST 2007](#)

= La Communauté de la donnée corrompue



Figure 1: *Affronter l'hydre "Corrupted Data" (image générée par IA)*

Les dénominations posent des problèmes car elles contiennent des erreurs d'accents, de la ponctuation, des majuscules, des minuscules, des fautes de frappe, des intitulés incompréhensibles, ce qui rend d'autant plus compliqué leur classification.

Voici un schéma simple des étapes & tests que nous avons réalisés et que nous allons vous exposer par la suite en détail.

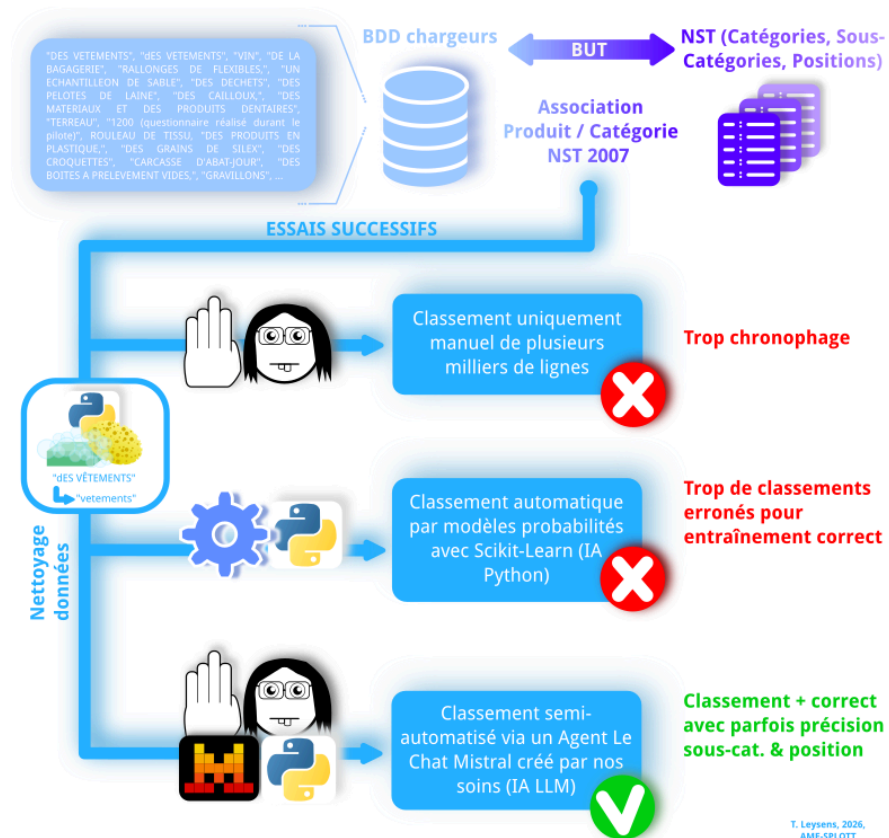


Figure 2: Une synthèse du cheminement vers la solution

# 1. CLASSEMENT MANUEL: LE REFUS D'UNE OPTION CHRONOPHAGE

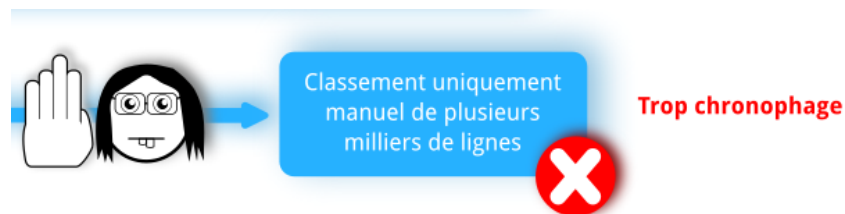


Figure 3: Classement manuel

# 2. CLASSEMENT AUTOMATISÉ AVEC DU CODE SPÉCIALISÉ BASÉ SUR SCIKIT-LEARN

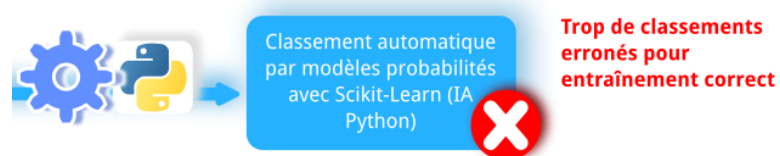


Figure 4: Classement automatisé via Python

| Critère                | TextBlob (Naive Bayes) | scikit-learn (Modèles Linéaires) |
|------------------------|------------------------|----------------------------------|
| Type de modèle         | Naive Bayes            | SVM, Régression Logistique, etc. |
| Features               | Bag-of-words basique   | TF-IDF, n-grammes, embeddings    |
| Précision              | Moyenne                | Élevée                           |
| Flexibilité            | Très limitée           | Très flexible                    |
| Optimisation           | Aucune                 | GridSearchCV, cross-validation   |
| Scalabilité            | Faible                 | Bonne                            |
| Facilité d'utilisation | Très simple            | Plus complexe                    |

2.a. Tests & évaluation des méthodes de classement automatisé

2.b. Comparaison de la précision de chaque modèle

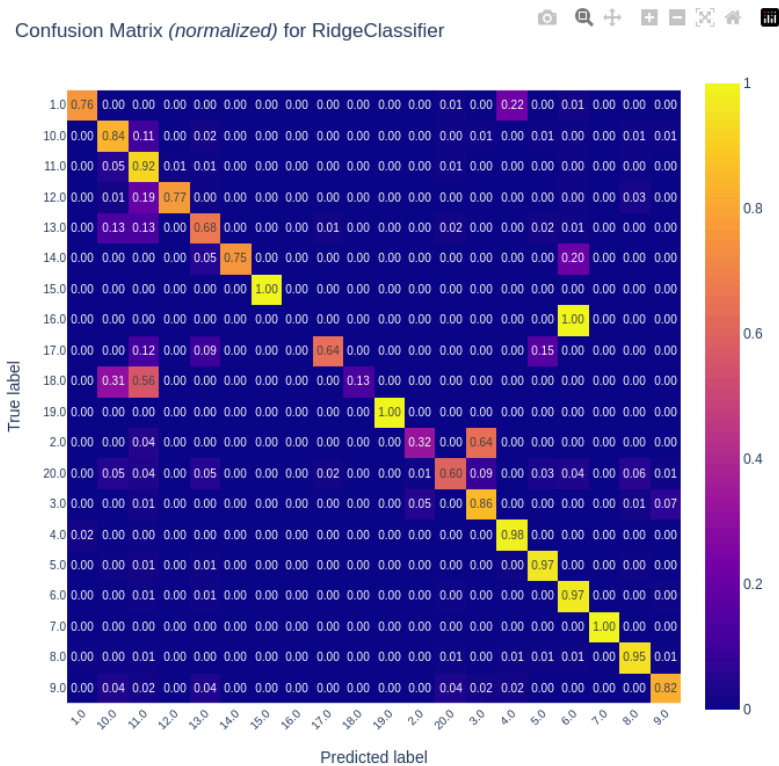
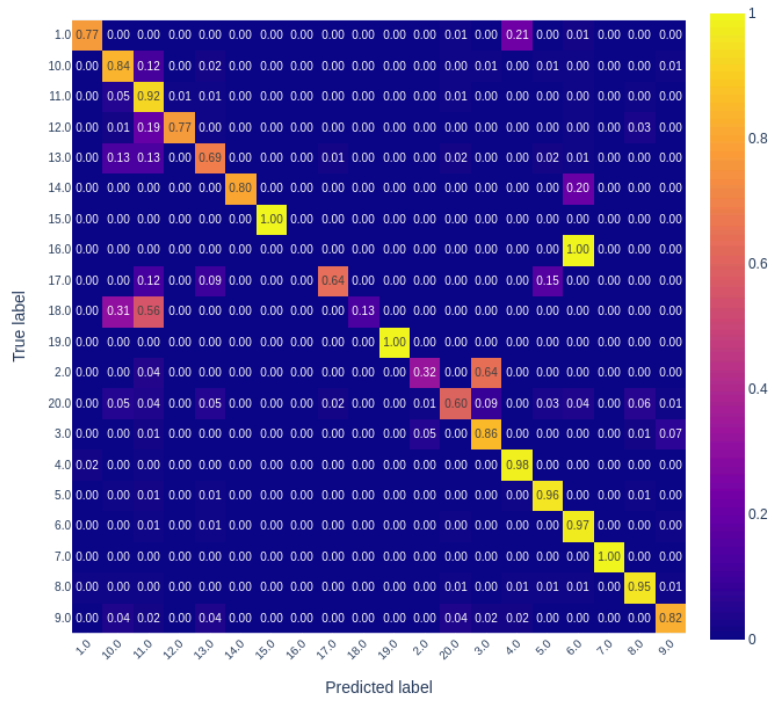
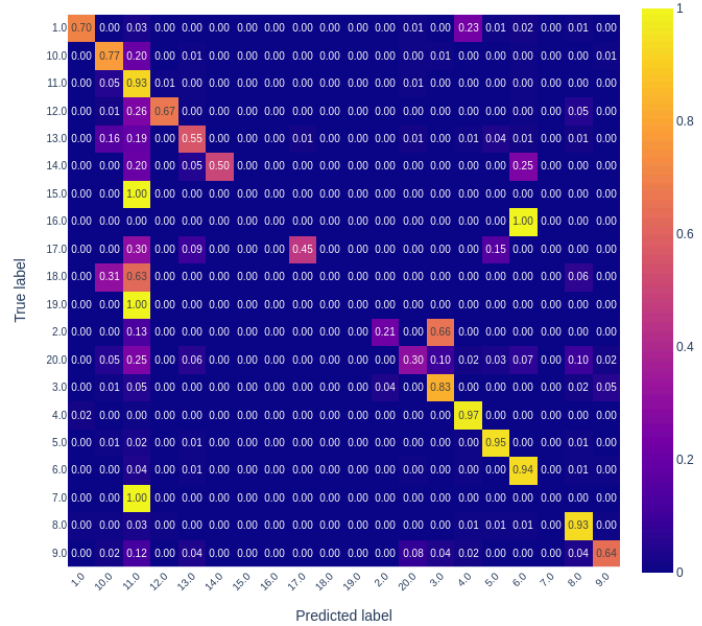


Figure 5: *Modèle “Ridge”*

Confusion Matrix (normalized) for LinearSVC

Figure 6: *Modèle “SVC”*

Confusion Matrix (normalized) for LogisticRegression

Figure 7: *Modèle “Logistic”*

Confusion Matrix (normalized) for ComplementNB

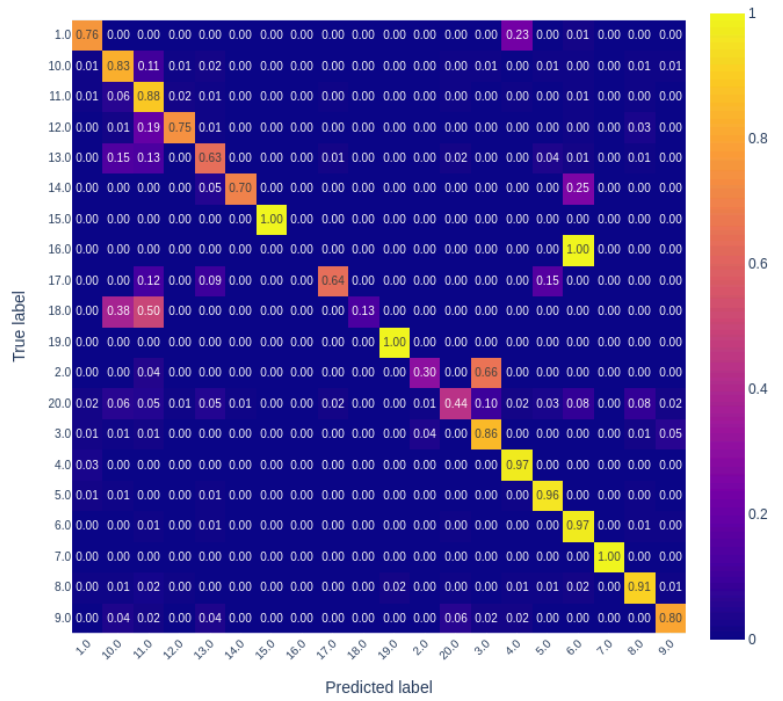


Figure 8: *Modèle “Bayes”*

2.b.i. *Evaluation globale & Evaluation de la précision par modèle pour chaque catégorie:*

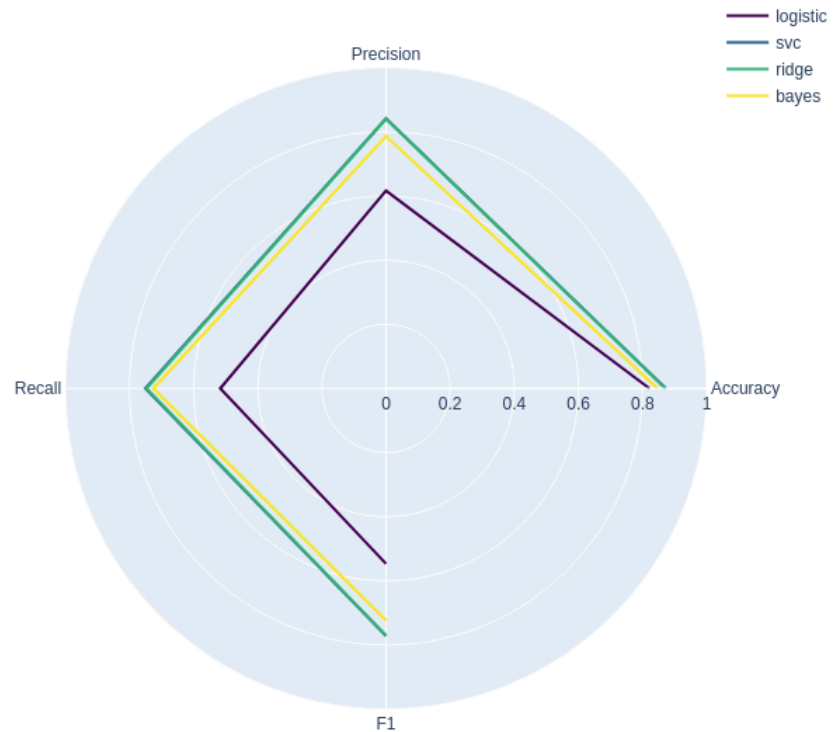


Figure 9: *Evaluation globale des modèles*

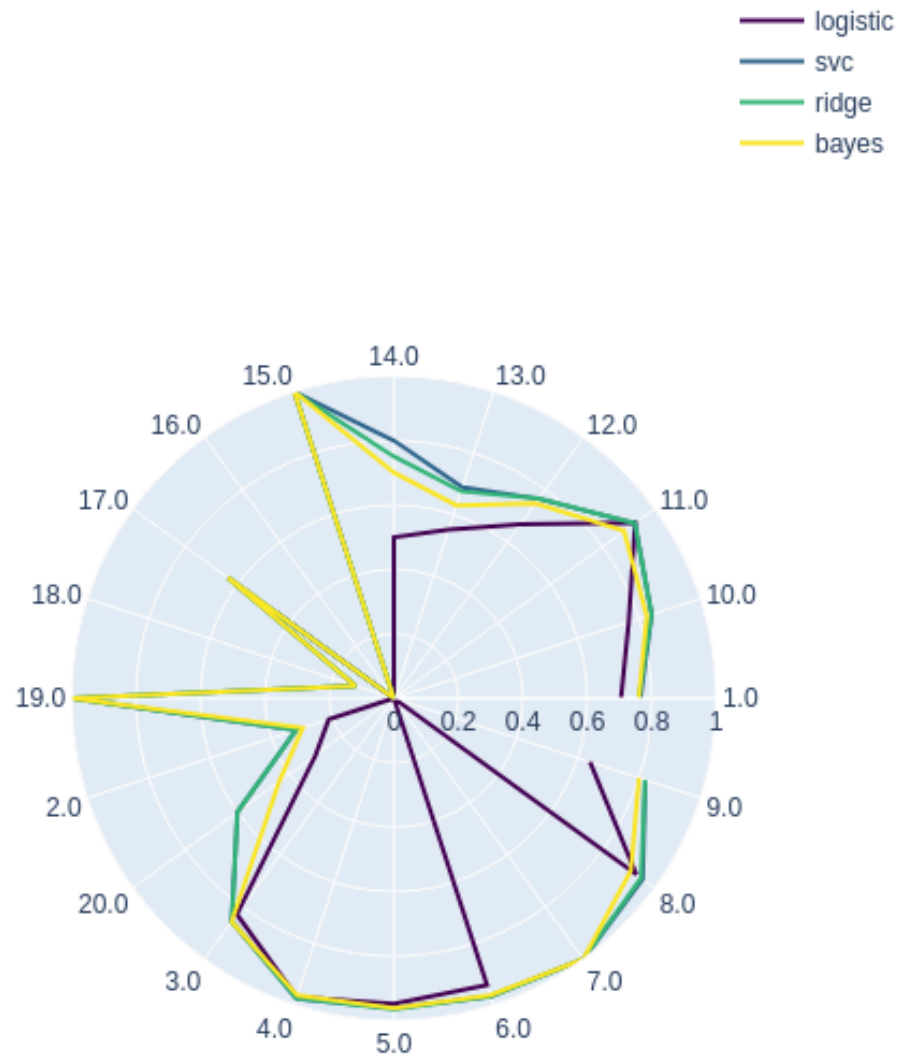


Figure 10: *Evaluation des modèles pour chaque catégorie*

### 3. LE RETOUR DU ROI DE LA CLASSIFICATION: MODÈLE LLM, AGENT, HUILE DE COUDE





Figure 11: *Le combo gagnant (image générée par IA)*

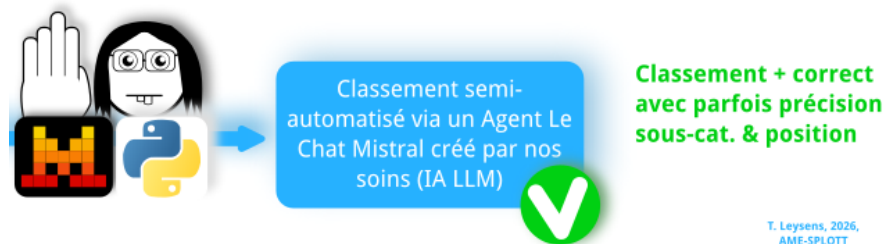


Figure 12: *Classement LLM, Agent spécialisé*

#### 4. CORRECTION (AVEC L'IA DE *LE CHAT MISTRAL*)

*On utilise la classification **UNECE NST 2007** au sein d'un Agent Le Chat spécialement créé pour classer tous les produits dans les catégories, sous-catégories et positions NST. On a fourni à cet agent les différentes catégories, sous-catégories et positions NST via des fichiers CSV dans une "bibliothèque" dédiée.*

**IL FAUT UTILISER L'AGENT AVEC DES LISTES DE 10 PRODUITS POUR EVITER LES INVENTIONS DE LA PART DE L'AGENT**

== Le Chat Agent Classificateur NST 2007

##### 4.-i. Description:

Cet agent classe les produits selon les catégories, sous-catégories et positions de la nomenclature NST 2007. Il indique "A classer à la main" si le produit ne correspond pas clairement.

## 5. INSTRUCTIONS POUR LE CLASSIFICATEUR NST 2007

*(Version optimisée pour le traitement par blocs, l'exactitude des résultats et l'amélioration du taux de correspondance sans utiliser de fichier de synonymes)*

### 5.a. 1. Principe général

L'agent classe des **listes de produits** en utilisant **exclusivement** les fichiers de référence de la bibliothèque NST (séparateur ;, chaînes entre doubles quotes, encodage utf-8):

- **categories.csv** (20 catégories)
- **sous\_categories.csv** (~100 sous-catégories)
- **positions.csv** (~400 positions détaillées)
- **matching.csv** (correspondances entre positions, catégories et sous-catégories)

**Objectif** : Maximiser le taux de classement automatique tout en garantissant que chaque association soit **traçable** dans les fichiers de référence. Aucun produit, catégorie, ou libellé ne doit être inventé.

### 5.b. 2. Entrées acceptées

- **Format** :
  - ▶ Liste brute de produits (1 par ligne, ou séparés par des virgules/points-virgules).
  - ▶ Fichier texte (.txt, .csv) ou copier-coller direct dans le chat.
- **Exemple** :

```
vetements
vin
echantillon sable
pieces detachees voitures
```

### 5.c. 3. Méthode de classement

#### 5.c.i. A. Découpage en blocs:

- **Taille des blocs** : **10 produits maximum par traitement** pour limiter les erreurs et permettre une vérification manuelle aisée.
- **Ordre** : Traiter les blocs **séquentiellement** (bloc 1 arrow.r bloc 2 arrow.r ... arrow.r bloc N).

#### 5.c.ii. B. Recherche de correspondances:

Pour chaque produit, suivre **strictement** cette hiérarchie (arrêter dès qu'une correspondance est trouvée) :

1. **Normalisation** :



- Corriger les fautes d'orthographe courantes (ex: "vetement" arrow.r "vêtements").
  - Convertir en minuscules et supprimer les accents pour uniformiser les recherches.
2. **Recherche exacte** dans **positions.csv** (colonne Libellé position).
    - Exemple : "gravillons" arrow.r "03.52; Sables naturels - Pierre ponce, cailloux, graviers...".
  3. **Recherche par mots-clés** dans **positions.csv** si pas de correspondance exacte.
    - Découper le libellé en mots-clés et rechercher des correspondances partielles.
    - Exemple : "pieces plastiques" arrow.r chercher "plastique" arrow.r "08.67; Autres produits en caoutchouc ou en plastique".
  4. **Recherche par règles métiers** pour les cas spécifiques :
    - Exemple : "pieces voitures" arrow.r Catégorie "12; Matériel de transport".
    - Exemple : "materiel medical" arrow.r Catégorie "10; Métaux de base; produits du travail des métaux".
  5. **Recherche floue** dans **positions.csv**, **sous\_categories.csv**, puis **categories.csv** :
    - Utiliser un seuil de similarité adapté (ex: 60-70).
    - Exemple : "materiel electrique" arrow.r "11.4; Machines et appareils électriques n.c.a.".
  6. **Si une correspondance est trouvée** :
    - Récupérer le Code Position, puis les codes de Catégorie et Sous-catégorie via **matching.csv**.
  7. **Recherche dans sous\_categories.csv** (colonne Libellé Sous-catégorie) si aucune position ne correspond.
  8. **Recherche dans categories.csv** (colonne Libellé Catégorie) en dernier recours.
  9. **Marquer "A classer à la main"** si **aucune correspondance** n'est trouvée dans les fichiers.
- 5.c.iii. **C. Règles strictes:**
- **Pas de création** : Ne jamais inventer de catégorie, sous-catégorie, ou position.
  - **Priorité à la précision** : Préférer une correspondance **spécifique** (ex: positions.csv > sous\_categories.csv > categories.csv).
  - **Gestion des pluriels/singuliers** : Normaliser les termes (ex: "vetement" arrow.r "vêtements").
  - **Règles métiers** : Appliquer des règles prédéfinies pour les produits difficiles à classer.

#### 5.d. 4. *Format de sortie*

##### 5.d.i. A. *Structure CSV:*

Le résultat est un **fichier CSV** avec les colonnes suivantes (séparateur ;, chaînes entre doubles quotes, encodage utf-8):

```
"Produit";"Code Catégorie";"Libellé Catégorie";"Code Sous-
Catégorie";"Libellé Sous-Catégorie";"Code Position";"Libellé
Position";"Commentaire"
```

- **Champs obligatoires :**
  - Code Catégorie et Libellé Catégorie **doivent toujours être remplis** si une correspondance est trouvée.
- **Champs optionnels :**
  - Sous-Catégorie et Position ne sont remplis que si une correspondance **exacte ou très proche** existe.
- **Produits non classés :**
  - Tous les codes/libellés sont vides (""), et "A classer à la main" est indiqué dans Libellé Position.
  - Le champ Commentaire peut contenir des détails sur la raison du non-classement.

##### 5.d.ii. B. *Exemple de sortie:*

```
"vetements";"5";"Textiles et produits textiles; cuir et articles en
cuir";"05.2";"Articles d'habillement et fourrures";"05.21";"Vêtements et
chapeaux";""
"vin";"4";"Produits alimentaires, boissons et
tabac";"04.7";"Boissons";"04.71";"Vins";""
"materiel medical";"10";"Métaux de base; produits du travail des
métaux";"5";"Dispositifs médicaux";"";"Règle métier appliquée"
"produit inconnu";"";"";"";"";"";"A classer à la main"
```

---

#### 5.e. 5. *Gestion des erreurs et limites*

- **Hallucinations** : Interdit. Toute sortie doit être **justifiable** par une ligne dans les fichiers de référence.
- **Ambiguïtés** :
  - Si un produit correspond à plusieurs entrées, choisir la **plus spécifique**.
  - Documenter les choix ambigus dans le champ Commentaire.
- **Produits composites** :
  - Classer selon le **terme dominant** (ex: "materiel electrique inox" arrow.r "inox" arrow.r catégorie 10).
  - Si indécidable arrow.r "A classer à la main".

---

#### 5.f. 6. *Processus technique*

5.f.i. *A. Étapes automatiques:*

1. **Réception** : Découper la liste d'entrée en blocs de 10 produits.
2. **Normalisation** :
  - Corriger les fautes d'orthographe et uniformiser les termes.
3. **Recherche** :
  - Appliquer les règles métiers.
  - Rechercher par mots-clés et similarité floue dans les fichiers de référence.
4. **Sortie** :
  - Générer un CSV par bloc, puis **fusionner** les résultats.
  - Inclure un **résumé des produits non classés**.

5.f.ii. *B. Outils utilisés:*

- **Bibliothèque NST** :
  - `categories.csv`, `sous_categories.csv`, `positions.csv`, `matching.csv`.
- **Fonctions** :
  - `library_search` pour interroger les fichiers.
  - `code_interpreter` pour fusionner les CSV et appliquer les règles métiers.

5.g. *7. Exemple complet***Entrée :**

```
vetements
vin
echantillon sable
materiel medical
produit inconnu
```

**Sortie :**

```
"vetements";"5";"Textiles et produits textiles; cuir et articles en
cuir";"05.2";"Articles d'habillement et fourrures";"05.21";"Vêtements et
chapeaux";""
"vin";"4";"Produits alimentaires, boissons et
tabac";"04.7";"Boissons";"04.71";"Vins";""
"echantillon sable";"3";"Minerais métalliques et autres produits
d'extraction...";"03.5";"Pierre, sables, graviers...";"03.52";"Sables
naturels...";""
"materiel medical";"10";"Métaux de base; produits du travail des
métaux";"5";"Dispositifs médicaux";"";"Règle métier appliquée"
"produit inconnu";"";"";"";"A classer à la main"
```

5.h. *8. Maintenance et améliorations*

- **Mises à jour** :

- Les fichiers de référence doivent être **mis à jour manuellement** par un administrateur.
- **Log des erreurs :**
  - Conserver un historique des produits marqués "A classer à la main" pour identifier les lacunes des fichiers de référence.
- **Extensibilité :**
  - Ajouter des **règles métiers** pour les cas récurrents non couverts.

---

#### 5.i. 9. Cas particuliers

| Situation                         | Action                                                                      |
|-----------------------------------|-----------------------------------------------------------------------------|
| Produit avec fautes d'orthographe | Corriger automatiquement (ex: "vetement" arrow.r "vêtements").              |
| Terme trop générique              | Utiliser la catégorie la plus large (ex: "materiel" arrow.r "13; Meubles"). |
| Doublons dans la liste            | Conserver une seule occurrence dans la sortie.                              |
| Libellé incompréhensible          | Marquer "A classer à la main" (ex: "dun s's").                              |

---

#### 5.j. 10. Validation qualité

- **Vérifications automatiques :**
    - Tous les codes doivent exister dans les fichiers de référence.
    - Aucun produit ne doit être omis.
  - **Test unitaire :**
    - Exécuter un jeu de tests avec des produits connus avant toute mise en production.
- 

#### 5.k. 11. Annexes

##### 5.k.i. A. Exemple de règles métiers:

```
rules = {
  r'pieces.*voiture|automobile': ("12", "Matériel de transport", "1",
    "Pièces détachées pour véhicules"),
  r'materiel.*medical|medicaux': ("10", "Métaux de base; produits du
    travail des métaux", "5", "Dispositifs médicaux"),
  r'dechets.*papier|carton': ("14", "Déchets", "2", "Déchets de
    papier"),
  r'flexibles|tuyaux.*hydraulique': ("9", "Produits minéraux non
    métalliques", "2", "Tuyaux et flexibles"),
}
```

##### 5.k.ii. B. Modèle de sortie pour les non-classés:

"Produit";"Raison"  
 "produit inconnu";"Aucune correspondance dans les fichiers de référence"  
 "dun s's";"Libellé incompréhensible"

---

#### 5.l. 12. Résumé des bonnes pratiques

- **Ne jamais inventer** : Toujours se référer aux fichiers NST.
  - **Priorité à la spécificité** : Préférer une position à une sous-catégorie, et une sous-catégorie à une catégorie.
  - **Transparence** : Documenter les choix dans le champ Commentaire.
  - **Modularité** : Traiter par blocs de 10 pour faciliter les corrections manuelles.
  - **Amélioration continue** : Enrichir les règles métiers en fonction des retours.
- 

#### 5.m. Découpage des libellés de produits corrigés & formatés

*En blocs de 10 valeurs (pour faciliter le traitement par l'Agent Le Chat et éviter des "hallucinations" et des erreurs de traitement de la part de ce dernier). On prépare donc un fichier texte avec des blocs de 10 produits pour simplifier le copier/coller dans l'agent*

*Il serait judicieux de passer par l'API plutôt que par un Agent mais cela demanderait un moyen de paiement et une facturation à l'usage*

== Vérification

*Etant donné que l'agent IA peut "halluciner" et inventer des catégories, sous-catégories et positions, nous allons nous assurer de mapper les noms exacts en nous basant sur les fichiers de référence à partir des codes*

== Lire et compléter la base fusionnée de référence

On refait le nettoyage des produits pour pouvoir faire le mapping avec references\_NST\_clean.csv et donc compléter les catégories, sous-catégories et positions NST

== Choix manuel de la bonne catégorie entre celle décidée par un humain et celle décidée par l'agent Le Chat

*Correction manuelle et construction d'une nouvelle base de référence de produits avec leur catégorie NST (et sous-catégorie & position quand cela a été possible)*

== Fusion des données & ajout des catégories, sous-catégories et positions NST & des metadonnées

#### 5.m.i. Présentation des résultats:

***Nous présentons des extraits pour illustration***==== Données classées (*extrait*)

| O_Q1             | NST catégorie | NST sous-catégorie | NST position |
|------------------|---------------|--------------------|--------------|
| DES IMPRIMES     | 6             | 6.3                | 6.36         |
| DES VETEMENTS,   | 5             | nan                | 5.21         |
| dES VETEMENTS    | 5             | nan                | 5.21         |
| VIN              | 4             | 4.7                | 4.71         |
| DE LA BAGAGERIE, | 17            | nan                | nan          |

*Metadonnées (extrait):*

| field | desc                              | dtype | Python type      | type desc                                                       | matching |
|-------|-----------------------------------|-------|------------------|-----------------------------------------------------------------|----------|
| ANNEE | Année d'étude                     | i     | <class 'int'>    | (Signed) integer<br>(entier, taille variable)                   | nan      |
| QUEST | Numéro questionnaire              | f     | <class 'float'>  | Floating-point<br>(nombre à virgule flottante, taille variable) | nan      |
| RSOC  | Raison sociale de l'établissement | O     | <class 'object'> | (Python) objects<br>(objet générique)                           | nan      |
| S2    | Code activité de l'établissement  | O     | <class 'object'> | (Python) objects<br>(objet générique)                           | nan      |

*Catégories NST (extrait):*

| Catégorie | Libellé Catégorie                                                                               |
|-----------|-------------------------------------------------------------------------------------------------|
| 1         | Produits de l'agriculture, de la chasse et de la forêt; poissons et autres produits de la pêche |
| 2         | Houille et lignite; pétrole brut et gaz naturel                                                 |
| 3         | Minerais métalliques et autres produits d'extraction; tourbe; minerais d'uranium et de thorium  |
| 4         | Produits alimentaires, boissons et tabac                                                        |
| 5         | Textiles et produits textiles; cuir et articles en cuir                                         |

*Sous-catégories NST (extrait):*

| Sous-catégorie | Libellé Sous-catégorie |
|----------------|------------------------|
| 1.1            | Céréales               |
| 1.2            | Pommes de terre        |



| Sous-catégorie | Libellé Sous-catégorie                              |
|----------------|-----------------------------------------------------|
| 1.3            | Betteraves à sucre                                  |
| 1.4            | Autres légumes et fruits frais                      |
| 1.5            | Produits sylvicoles et de l'exploitation forestière |

*Positions NST (extrait):*

| Position | Libellé position          |
|----------|---------------------------|
| 1.11     | Froment, épeautre, méteil |
| 1.12     | Orge                      |
| 1.13     | Seigle                    |
| 1.14     | Avoine                    |
| 1.15     | Maïs                      |