Journées de méthodologie statistique de l'Insee 2022

Fuzzy matching on big-data

An illustration with scanner data and crowd-sourced nutritional data

Lino Galiana (*) , Milena Suarez Castillo (**)

(*) Insee, Département des Etudes Economiques
(**) Insee, SSP-lab

lino.galiana@insee.fr, milena.suarez-castillo@insee.fr,

Mots-clés. Fuzzy matching, ElasticSearch, natural langage processing, Siamese neural networks, word embeddings.

Domaines. 5.1 ; 5.2 ; 11.4 ; 11.5.

Abstract

Food retailers' scanner data provide unprecedented details on local consumption, provided that product identifiers allow a linkage with features of interest, such as nutritional information. In this paper, we enrich a large retailer dataset with nutritional information extracted from Open Food Facts, completed with the ANSES Ciqual dataset. To compensate for imperfect matching through the bar code, we develop a methodology to efficiently match short textual descriptions. After a preprocessing step to normalize short labels, we resort to fuzzy matching based on several tokenizers (including n-grams) by querying an ElasticSearch customized index and validate candidates echos as matches with a Levenstein edit-distances. The pipeline is composed of several steps successively relaxing constraints to find relevant matching candidates. We finally develop a similarity based on a word embedding obtained by training a siamese network on bar code matches. This alternative measure is used to evaluate our final matching.

Résumé

Les données de caisse pourraient offrir une description très riche de la consommation locale, à condition que les identifiants des produits permettent des enrichissement avec des sources externes, comme des informations nutritionnelles. Dans cet article, nous enrichissons des données de la grande distribution avec des informations nutritionnelles extraites d'Open Food Facts, complétées par les données Ciqual de l'Anses. Pour compenser l'appariement partiel via le code-barre, nous développons une méthode permettant d'apparier efficacement des libellés courts. Après une étape de prétraitement pour normaliser des libellés courts, nous utilisons des techniques d'appariement floue basée sur plusieurs tokenizers (y compris les n-grammes) en interrogeant un index personnalisé ElasticSearch et en validant les échos candidats avec une distance de Levenstein. Le pipeline est composé de plusieurs étapes relachant successivement les contraintes pour trouver des candidats pertinents. Enfin, nous développons une mesure de similarité basée sur un word embedding obtenu en entraînant un réseau siamois sur l'appariement exact via les code-barres. Cette mesure alternative est utilisée pour évaluer notre appariement final.

1 Introduction

Socio-territorial inequalities in health have been the subject of sustained attention for several years. Recently, some diet-related pathologies, notably obesity, have been identified as an important risk factors in the development of severe forms of Covid-19 (Gao et al. 2021). In France, these co-morbidities are more frequent in low-income groups (Barhoumi et al. 2020) which may partly explain socioeconomic inequalities in Covid-19 burden (Galiana et al. 2022). Besides the renewed interest in obesity distribution following Covid-19, a larger body of epidemiological and economic research has sought to understand the causes and effects of nutritional inequalities.

In recent decades, the share of processed products in consumer basket has has grown, while raw foods share decreased (Caillavet et al. 2019; Nichèle et al. 2008; Larochette and Sanchez-Gonzalez 2015). In the United States, inequalities in nutritional quality across social groups have increased during the years 2000-2010 (Wang et al. 2014). As for France, food quality has become slightly more uniform, but the differences between high and low income persist (Caillavet et al. 2019). These persistent nutritional inequalities have led to public policies such as the adoption of synthetic indices aimed at improving the understanding of the nutritional quality. The most famous measures are the Nova classification (measure scaling from 1 to 4 and representing the degree of transformation of a product) and the Nutri-score (a score from A to E built by valuing positively the nutrients to be favored, for instance fiber, and negatively the nutrients to be avoided, for instance saturated fat). These two indices are based on the work of nutrition researchers (see, for example, Julia et al. (2018) for the Nutri-score). In the United States, the program *"Let's Move"* that was promoted by by Michelle Obama has led to questions about the causes of food inequality, in its social and spatial dimensions. Bitler and Haider (2011) and Allcott et al. (2019) are thus seeking to determine, the factors explaining the different availability of healthy products at affordable prices (the problem of *food deserts*). According to Allcott et al. (2019), affluent people can consume healthier products because supermarkets where they consume have adapted to the greater value that these populations place on these products and offer a more diversified range of healthy products. In France, the map of consumption quality is still to be done.

In order to produce this detailed panorama of the consumption of fatty, salty and sweet foods it is necessary to enrich consumption data with nutritional characteristics. Two sources can be used to obtain consumption data: surveys or scanner data. For example, Caillavet et al. (2019) use, in a longitudinal way the budgetary items of the INSEE "Family Budget" survey in order to determine the nutritional composition of French household consumption over a long period of time. The disadvantage of this survey is that it breaks down food consumption into aggregate items. Thus, within some items, the heterogeneity of nutritional quality is ignored. Dedicated food consumption surveys may be suject to small sample size, are resource-intensive and possibly

subject to misreporting (Bandy et al. 2019). The use of scanner data - when systematically collected by food retailers - can overcome these limitations and provide new insights at detailed geographical level. For some years now, scanner data from supermarket chains have been used to compute consumer price indices. Besides price evolution, these data also provide information on the composition of food consumption in France and can thus be used to assess the spatial heterogeneity of consumer baskets. However, these data need to be enriched with an external source in order to have, in addition to the sales figures for each product nutritional characteristics (weight, units, calories, etc.).

The methodological contribution of this study is the enrichment of big-data scanner data with several sources of information using advanced fuzzy matching methods. The scanner data used were provided by the startup `RelevanC`, subsidiary of the Casino group, and cover the entire field of this group (mainly Casino, Franprix, Monoprix and Leader Price) over the 2017-2018 period. These data are enriched with the databases `Open Food Facts` (*crowd-sourced* dataset of food products) and `Ciqual` (nutritional table of ANSES). When the product identifier, called EAN code (equivalent to a bar code), does not allow a systematic matching between sources, fuzzy matching methods, exploiting the proximity between textual labels, are mobilized.

2 Data

Figure 2.1 depicts the structure of the datasets explaining the relationships between the different sources. Each dataset is going to be detailed afterwards. This paper will focus on the enrichment of product level data. Most of the issues at stake presented in this paper also apply to linkage that rely on the geographic dimension.

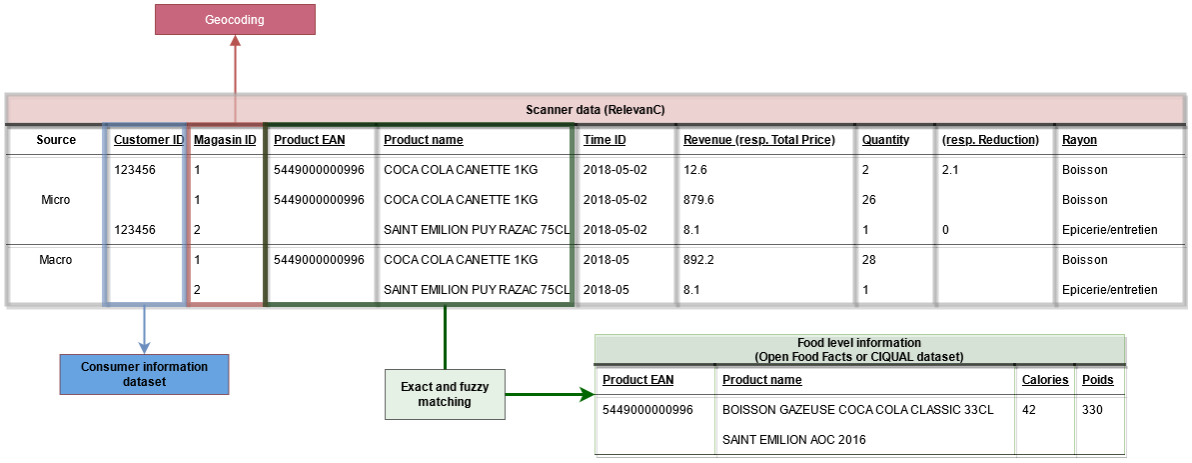

Figure 2.1: Datasets schema and relation between them

2.1 Scanner data: `RelevanC` dataset

We use supermarket data provided by a start-up called `RelevanC` that belongs to the `Casino` group.[1] Casino group consists of several supermarket chains that are specialized by geographic areas : Franprix and Monoprix are mostly urban stores while Casino presents both small groceries shop in city centers and larger supermarkets in suburbs or in rural areas. Store-based data record sales by day and product identifier: a bar code (EAN code) and a short text label describing the product. Individual-based data records sales by fidelity card, product identifiers and date. Table 2.1 depicts the annual revenue of each group. Overall, the Casino group represents around 10 billion euros.

[1] In 2019, according to Kantar, Casino group represented 8.7% of market share in France, see https://www.kantarworldpanel.com/fr/grocery-market-share/france/snapshot/15.01.19/

The dataset provided by RelevanC presents billions of transactions. By consolidating the dataset over our time period between different chains, we end up with approximately 250,000 unique products that need to be matched to external sources.

Supermarket	Revenue (billion euros)
Casino	6
Franprix	1
Monoprix	3

* To ensure confidentiality, revenues are rounded to the closest billion

Table 2.1: Supermarket revenues in 2018

2.2 Crowd-source nutritional data: Open Food Facts

Open Food Facts is a collaborative project and a *crowd-sourced* open access database of food products characteristics. This French initiative contains data on more than 1,900,000 products. Contributors add products by scanning the barcode and sending photograph of the product information available on its packaging. Thus, records may include barcode, commercial name, brand, labels and certification such as the Nutri-Score or the Nova classification, manufacturing place, ingredients list, nutrient levels for a given quantity (energy, fat, saturated fat, carbohydrates, sugars, fiber, proteins, salt …). Chazelas et al. (2020) presents an overview of the characteristics of the products in the database, in particular regarding the additives.

Open Food Facts provides nutritional information for 1,943,892 products. The energy value of each product, measured by the number of kcal per 100 grams, is available for 1,502,324 products (77.3 % of values). As can be seen in Figure 2.2, most features have the same completion rate. However, only 78,768 products have all the characteristics filled (4.1 % of values). Restricting ourselves to products with complete information would lead to very low matching rates. We thus focus, for each characteristic, on the products presenting the information to ensure the consistency of the matching.

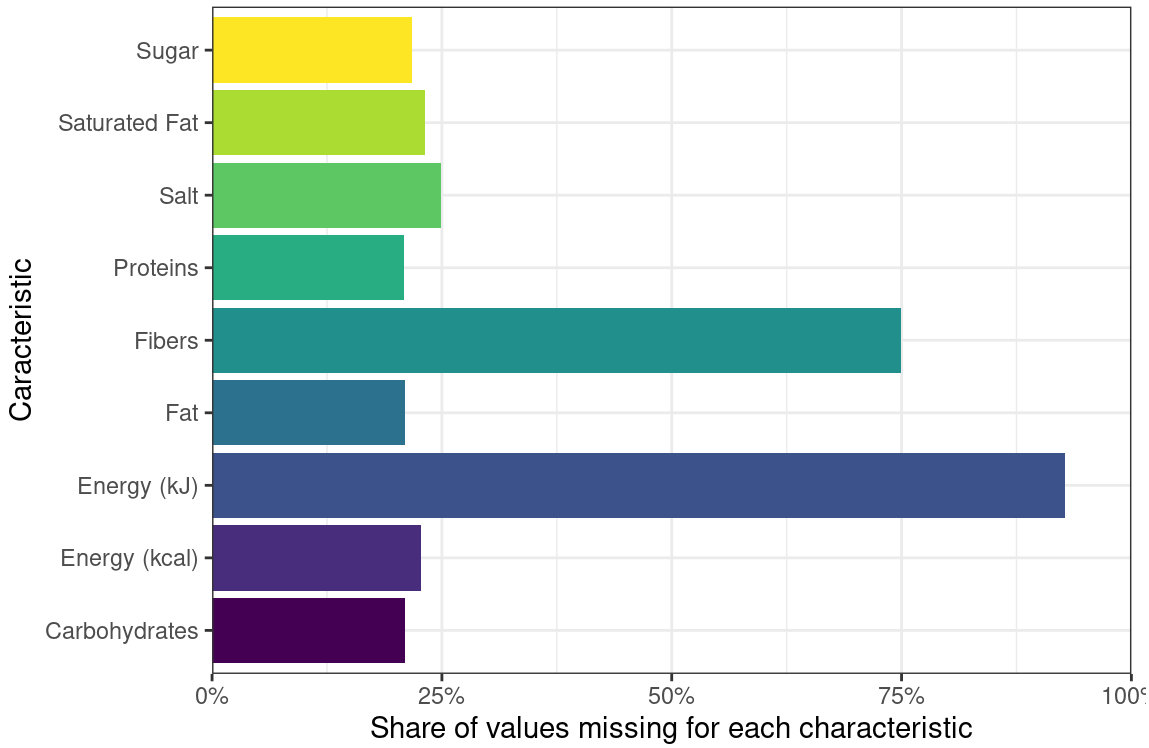

Figure 2.2: Share of missing nutritional values in Open Food Facts

2.3 Administrative data (CIQUAL) augmented with scrapped dictionnaries of products (`MediaWiki`)

The French Agency for Food, Environmental and Occupational Health & Safety `Anses` is a French government agency whose mission includes assessing food health risks. As part of its public mandate, the agency produces reference tools to monitor the nutritional quality of food. The Food Quality Information Center (Ciqual) publishes the eponymous table, a reference tool for the nutritional composition of foods consumed by the French population. It can be consulted and downloaded free of charge on the Internet[2]. About 4,000 generic products are described is terms of energy, protein, glucides, lipids, sugars etc.

Ciqual product entries are generic e.g. *"White wine (dry)"* or *"Dry goat cheese"*. This generic wording thus differs from a typical scanner data label entry which tends to include brands or designations of origin. We thus map generic entries to `Wikipedia` category members[3] for beers, wine, cheese, charcuterie, chips, coffee, meat pieces and liquor (whisky, rhum etc.). About 3,200 products entries are built, under their wikipedia designation with ciqual-derived nutrional quality, leading to a Ciqual-augmented database.

2.4 Additional elements

[2] https://ciqual.anses.fr/

[3] For instance, the ciqual entry "White beer" is mapped to any of the entities belonging to the Wikipedia catégorie "White beer", such as "Edelweiss" or "Moulins d'Ascq" as found here https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:Bi%C3%A8re_blanche. The ciqual entry "Potato chips, plain or flavored, standard" is mapped to Wikipedia categorie "Chips brand", such as "Cheetos", "Doritos", "Lay's" or "Vico" https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:Marque_de_chips

2.4.1 `Fasttext` model for COICOP classification

The *Classification of Individual Consumption According to Purpose* (COICOP) is one of the functional classifications of expenditures used in the National accounts system. This classification is organized by types of products. It allows to know the expenses which households dedicate to food, health, education etc. We use a classifier of short label into the COICOP developed at Insee. A neural-network model has been trained with `Fasttext` on scanner data labels of large retailers in France to classify into the 6-digits COICOP after a preprocessing stage aimed at normalizing labels. We pass all product labels through the preprocessing and classifier steps to obtain a prediction of its 6-digits COICOP class.

> **Remarque**
>
> The `predicat` project developped at Insee uses a `Fasttext` neural-network model to make a vector representation of labels based on a large scale embedding. The API is showcased at https://api.lab.sspcloud.fr/predicat/. Source code is available on Github[4]. Slides from an internal presentation of this project (*"Le Machine Learning pour coder dans une nomenclature : les libellés des tickets de caisse"*) are available on Insee's website[5]. The model was trained on more than 23 million products from several large retailers and performance was evaluated on more than 2.5 million products. Unweighted accuracy, precision, recall and F1-score were respectively 0.85, 0.86, 0.85 and 0.85. Weighted by product sales, the same metrics were 0.91, 0.92, 0.91 and 0.91.

2.4.2 IRI Dataset

IRI is a company that collects information on grocery stores and products on a weekly basis. Insee uses some of the provided information for the computation of consumer price index (CPI). The IRI database does not present much information on nutrients but offers some interesting information to compare products, including weight. Further investigations should use this information, in addition to insights that can be extracted from the product name.

2.5 Pipeline of feature engineering

[4] https://github.com/InseeFrLab/product-labelling
[5] https://www.insee.fr/fr/information/5357816

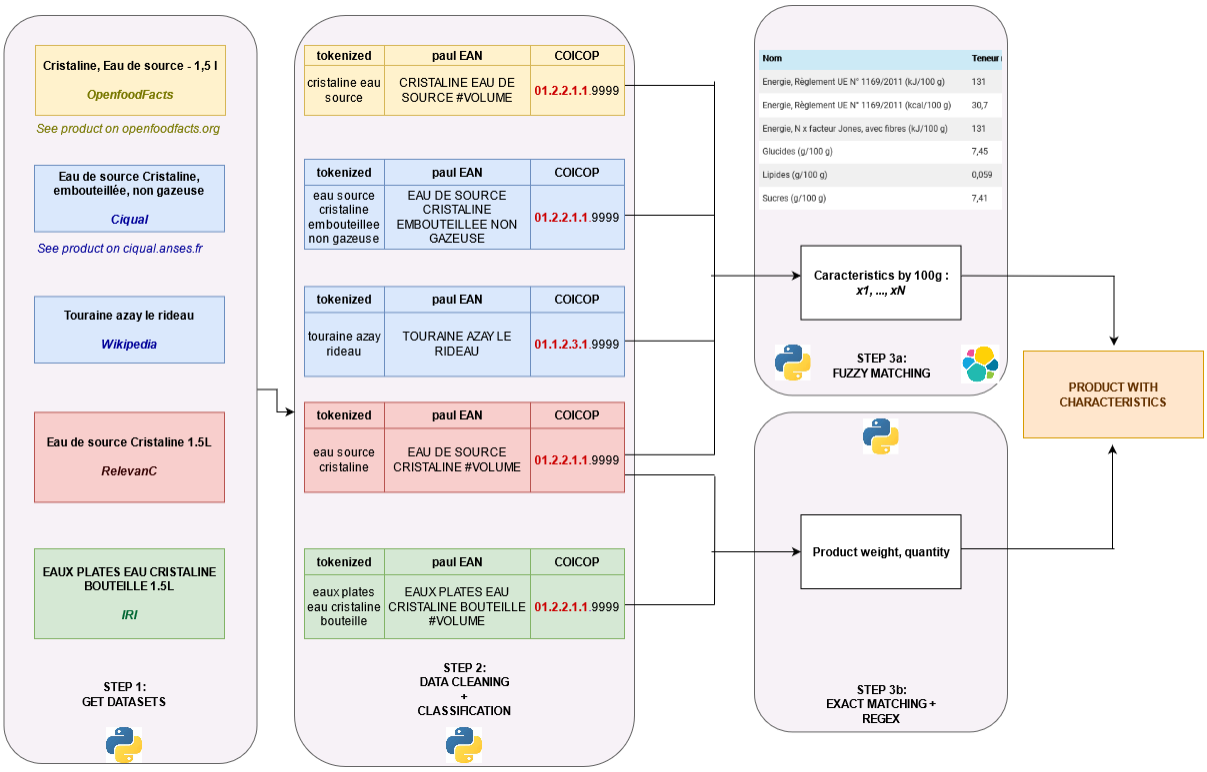

Figure 2.3: Macro view of the pipeline

Figure 2.3 depicts the general approach we followed. The final objective of our approach is to enrich scanner-data with product-level features such as nutritional characteristics (step 3a in Figure 2.3). To make fuzzy matching relevant, some effort must be spent in harmonizing textual data to reduce the noise in the voluminous corpus we use (step 2 in Figure 2.3).

2.5.1 Preprocessing

The objectives of preprocessing are multiple. First, preprocessing helps to eliminate non-food products in diversified groups (e.g. groceries) from a dictionary of words or brands that cannot be associated with food (e.g. *"Whiskas"*). However, the main benefit of preprocessing is to harmonize our different corpus by eliminating elements that bring noise in the matching (stop words, punctuation, etc). Preprocessing is also useful to retrieve information within product labels such as weight, volumes, or the number of units in package. This type of information can be removed from the label used for linkage between sources but conserved for latter use since it is a useful information to compare products.

Figures 2.4 and 2.5 illustrate the interest of the preprocessing steps. Initial corpus do not share many common features and are filled with information on products that are not relevant to identify it. Conversely, the final datasets are more suited for linkage since they exhibit more frequently elements that are useful to identify a product.

Figure 2.4: Word clouds before (left) and after (right) preprocessing

Figure 2.5: Word clouds before (left) and after (right) preprocessing

Table 2.2 shows some examples to illustrate some steps involved in our preprocessing pipeline: removing stopwords, removing volume or packaging information (e.g. number of units) and tokenization. Most of them are quite standard in natural langage processing (NLP) but some of them, for instance, identifying and removing packaging information, are specific to our problem.

Original text	Tokenized text	Tokenized text for classification
Cristaline, Eau de source - 1,5 l	cristaline eau source	CRISTALINE EAU DE SOURCE #VOLUME
Eau de source Cristaline, embouteillée, non gazeuse	eau source cristaline embouteillee non gazeuse	EAU DE SOURCE CRISTALINE EMBOUTEILLEE NON GAZEUSE
Touraine azay le rideau	touraine azay rideau	TOURAINE AZAY LE RIDEAU
Eau de source Cristaline 1.5L	eau source cristaline	EAU DE SOURCE CRISTALINE #VOLUME
EAUX PLATES EAU CRISTALINE BOUTEILLE 1.5L	eaux plates eau cristaline bouteille	EAUX PLATES EAU CRISTALINE BOUTEILLE #VOLUME
Eau de source Cristaline 6X1.5L	eau source cristaline	EAU DE SOURCE CRISTALINE #LOT #VOLUME

Table 2.2: Examples of preprocessing output from our pipeline

2.5.2 Classification

In linkage theory, blocking is a technique to reduce the computational burden (Steorts et al. 2014). This principle can also be useful in fuzzy matching where the computational issue is even more important. In our case, the number of pairs to compare is huge (we want to find around 200 000 products into a corpus of 2 million entries). The number of possible pairs to compare has an order of magnitude of 10^{13} (number of cells in the human body). This is very computationally intensive and without a search engine approach this would

not be feasible. In order to reduce the number of potential pairs, we used the fact that it is more efficient to search first for a product in a set of similar products. Since we cannot map categories between data providers, we use a harmonized classifier that comes from a functional consumption nomenclature, the COICOP code.

Since the COICOP code of a product is unknown, we used an automatic classifier for supermarket products based on the label of the product that has been trained in another Insee's project.[6] This automatic classification is based on a deep learning model with a `Fasttext` architecture, trained on exhaustive supermarket data (food and non-food products) used to built the consumer price index (CPI). Without having access to the trained data that is reserved for the computation of CPI, we used the binary of the `Fasttext` model that is sufficient for prediction purpose. The typology obtained is very fine and represents well, for a given product, those towards which one would like to search in priority. For instance, the `01.2.2.1.1` label represents the *"Eaux minérales et de source"* group (Mineral and spring waters).

Table 2.3 depicts some random examples of classified products. Generally, the model is very good at putting products in the right category. When the model misclassifies a product, it is usually when a word has been misunderstood. For instance, in French, a "potée" is a meal that generally mixes several vegetable (mostly cabage and potatoes). However, the *"potee auvergnate"* is a special version where vegetable and meat are mixed. This helps understand why *"POTE AUVERGNATE 400G"* is missclassified as a vegetable product. Misclassification is nevertheless a minor problem because products for which no satisfactory match is found in the same group will be caught up in the next step of the matching procedure.

Libellé d'origine	Libellé tokenisé	COICOP	Label
LE PANIER FAISSELLE BIO 4X100G	panier faisselle bio	01.1.1.5.1.9999	Bread and cereals
NAVARIN AGNEAU 1,2KE	navarin agneau	01.1.2.8.3.0010	Meat
SAUMON SAUVAGE PROV.MSC 330G BQ	saumon sauvage prov msc	01.1.3.6.1.9999	Fish and shellfish
ABRICOT 35/45 BQ 1KG	abricot	01.1.6.3.1.0005	Fruits
POTE AUVERGNATE 400G	pote auvergnate	01.1.7.6.1.9999	Vegetables
MIEL ROMARIN HAUTE VALLES 480G	miel romarin haute valles	01.1.8.4.1.0016	Confectionery and frozen products
ENTREMETS CITRON MERINGUE 6P 500G	entremets citron meringue	01.1.8.5.1.9999	Confectionery and frozen products
HERBE MENTHE POT	herbe menthe pot	01.1.9.2.1.0017	Salt, spices and sauces
COCA-COLA ZERO PET 1.5LX6 CONT MAST	cocacola zero pet cont mast	01.2.2.2.1.0006	Other soft drinks
BISTROT DE FRANCE RS BIB 5L	bistrot france rs bib	02.1.2.1.1.0004	Wines, ciders and champagne

Table 2.3: Exemple of classification in COICOP nomenclature

3 Linkage methodology

We use a strategy to go from the most certain matching identifier (barcode) to the least certain. As we will develop later, at each step of our linkage procedure we use several criteria to validate or reject a pair to limit the selection of false positives. Using distance measures of different nature (edit distance and embedding-based) allows to take into account that two products can be similar despite very different names.

We first describe the main concepts of textual similarity used in our linkage procedure to then given an overview of its main steps. We then detail each linkage steps with the technology used, how we retrieve match candidates and validate or not a matching pair.

[6] The source code of this project can be found in `InseeFrLab`'s `Github` repository (https://github.com/InseeFrLab/predicat)

3.1 Fuzzy matching principle

Fuzzy matching consists in linking different textual designations of the same entity (here a supermarket product) by finding similarities between a text entry and one or more entries in a collection of records (here, nutritional databases). We used different approaches of approximate similarity to gain flexibility when conservative approaches were not able to link a product between scanner and nutritional datasets. There are mainly three families of distance in the natural langage processing (NLP) field:

1. Edit distances such as the **Levenshtein distance** define distance between two strings by counting deletions, insertions and substitutions needed to get the two string exactly match. We use as a validation distance the Levenshtein distance normalized by the size of the shortest string to take into account the difference in length between strings.[7]
2. Some words help to match products because they are rare. For instance, in terms of Levenshtein distance, `Nutella` and `Nuts` are not far. However, when we see Nutella in scanner data, we can not match any Nutella product in nutritional data to Cashew Nuts, Hazelnuts, etc. In that situation we want the distance to reflect how important is a specific word for matching in a collection of text records. **TF-IDF (term frequency–inverse document frequency) distances** are appropriate for that objective. If `Nutella` is relatively rare over the corpus, while `Nuts` appears more frequently, the inverse distance frequency downweights `Nuts` over `Nutella`.
3. Similar products might share very different labels between datasets. In that case, both Levensthein and TF-IDF will fail to understand the similarity between products. **Word embeddings** are intended to transform semantically close labels into neighbors vectors in a high dimensional space, while labels do not necessarily share common words. This approach makes it possible to match textual fields by nearest neighbor techniques in multidimensional spaces but also to transform textual relations into numerical operations (one of the best known examples is the embedding $\text{king} - \text{man} + \text{woman} = \text{queen}$ mentioned by Vylomova et al. (2015)).

To abstract from spelling mismatch and from word order in these frequently abbreviated labels, it is possible to use the overlap of *n-grams* of characters, that are sub-sequences of *n* characters from entry and candidates.

3.2 Overview of linkage steps

We use the following procedure, which we detail afterwards :

1. Whenever possible, we use the EAN code to link scanner data with Open Food Facts. However, as it sometimes happens that the EAN codes are erroneous, we validate this pair only if it brings close pairs in the sense of at least one distance, edit-based (normalized Levensthein) or embedding-based (cosine similarity in a customized embedding space).
2. For the remainder, we resort to fuzzy matching by linking labels in scanner data with Open Food Facts. We validate the matching if both edit-based and embedding-based distances are below a given threshold, deliberately conservative. It should be noted that the obtained matches are only validated based on these two pair-wise distances, but other similarity tools (e.g. n-grams and tf-idf) are used to retrieve match candidates. Indeed, the search engine technologies that we use to retrieve the most relevant echo gives more freedom in the evaluation of a textual distance than a classical edit distance allows. To speed-up computations at this step, we only confront products in scanner-data to counterparts that belong to the same COICOP group ;
3. For the remainder, we once again use fuzzy matching with the Open Food Facts dataset and validate using the same Levensthein and embedding distances. However, in this step, we confront each products in scanner-data with the whole Open Food corpus to allow for COICOP misclassification ;
4. For the remainder, we use fuzzy matching with our augmented CIQUAL dictionary. This means that products at this step are linked with a general entry. For instance, we seek to match red wine appellations to a unique generic entry *"Red wine"*. We thus favor a generic match over no match, even though in this last step, nutritional heterogeneity is underestimated. It turns out that this step recovers many alcohol products.

In other words, we prioritize Open Food Facts so as to be as precise as possible, but when failing, we allow for searching for a generic product entry in augmented-Ciqual.

3.3 Exact matching procedure

[7] We rely on a `Python` package called `rapidfuzz` for the implementation.

EAN code is a unique European identifier that is printed on barcodes. Open Food Facts being a crowd-sourced dataset, the EAN code has to be filled by users to be available. The EAN code being a sequence of 13 digits rather cumbersome to record, it is prone to errors. Product name and nutritional information important to consumers are more frequently recorded.

RelevanC	Open Food Facts
FROMAGERIE BLANC CAMPGNE	LA FROMAGEE DU BERRY
CREME DESSERT CHOCO	LPG CREME DESSERT AU CHOCOLATPARFUM
GAILLAC PERLE CS BL	GAILLAC PERLE
POM POT BRAS FRUIT RGE	POMPOTES BRASSES FRUITS ROUGE
BIO FAUX FILET CHARAL 175G	FAUX FILET DE BOEUF BIO
BOITE CERISES NOIRES	CHOCOLATS GANACHES CERISES
QUINOA BLANC	QUINOA
BEURRE AUX TRUFFES	BEURRE AUX TRUFFES
SCISSE COCTAIL PORC	SAUCISSES COCKTAILS PUR PORC SAUCE KETCHUP
RICARD FA18	PASTIS DE MARSEILLE
GAUFRE GOUD NOIX CO	GAUFRETTES GOUDA NOIX CO
TARTIFLETTE WILLIAM SAURIN	LA TARTIFLETTE AU REBLOCHON

Table 3.1: Exemples of products matching through their EAN code

Overall, after removing some false positives, the EAN code allows to match exactly 117,889 products (46.5% of products). Products for which consumers register the EAN are more likely to be frequently consumed products. The 46.5% products matched at this step represent 69% of product solds and 67% of revenues.

As illustrated with Table 3.1, products with an exact match in Open Food Facts share similar short labels. Thus, we might expect that this similarity between sources also holds for the remainder of the products. This justifies to rely on fuzzy matching based on tokenized text.

This matching gives us, for many products, variation on the same product name. For instance, the antepenultimate row creates a link between `RICARD` and `PASTIS` that can be considered as synonyms. While an approach based on editing distance will fail to account for this synonymy, an embedding that would have been trained to recognize these pairs will help to build a semantic distance that could allow to associate these products.

Therefore, we use this exact matching as the training set of a neural network model to create a word embedding. The latter is used to propose an alternative measure to better identify false positives (high Levensthein distance but no semantic proximity) but also to limit the number of false negatives (low Levensthein distance but synonymous labels).

3.4 Indexation Principle

Search engines, to speed up information retrieval, are generally based on the principle of indexes. Indexes are a way to organise some key features of the corpus in order to find some entries latter on more easily. For instance, at the end of some books, one can rapidly find paragraphs about a concept or a person by looking at the index in alphabetical order.

Modern search engines are a generalization of this approach. Among them, `Elasticsearch`, which is a search engine based on the open-source information retrieval library `Apache Lucene` is well-known for its ability to provide extremely fast text search as well as flexibility in the criteria used to find a responses for a request. One of the most attractive feature of `Elasticsearch` is indeed that it gives the user the possibility to mix string comparison algorithms and give them more or less importance in the final score used to compare a string with a potential match. In particular, `Elasticsearch` allows search based on approximate similarity algorithms presented in Section **??** augmented with additional multiple criteria. After the

preprocessing steps, we ingest nutritional databases in Elasticsearch indexes - and use several built-in Elasticsearch analyzers to query supermarket data and retrieve a fuzzy match candidate with its nutritional characteristics.

`Elasticsearch` is known to be a very efficient search engine.[8] Table 3.2 compares the performance of `Elasticsearch` with two python based librairies to compare the distance between textual fields using a Levensthein distance. `Elasticsearch` is around 100 times faster without special optimization from the user. The efficiency of `Elasticsearch` comes from the server side mostly from parallel queries, optimization from indexation retrival and query analysis… Larger gains would be possible, for instance by using asynchronous search to reduce communications volume between the `Python` client and the `Elastic` server.

Implementation	Speed (sec.)
Mixed implementation	
`ElasticSearch`[a]	0.92
Python	
`Rapidfuzz`[b]	97.20
`Fuzzywuzzy`	99.90

[a] `ElasticSearch` time includes the loss of time of sending data from Python to ElasticSearch server and bringing back data from the best match

[b] `Rapidfuzz` uses, behind the stage, `C` rather than relying on a slow pure `Python` implementation

Table 3.2: Performance to link 100 scanner products data with Open Food Facts

The following box illustrates some of the key features of our fuzzy matching methodology by describing our `Elastic` request.

[8] `Elasticsearch` proposes analyzers that are methods to transform, behind the stage, the request that a user submitted. For example, it is possible to automatically apply stemming at the request level rather than configuring this transformation yourself. We decided to separate the standard cleaning steps (stop words removal, tokenization, etc.) that are performed at the preprocessing level from more refined transformations that are efficiently performed by Elastic (stemming, n-grams)

Remarque

Assume we want to find the energetic value of *"pulco delice agrume"* by finding the closest product within the `Open Food Facts` database.

To be more efficient in our search, we restrict the search to *"Mineral waters, soft drinks, fruit and vegetable juices"* (products that share the same 5 first digits than our COICOP: *"01.2.2.2"*) and even give an extra score if they belong to the same subclass : *"refreshing drinks"* (products that share the same 6 first digits than our COICOP: *"01.2.2.2.1"*).

Four matching criteria are used, among which at least three should provide a non-zero score for a candidate to be considered. Scoring is based on the sum of the four match criteria scores which are computed per candidate product `d` to rank them according to relevance.

The first match criterion is boosted to weight ten times more then the other (`boost = 10`), and is based on a word-by-word tokenizer, e.g. *"pulco agrume"* is encoded as `['pulco','agrume']`. That is, with this tokenizer, "pulco agrume" will never match with the word *"agrumes"*.

However, we introduce fuzziness in two dimensions to gain flexibility when comparing labels:

- The second match is based on a n-gram tokenizer: all words are decomposed into n-grams of 3 and 4 characters, e.g. *"pulco agrume"* is encoded as `['pulc','ulco','pul','ulc','lco','agru','grum','rume','agr',...]` and thus has a non-zero score with *"agrumes"*. We use the a user-defined analyzer `ngr` based on the efficient Elastic tokenizer `ngram` to account for that.
- To avoid being penalized by small differences between corpus, in particular plural vs singular in one source, we also stem the data. We use a french stemming tokenizer e.g. *"pulco agrume"* is encoded as `['pulco','agrum']` and thus has also a non-zero score with *"agrumes"* under this tokenizer.

The fourth match criterion scores the prefix correspondences between 6-digits COICOP code.

For each token-based match criterion, the score is based on TF-IDF measures. These measures take into account that if two products contain a rare word, this word is a valuable information to match the products. More precisely, the score of a document `d` is the sum over query tokens `t_i` of the expression `boost*tf(t_i,d)*idf(t_i).tf(t,d)` is an increasing function of the frequency of token `t` in document `d` (often, a token appears only once) and a decreasing function of the token length of document `d` compared to other documents. `idf(t)` is a decreasing function of the number of documents containing token `t` in the index.

```
{
  "query": {
    "bool": {
      "should": [
        { "match": { "libel_clean_OpenFoodFact":
                      { "query":  "pulco delice agrume" , "boost" : 10}}},
        { "match": { "libel_clean_OpenFoodFact.ngr":   "pulco delice agrume" }},
        { "match": { "libel_clean_OpenFoodFact.stem":   "pulco delice agrume"
        }},
        { "match":  { "COICOP.6digits": "01.2.2.2.1.9999" }}],
      "minimum_should_match": 3,
      "filter": [
        { "match":  { "COICOP.5digits": "01.2.2.2.1.9999" }},
        {"exists" : {
               "field" : "energy_100g"
        }}
      ]
```

```
    }       }       }
```

4 Semantic similarity from a siamese neural network

4.1 Justification

In some situations, it is necessary to extend the definition of fuzziness to take into account the richness of our corpus where the same product can be encoded with different terms by the distinct data providers. Proving useful in many NLP tasks, word embeddings have been shown to be able to represent semantic similarities between individual words. Word embeddings are dense representations of texts in $\mathbb{R}^N$. In this high-dimensional space, words having a similar meaning tend to be close by. Following Mikolov et al. (2013), these representations are now mostly learnt by solving a deep learning problem.

To challenge and improve upon our main approach, we create a short-label embedding specially designed for our problem, by learning this representation from known matched pairs. By defining an euclidean distance in this embedding space, we define a distance which is adapted to our problem and intended to represent semantic proximity. This customized distance currently serves as an alternative validation score to the edit-based Levenshtein distance. To build the embedding space, we train a siamese network on the natural training set obtained with the exact matching between Open Food Facts and supermarket data. Word embedding consists in transforming labels into vectors in $\mathbb{R}^N$. Each dimension of the vector should represent a latent feature in the space of products. For instance, one could imagine that some products will be closed in i^{th} dimension because they share a flavor (for instance chocolate) while others will be close in the j^{th} dimension because they share a packaging characteristic.

4.2 Principle

Siamese networks are deep learning models where a neural network learns to dissociate pairs of twin labels from two random concurrent products. In practice, to take back an example from Table @ref(tab: examples-step1), it means that we train a model to understand that *"RICARD FA18"* and *"PASTIS DE MARSEILLE"* are twins while *"LA TARTIFLETTE AU REBLOCHON"* and *"BEURRE AUX TRUFFES"* should be discarded matches. Siamese networks were first introduced in the early 1990s to solve signature verification as an image matching problem (Bromley et al. 1993), and is used to produce similarity measures for tasks such as face recognition. A Siamese network consists of twin (or triplets, quadruplets..) networks which accept the same input type and produce with the same functional equally many vector outputs.

At network exit, these output representations enter a loss function. This loss function is typically a contrastive loss function, incurring loss with the proximity between distinct input pairs and with distance between similar input pairs. The parameters of the twin networks are tied. Weight tying guarantees that two extremely similar inputs could not possibly be mapped by their respective networks to very different locations in feature space because each network computes the same function. In this sense, this approach preserves syntactic proximity.

We opt for a quadruplet network which is trained with quadruplet inputs: a supermarket entry (anchor), its match entry in Open Food Facts (positive example), two randomly selected entries in Open Food Facts (negative examples). Our network architecture is as follows :

1. A short label is divided into at most its 20 first words which are mapped into a list of at most 20 integers indexing words in a pre-specified vocabulary ;
2. An embedding layer maps each vocabulary index into a vector of dimension `100`. A label is thus a collection of vectors of dimension 100. Given some initial weights, this embedding layer is further improved through our task-specific training step ;
3. A small number of linear layers with Rectified Linear Unit (ReLU) activation.

The loss function writes as a function of three squared Euclidean distances:
- $d_{a,p}$: between the anchor and the positive label (the true match)
- $d_{a,n}$: between the anchor and the first negative label (a false match)
- d_{n_1,n_2}: between both negative labels (a false match)

$$\text{Loss}(a, p, n_1, n_2) = (d_{a,p} - d_{a,n_1} + 2)_+ + (d_{a,p} - d_{n_1,n_2} + 1)_+$$

The loss function becomes zero if the distance to be minimized $d_{a,p}$ is lower with a margin than the other two distances, with even more margin for the contrast between the anchor with the negative match.

4.3 Model training

We train our model on 80% of the pairs derived from EAN linkage and use 10% of our pairs for validation (the final 10% are used to evaluate performance in a test set). Model is trained over 50 epochs.

This layer is initialized with the word embeddings derived from the `Fasttext` model that has been used to label products into our COICOP classification. Intuitively, words which tend to predict the same COICOP class tend to be close in this space. Because trained on similar supermarket short labels for deriving a product class, we expect this first embedding to provide a good starting point.

We train this model with about 100,000 matched pairs. Performance is evaluated on a test set through a matching task. Figure 4.1 represents how training improves the performance of the model. After initialization according to the `Fasttext` model weights, the training after transfer quickly provides a performance gain. After a certain number of epochs, the performance improvement becomes marginal.

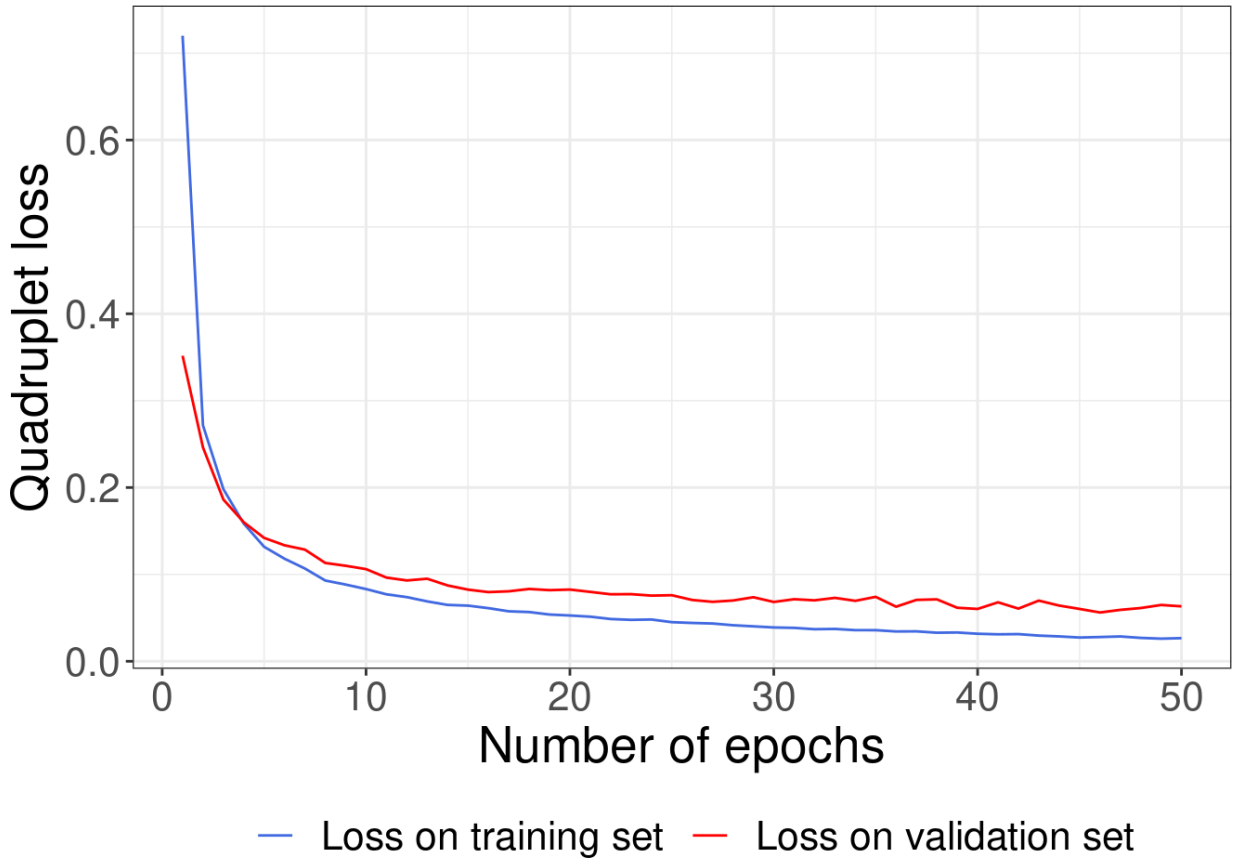

Figure 4.1: Training and validation losses over epochs

We define our siamese similarity as the cosine similarity that is defined as a normalized dot product between the final embeddings of any two short labels:

$$d_{X,Y} = \frac{\mathbf{X} \cdot \mathbf{Y}}{\|\mathbf{X}\|\|\mathbf{Y}\|}$$

We first find that such a distance is well fitted to identify false matches that would arise when letting `ElasticSearch` choose matches by itself. As illustrated in Table 4.1, this appears clearly by inspecting cases where the siamese similarity is lower than 0.1.

RelevanC	Open Food Facts or Ciqual
JAMBON EXCELLENCE MORILLES	LINDT EXCELLENCE POIRE
MINIS BEURRES AROMATISE	VINAIGRE AROMATISE
CKT CITRON A CONGELER	SUCETTE A CONGELER
TOM COEUR PIGEON FR	OEUF PIGEON ACIDULE
CREVET CTE PE BLANCHE ANISEE KG	BIERE BLANCHE

Table 4.1: Echoes from Elastic where siamese similarity is below 0.1

The second objective of building an embedding similarity was to link products that were available under very different names between our data providers. Table 4.2 gives some insights regarding the difference between Levensthein distance and embeddings distance.

Nearest neighbors in embedding space		Distance measures	
Product name in OpenFood Facts	Tokenized name	Cosine similarity	Levensthein similarity
Queried product: CHIPS LAYS			
Chips Bret's classiques	chips brets classiques	98	75
Chips Brets	chips brets	98	75
Chips Bret's Classique	chips brets classique	98	75
Chips Pringles Original	chips pringles original	97	75
Queried product: DRAGIBUS			
dragolo 750g	dragolo	96	73
Pulmoll Classic	pulmoll classic	96	31
Bestfizz	bestfizz	95	36
Nerds orange	nerds orange	95	50
Queried product: CROCODILES HARIBO			
Haribo Haribo Pico-balla 100 g	haribo haribo picoballa	96	60
Haribo Haribo Wummis 200 g	haribo haribo wummis	96	60
Sour-mix Haribo	sourmix haribo	96	73
Lagartos Haribo	lagartos haribo	96	72
Queried product: NUTELLA			
Nocciolata Milchfrei	nocciolata milchfrei	93	29
Nocciolata	nocciolata	93	44
Chocolats artisanales	chocolats artisanales	92	33
Pralines Spécialité	pralines specialite	91	44
Queried product: NACHO			
Lays Natural Classic Flavor	lays natural classic flavor	88	44
Mostaza molida	mostaza molida	88	40
Innocent mangonifico	innocent mangonifico	86	57
Copacabana	copacabana	86	57
Queried product: RICARD			
Anisette Phenix	anisette phenix	87	29
Les Spiritueux	spiritueux	84	40
Tête de Souris	tete souris	84	44
Tête d'Allulette	tete dallulette	84	17

Note:

By construction, when both the query and the echo are identical, both string measures are maximal. These cases have been removed from this table. Best echoes where the queried terms appears have been removed from this table to stress cases where the embedding finds unexpected echoes from an editing distance perspective.

Cosine similarity is multipled by 100. It has been computed with `faiss` library.

Levensthein similarity is based on the normalizing edit distance implemented in `rapidfuzz`.

Table 4.2: Examples to understand siamese network embedding

5 Results

5.1 Siamese model performance

Performance is evaluated on a matching task over the test set, 16,000 pairs of matched RelevanC and Open Food Facts products, left aside during training. Each product label in the test set pairs is projected on our final embedding. We then recover the k nearest neighbors of each embedded label from Relevanc in the embedded labels from Open Food Facts. Performance is evaluated on the ability to retrieve the correct Open Food Facts pair in the top-k nearest neighbors (Top-k accuracy).

The size of the test set affects the number of products in which it is necessary to search to find the correct pair. Thus, Figure 5.1 depicts the performance of the Siamese neural network on a test set for two settings. We represent Top-k accuracy for the matching task performed on the whole test set (about 16,000 products) or on a subset of this sample (2,000 randomly selected products).

As expected, a smaller test set provides better accuracy by reducing the decision set. The siamese model is able to find about 24% of the correct matches as the closest Open Food Fact neighbor in the whole test set. Relaxing the constraint to allow two nearest neighbors brings the accuracy to 32% (Top-2 accuracy). Then, the gains from increasing the set of nearest neighbors are concave. From 8 admissible neighbors, we find the correct match one time out of two. With the smaller test set, a 45 % accuracy is achieved with the Top-1. Searching among 2,000 Open Food Facts product label, the Top-10 accuracy is about 80 %.

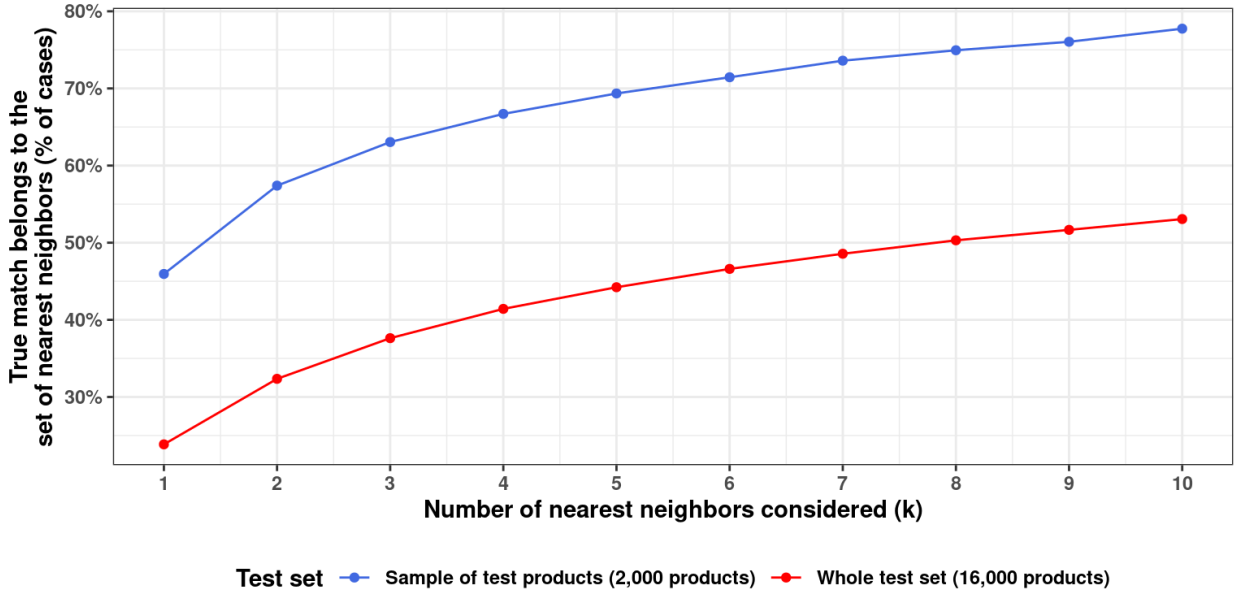

Figure 5.1: Siamese neural network performance on a test set (Top-k accuracy)

Figure 5.2 represents the Top-1, Top-3 and Top-8 accuracy of the matching task among 16,000 products by queried product family. The Siamese model performs best in the most populated product families. Overall, increasing the number of matching candidates from 1 to 8 improves performance similarly across categories. Accuracy remains weaker in the beverage categories.

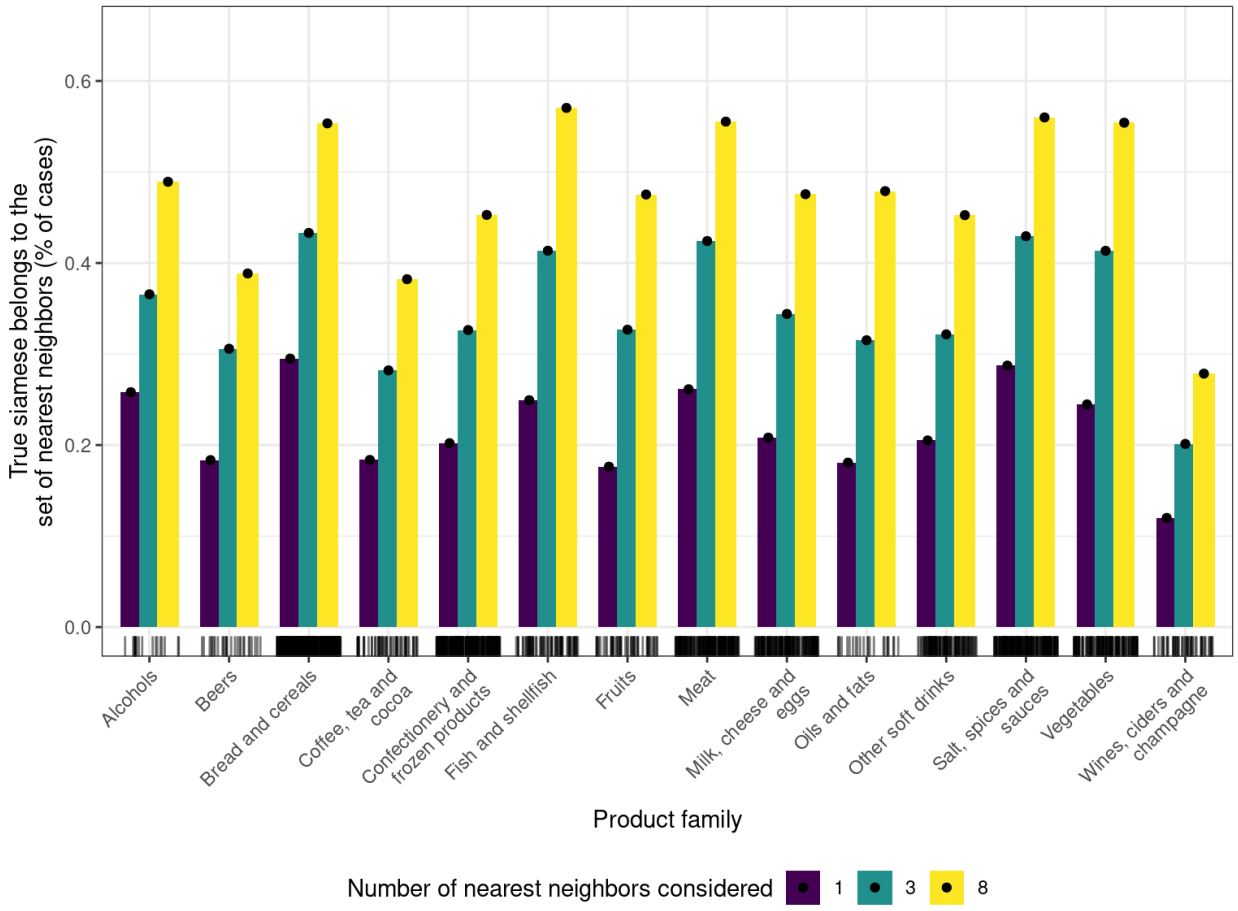

Figure 5.2: Siamese neural network performance on a test set (Top-k accuracy) by product category

5.2 Linkage completeness

After applying our linkage algorithm, we end up with a very high share of products for which we are able to give a nutritional value (Table 5.1). Figure 5.3 shows that this result holds for all nutrients. Whether we are concerned with revenues or sales volume, matching performance exceeds 95% even when the first step (exact matching) was poor (see fibers or energetic value in KiloJoules).

Supermarket chain	No match for the product (% revenues)
Casino	1.36
Franprix	0.92
Monoprix	1.33

Table 5.1: Share of 2019 revenues where our methodology fails to find an energetic value

Figure 5.3 depicts the completeness rate of each category as we move through the matching process. The most notorious nutritional characteristics (calories, sugars, fats, carbohydrates…) share the same profile. The first step (exact matching) allows to enrich products that represent 72% of sales (65% of revenues). The first step of fuzzy matching, where the search is only conducted within common shelves (COICOP classification), a substantial part of the products (% of sales and of revenues). The next step, which consists of abstracting from the shelf, allows only a minor part of the data to be enriched. Finally, the last step, which consists in matching

products from the scanner data with standardized products, allows to cover approximately the same part of supermarket revenues. However, two nutrients have a different profile. These are fibers and calories (in KJ).[9] For these two characteristics, which are much less frequently reported in the Open Food Facts than the others (Figure 2.2), the first-stage linkage rate is naturally worse. Nevertheless, fuzzy matching allows many products to be caught up and thus to achieve a slightly lower completeness rate than for the other characteristics.

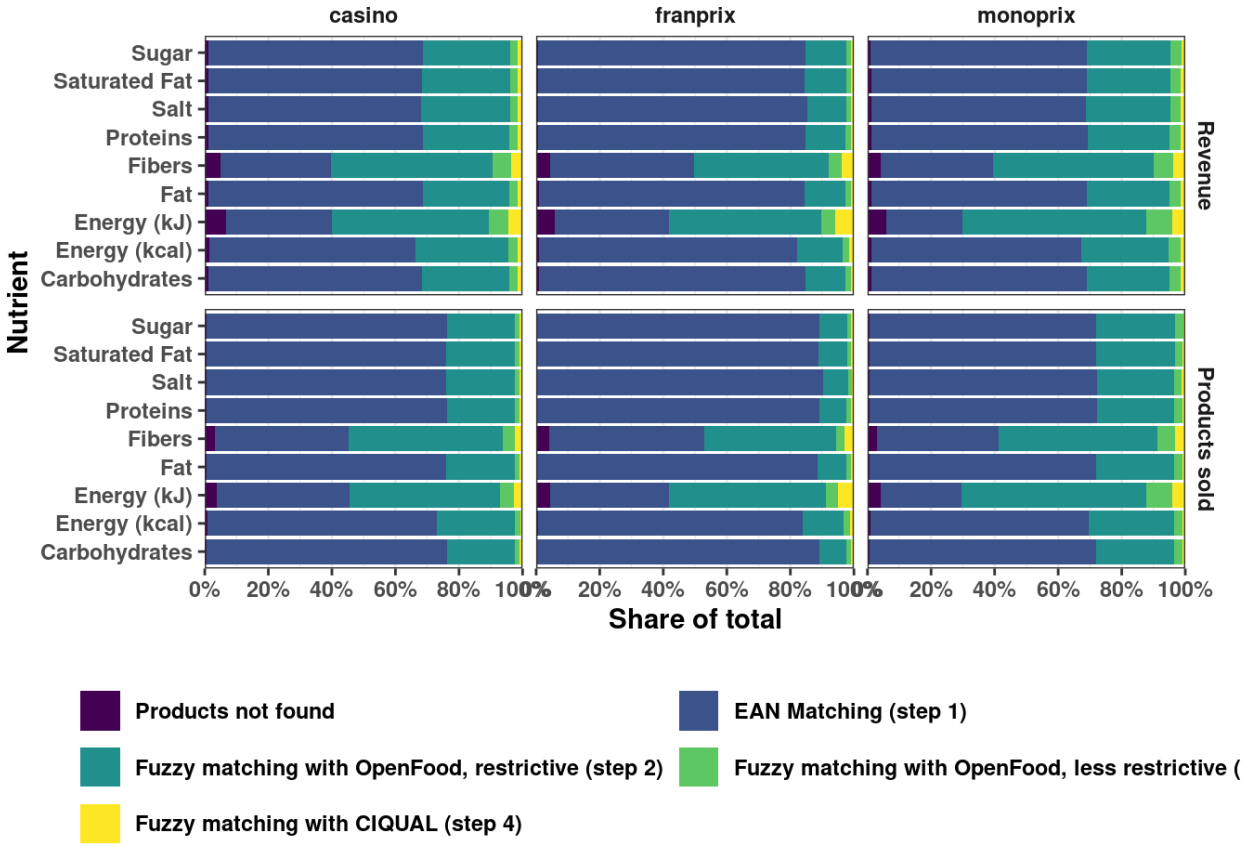

Figure 5.3: Share of revenues covered by each linkage step

Figure 5.3 represents the same statistics for the COICOP groups of food products. There is a strong diversity according to the product class. The classes of mass consumption products, excluding alcohol, are well informed from the first two matching steps. Other relatively standardized products, such as coffee or tea, are well described thanks to the fuzzy matching steps. Matching on the barcode is very uncommon in the groups of alcoholic beverages. The fuzzy matching steps can already greatly improve the completeness rate of this department. In addition, for the latter products, the use of a scrapped dictionary of alcohol brands helps to recover a substantial number of products.

[9] The latter is not, in itself, necessary when we have information in kcal. However, the use of KJ can be used to detect outliers of calories in kcal that have been entered in the wrong unit. It is therefore interesting to keep this information on hand.

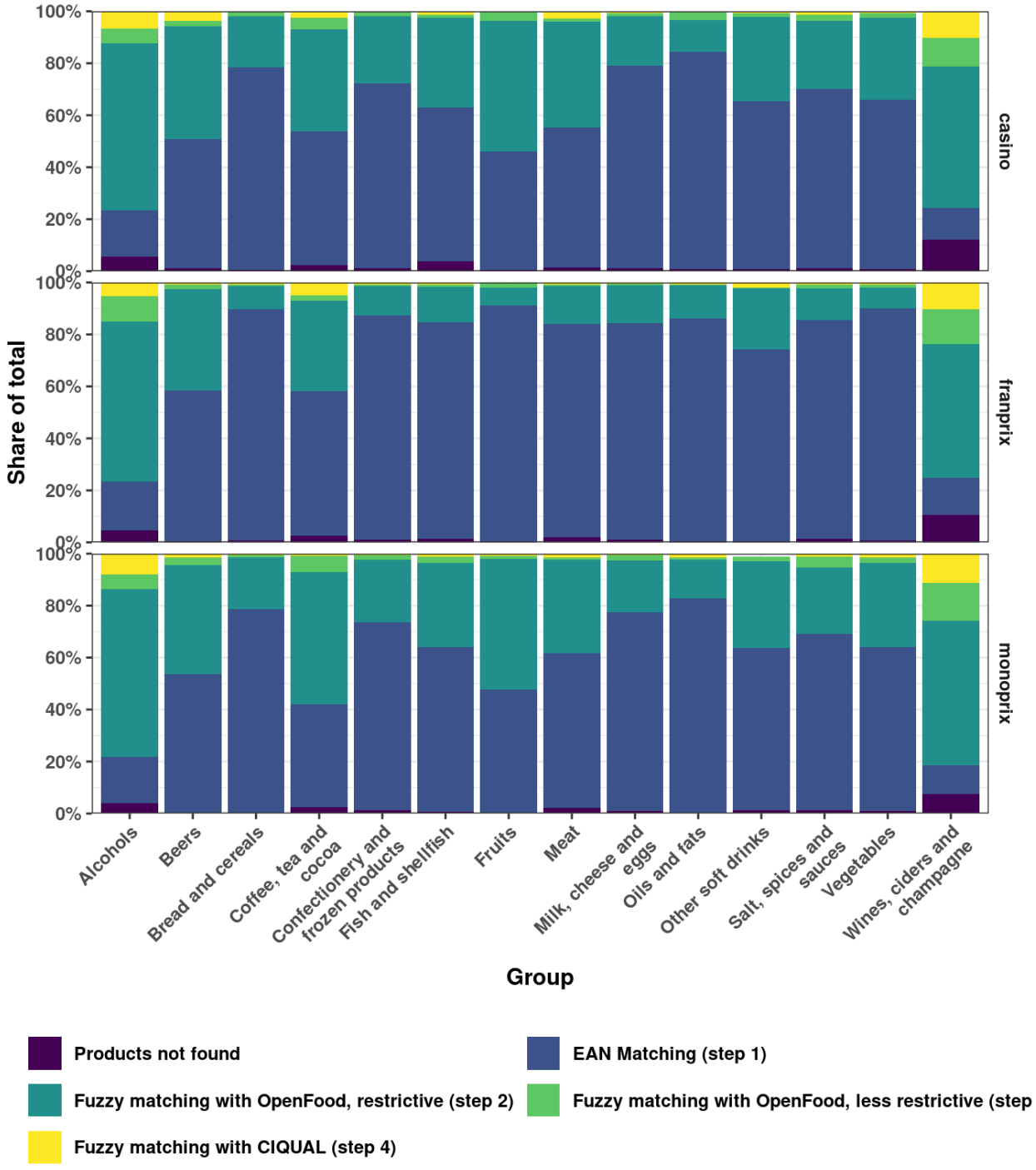

Figure 5.4: Share of revenues covered by each linkage step by COICOP group

5.3 Linkage quality

Table 5.2 shows some random examples of the energetic value derived from the record linkage. In most cases, the combination of products seems satisfactory. Even when the search terms are cryptic, but relatively identifying (such as *"RS"* a short cut for rosé wine), the `ElasticSearch` pipeline manages to find a consistent product. We can also see that sometimes we end up with products having an abnormally high level

of energy (see the row for savage salmon). This is the typical case where the user has recorded the data in the wrong unit, confusing kcal and KJ. However, this situation is quite easy to identify because the energy value is 4 times higher than that of similar products. By comparing the Kcal and KJ fields, we are able to correct most anomalies.

RelevanC label		Open Food Facts label	Nutrients
Original Label	Preprocessed label	Preprocessed label	Energy (by 100g)
HERBE MENTHE POT	herbe menthe pot	menthe bio pot	180
MIEL ROMARIN HAUTE VALLES 480G	miel romarin haute valles	miel romarin	336
POTE AUVERGNATE 400G	pote auvergnate	truffade auvergnate	682
LE PANIER FAISSELLE BIO 4X100G	panier faisselle bio	panier	389
COCA-COLA ZERO PET 1.5LX6 CONT MAST	cocacola zero pet cont mast	cocacola pet	126
NAVARIN AGNEAU 1,2KE	navarin agneau	navarin petit legumes agneau francais	197
ABRICOT 35/45 BQ 1KG	abricot	abricot	84
BISTROT DE FRANCE RS BIB 5L	bistrot france rs bib	rs	1540
SAUMON SAUVAGE PROV.MSC 330G BQ	saumon sauvage prov msc	msc oeufs saumon sauvage	866
ENTREMETS CITRON MERINGUE 6P 500G	entremets citron meringue	cone citron meringue	1142

Table 5.2: Random examples of the values linked between sources

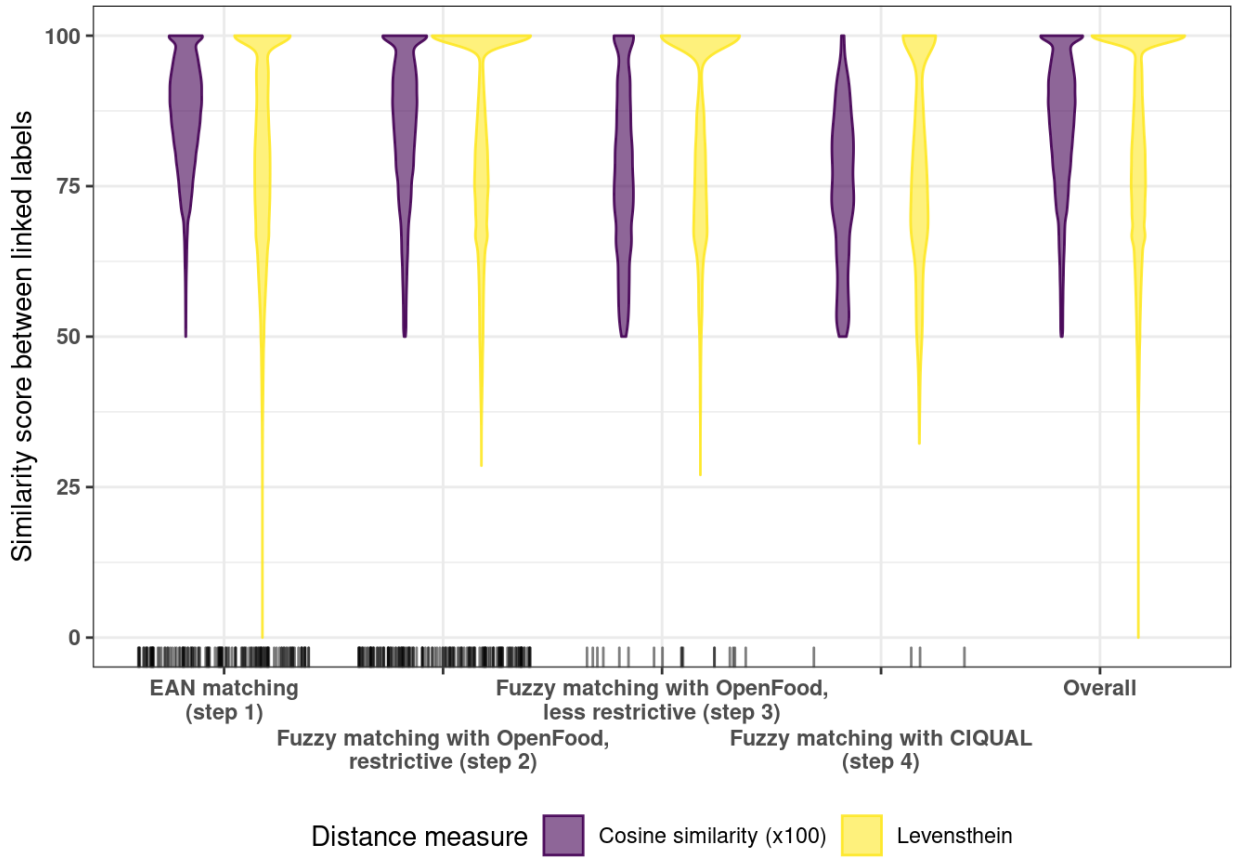

Figure 5.5: Similarity between linked labels by step

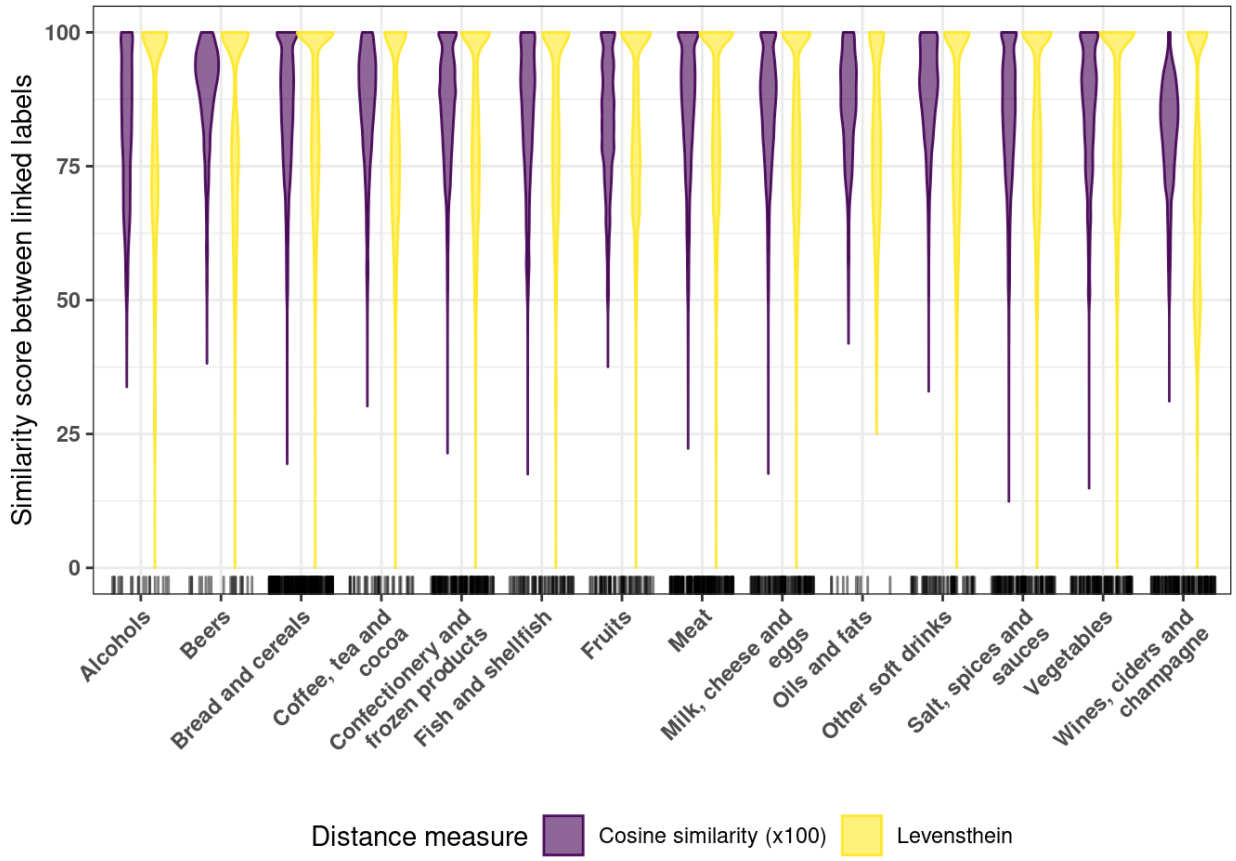

Figure 5.6: Similarity between linked labels by product family

Similarity between the matched product names according to our two distances is represented in Figure 5.5. Overall, the matched names are close. As the matching stages progress (from most certain to most uncertain), the linked labels are less similar. However, even for the last step (where products are matched with standardized labels), the similarity scores are reasonable. For the first two stages, which are quantitatively the most important (this is represented by the density of ticks in Figure 5.5), similarity measures are concentrated in high scores. Using two complementary textual distances to limit the selection of false positives thus results in consistent pairs overall.

A decomposition by product family is shown in Figure 5.6. Again, similarity scores are good. Once again, beverage families stand out with lower performance. However, this may reflect the fact that these products are relatively more matched to standardized products (step 4), whose labels may differ more.

Finally, Table 5.3 gives some examples of mismatches between the results of an exact match and those that would be obtained by fuzzy matching via Elastic Most of the different product names provided by Elastic are honest mistakes. Elastic often provides the same product as the expected true label under a slightly different name. Sometimes Elastic provides a name closer to the original label than the one obtained by EAN matching. In a few cases, Elastic makes mistakes by giving too much weight to some identifying terms. These are often brand names or typical product names. In Table 5.3, for example, as soon as Elastic notices a term “saint jacq”, it tends to match it to scallops, which is a popular dish in France, even when the product is very different.

This comparison also allows us to measure the effect that matching errors could have on the distribution of the nutritional values obtained. Indeed, one can accept not choosing the right product, for example because of redundant products in the nutritional databases, as long as one obtains a product whose nutritional characteristics correspond to those, which are not known, of the product in the scanner data. Linking to a

different product does not necessarily mean an error in the nutritional composition. Overall, the difference in nutritional composition is minimal. The median difference between the calories of products that are obtained by exact matching or fuzzy matching is 100kcal. Table 5.3 again helps to understand where the difference is significant. These are, in the first place, situations where one of the products in the nutritional data proposes an outlier, which is relatively easy to detect. The other typical case of a large difference in composition comes from an obvious matching error. To take the example of terrines and scallops, by making a mistake in the product, it is matched with a product of a different nature, which results in very different nutritional characteristics.

	Match by EAN		Fuzzy matching using ElasticSearch	
Tokenized label in scanner data	Tokenized label in Open Food Facts	Energy (kcal/100g)	Tokenized label in Open Food Facts	Energy (kcal/100g)
st mini kit kat nestle	kitkat mini barre chocolat lait	2152	nestle kit kat singles	2151
figues moelleuses mono	figues turquie	1029	figues moelleuses	1133
terrine breton st jacq	terrine noix saintjacques a bretonne	758	coq st jacq poireaux	498
tic tac ref saisonniere	bonbons tic tac gout gentle orange menthe	1698	tic tac	1046
rill st jacq gp	rillettes noix st jacques a bretonne	610	coq st jacq poireaux	498
pizza chevre lardons co	pizza a pate fine chevrelardons	1039	pizza chevre lardons	1109
tube choco noisettes	pate a tartiner chocolat noisettes	1531	noisettes choco noir	2401
saumon fume atlant ab	saumon fume bio	745	saumon atlantic fume	841
confiture abricot	confiture dabricot	828	confiture abricot	1192
doritos nat	doritos gout nature maxi format	2081	doritos	5313

Table 5.3: Comparison between matches given by fuzzy matching vs exact EAN matching

5.4 Conclusion

By consistently applying some state of the art NLP techniques, we are able to fuzzily match several high dimensional sources. This approach can be used in areas other than product name matching. Combining unstructured sources from textual fields is indeed a recurrent task of administrations or companies that collect data automatically.

The specificity of our corpus is the great diversity of variations of close labels. For instance, two soda packs can be distinguished only by the size of the containers, without any other product difference. Preprocessing harmonizes product names which reduces the noise preventing the identification of two similar products because of unnecessary details (volume, number of products in the pack, etc.).

We rely on some of the most advanced methods and tools in the data-science industry to apply fuzzy matching. The tools used aim to apply some of the best techniques to scale up some of the intuitions expected from a good match. The methodology implemented is a trade-off between flexible matching, which can lead to the selection of false positives, and conservative selection criteria, which can lead to false negatives. The combined use of selection methods based on syntactic (TF-IDF measures, n-grams, Levensthein distance) and semantic criteria (neural network to build an embedding) is a compromise to allow similarities despite different wordings.

At the aggregate level, this approach provides nutritional information for around 98% of the food products in the scanner data. The best matched products are mass market products. This is indeed one of the limits of the neural network which has learned on a dictionary which is certainly varied but nevertheless mainly issued from mass market products.

References

Allcott, Hunt, Rebecca Diamond, Jean-Pierre Dubé, Jessie Handbury, Ilya Rahkovsky, and Molly Schnell. 2019. "Food Deserts and the Causes of Nutritional Inequality."*The Quarterly Journal of Economics* 134 (4): 1793–1844.

Bandy, Lauren, Vyas Adhikari, Susan Jebb, and Mike Rayner. 2019. "The Use of Commercial Food Purchase Data for Public Health Nutrition Research: A Systematic Review."*PLoS One* 14 (1): e0210192.

Barhoumi, Meriam, Anne Jonchery, Philippe Lombardo, Sylvie Le Minez, Thierry Mainaud, Emilie Raynaud, Ariane Pailhé, Anne Solaz, and Catherine Pollak. 2020. "Les inégalités Sociales à l'épreuve de La Crise Sanitaire: Un Bilan Du Premier Confinement." Institut national de la statistique et des études économiques (INSEE).

Bitler, Marianne, and Steven J Haider. 2011. "An Economic View of Food Deserts in the United States."*Journal of Policy Analysis and Management* 30 (1): 153–76.

Bromley, Jane, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. "Signature Verification Using a" Siamese" Time Delay Neural Network."*Advances in Neural Information Processing Systems* 6.

Caillavet, France, Nicole Darmon, Flavie Létoile, and Veronique Nichèle. 2019. "Quatre décennies d'achats Alimentaires: évolutions Des inégalités de Qualité Nutritionnelle En France, 1971-2010."*Economie Et Statistique/Economics and Statistics*, no. 513: 69–89.

Chazelas, Eloi, Mélanie Deschasaux, Bernard Srour, Emmanuelle Kesse-Guyot, Chantal Julia, Benjamin Alles, Nathalie Druesne-Pecollo, et al. 2020. "Food Additives: Distribution and Co-Occurrence in 126,000 Food Products of the French Market."*Scientific Reports* 10 (1): 1–15.

Galiana, Lino, Olivier Meslin, Noémie Courtejoie, and Simon Delage. 2022. "Caractéristiques Socioéconomiques Des Individus Aux Formes Sévères de Covid-19 Au Fil Des Vagues Épidémiques."

Gao, Ya-dong, Mei Ding, Xiang Dong, Jin-jin Zhang, Ahmet Kursat Azkur, Dilek Azkur, Hui Gan, et al. 2021. "Risk Factors for Severe and Critically Ill COVID-19 Patients: A Review."*Allergy* 76 (2): 428–55.

Julia, Chantal, Fabrice Etilé, Serge Hercberg, et al. 2018. "Front-of-Pack Nutri-Score Labelling in France: An Evidence-Based Policy."*Lancet Public Health* 3 (4): e164.

Larochette, Brigitte, and Joan Sanchez-Gonzalez. 2015. "Cinquante Ans de Consommation Alimentaire: Une Croissance Modérée, Mais de Profonds Changements."*Insee Première* 1568: 2008–11.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Efficient Estimation of Word Representations in Vector Space."*arXiv Preprint arXiv:1301.3781*.

Nichèle, Véronique, Élise Andrieu, Christine Boizot-Szantai, France Caillavet, and Nicole Darmon. 2008. "L'évolution Des Achats Alimentaires: 30 Ans d'enquêtes Auprès Des ménages En France."*Cahiers de Nutrition Et de Diététique* 43 (3): 123–30.

Steorts, Rebecca C, Samuel L Ventura, Mauricio Sadinle, and Stephen E Fienberg. 2014. "A Comparison of Blocking Methods for Record Linkage." In *International Conference on Privacy in Statistical Databases*, 253–68. Springer.

Vylomova, Ekaterina, Laura Rimell, Trevor Cohn, and Timothy Baldwin. 2015. "Take and Took, Gaggle and Goose, Book and Read: Evaluating the Utility of Vector Differences for Lexical Relation Learning."*arXiv Preprint arXiv:1509.01692*.

Wang, Dong D., Cindy W. Leung, Yanping Li, Eric L. Ding, Stephanie E. Chiuve, Frank B. Hu, and Walter C. Willett. 2014. "Trends in Dietary Quality Among Adults in the United States, 1999 Through 2010."*JAMA Internal Medicine* 174 (10): 1587–95. https://doi.org/10.1001/jamainternmed.2014.3422.