FOPPA: A database of French Open Public Procurement Award notices

Lucas Potin, Vincent Labatut, Rosa Figueiredo, Christine Largeron, Pierre-Henri Morand

AVIGNON UNIVERSITÉ

FOPPA

A database of French Open Public Procurement Award notices

Lucas Potin[1]
Vincent Labatut[1]
Rosa Figueiredo[1]
Christine Largeron[2]
Pierre-Henri Morand[3]

02/10/2022

[1] Laboratoire Informatique d'Avignon – LIA EA 4128
{prenom.nom}@univ-avignon.fr

[2] Laboratoire Hubert Curien – UMR CNRS 5516
christine.largeron@univ-st-etienne.fr

[3] Laboratoire Biens, Normes, et Contrats - LBNC EA 3788
pierre-henri.morand@univ-avignon.fr

anr©

LABORATOIRE INFORMATIQUE D'AVIGNON

LBNC

Contents

This document presents the FOPPA database, as well as the process we used to initialize it based on European public open data. This work takes place in the context of the DeCoMaP ANR Project[1], which aims at automating the detection of fraud in public procurement.

We first summarize the main notions related to public procurement in Section 1. We then turn to our main data source, the TED, which we describe in Section 2, focusing on the problems that we identified in this database. In the rest of the document, we describe the methods that we propose to solve these problems. We start with two minor issues in Section 3, regarding the separation of agents and criteria. Then, in Section 4, we deal with the major problem, which is about agent identification. Finally, in Section 5, we perform a post-processing aiming at improving the quality of our data. We assessed the effect of each one of these steps upon the database. We conclude in Section 6 by summarizing the characteristics of the FOPPA database, identifying the issues that remain to be solved, and discussing the next steps of our work in the context of DeCoMaP.

1 Public Procurement and Related Notions

Public procurement refers to the purchase of goods, services and works by a public authority (the customer) from a legal entity governed by public or private law (the supplier). In this work, we focus on the case of French public procurement.

In French law, a *public authority* is a public or private buyer, which belongs to one of three possible categories[2]:

- Legal persons governed by public law;
- Legal persons governed by private law, pursuing a mission of public interest and controlled or funded predominantly by public funds;
- Legal persons governed by private law, constituted of public authorities, and aiming at conducting certain collective activities.

This includes, but is not limited to, public governments and state-owned enterprises.

A *public entity* is a specific type of public authority acting as a network operator, i.e. operating in certain particular activity domains related to water or energy networks.

Public procurement must follow a specific set of rules defined by law, and aiming at respecting three principles:

- *Freedom of access*: all potential candidates must be able to access the necessary information;
- *Level playing field*: all candidates must be treated equally by the public authority;
- *Transparency*: the awarding procedure and its outcome must be provided to all candidates.

1.1 Awarding Process

The general steps of the process consisting in awarding a contract to a supplier are as follows[3]:

1. Identification of the client's needs;
2. Breakdown of these needs in several parts;
3. Estimation of the value of each lot;
4. Selection of the most appropriate procedure (see below);
5. Precise specification of the needs taking the form of a public contract;
6. Advertisement for the public contract;

[1] https://anr.fr/Projet-ANR-19-CE38-0004

[2] https://www.economie.gouv.fr/daj/pouvoirs-adjudicateurs-et-entites-adjudicatrices-2019

[3] https://organisme-de-formation-professionnelle.fr/2019/04/08/marche-public-appel-d-offres-definition-deroulement/

7. Selection of the best offer, which is awarded to a supplier;
8. Entering into a contract and conclusion of the process.

At Step 2, the public authority may separate its needs into several parts called *lots*. Each lot is associated to one or several codes expressed using the CPV system (Common Procurement Vocabulary[4]) defined by the European Union. Each such code describes the main or secondary subjects of a contract, e.g. *Fruit seeds*, *Insulation work*.

The *procedure* that must be followed at Step 4 is an important aspect of public procurement. It depends mainly on the value estimated at Step 3, but also on other factors [1]: nature and activity domain of the public authority (state, local government, health institution...), and nature of the contract (goods, services, works).

Under certain very specific conditions, it is possible to use a negotiated procedure without a prior call for competition [1] (*Procédure négociée sans mise en concurrence*). These conditions include the occurrence of an emergency situation, the absence of any reasonably acceptable offer, and the case where the needs are so specific that they can be fulfilled only by a single supplier.

But in the regular case, the factors mentioned above are used to determine a so-called *European Threshold*, that ranges from 139 k€ to 5.35 M€ for the 2020–2021 period. These thresholds are revised every two years by the European Commission. We detail them in Table 10, in the appendix. If the estimated value of the contract is *below* this European threshold, the public authority must follow the so-called *Adapted Procedure* (Section 1.2). If it is *above* this threshold, it must follow the more constraining *Formalized Procedure* (Section 1.3).

The way the contract is advertised at Step 6 completely depends on the selected procedure. Most of the time, it is a call for tenders, that ends after a so-called *acceptance period*. Once the different offers have been studied at Step 7, the public authority decides whether or not to award the different lots to one or more candidates, who are called *winners*. The public authority indicates its choice with a *contract award notice*, which is the formal notice providing the details regarding the contract attribution. The criteria used to select the winner are an important part of the process, which we discuss in Section 1.4.

It is possible for the public authority to *correct* a notice, before the acceptance period of the offers is over[5]. Such a correction can aim at fixing some errors in the original notice, but also at changing the conditions for awarding. If these changes are significant, they must be published as a specific *correction notice*, using the same outlet as the initial notice. Such a correction may result in the extension of the acceptance period.

Finally, it is also possible for the public authority to cancel a contract[6]. This is the case when there is no offer, or no acceptable offer, i.e. the candidates do not meet the needs expressed by the public authority. It is also possible to cancel the contract in case of insufficient competition (too few offers) or for reasons of public interest.

1.2 Adapted Procedure

The adapted procedure, or `MAPA` (*Marché à procédure adaptée* – Adapted procedure market) leaves it up to the public authority to choose the conditions of attribution of the contract, provided the three principles mentioned before are respected (freedom of access, level playing field, transparency). However, additional thresholds control the way the contract must be advertised.

Below 40 k€ It is not compulsory to publicly advertise the procurement or to perform any competitive call.

[4] https://simap.ted.europa.eu/web/simap/cpv

[5] http://www.marchespublicspme.com/avant-la-reponse/lexique-des-termes-de-marches-publics/actualites/2020/12/29/avis-rectificatif-dans-les-marches-publics-qu-est-ce-que-c-est_15704.html

[6] https://www.economie.gouv.fr/files/files/directions_services/daj/marches_publics/conseil_acheteurs/fiches-techniques/mise-en-oeuvre-procedure/abandon-procedure-2019.pdf

Above 40 k€ It is compulsory to publicly advertise the contract. The advertisement medium depends on the estimated value of the contract[7]:

- **Between 40 k€ and 90 k€**: the contract must be advertised by whatever means the customer wants to use.
- **Between 90 k€ and the European Threshold**: the contract must be advertised in the BOAMP (*Bulletin Officiel des Annonces de Marchés Publics* – French official bulletin of public procurement notices).

For the sake of completeness, let us mention that social and specific services have a *specific status* that allows them to use different thresholds [1].

1.3 Formalized Procedure

Above the European Threshold, the public authority must advertise the contract through the BOAMP and the OJEU (Official Journal of the European Union). The online publication outlet of the OJEU is called the TED[8] (*Tenders Electronic Daily*), and we discuss it later in Section 2.1.

The public authority can choose between four types of formalized procedures, and must stick to the selected procedure until a winner has been identified [1].

Open Procedure The public authority publishes a call for tenders. Any interested candidate can submit a bid. This procedure is generally used when the needs are straightforward, the award process is simple, and the public authority expects only a few number of candidates.

Restricted Procedure The public authority also publishes a call for tenders, but only the candidates pre-selected by the public authority can submit a bid. It is two-stepped: first, the potential candidates are asked to express an interest to the contract under the form of a preliminary file; second, the public authority establishes a short list of candidates that are allowed to submit a full bid. This procedure is used for complex contracts and/or when many candidates are expected.

Competitive Dialogue The first step of the restricted procedure is applied iteratively, each candidate being able to revise its bid. The public authority can discard some candidates at each iteration. When the public authority is satisfied, it invites the remaining candidates to submit a full bid. This procedure is used for complex contracts, in particular when the needs cannot be identified clearly in advance.

Competitive Procedure with Negotiation This procedure is similar to competitive dialogue, except the public authority can decide not to negotiate, depending on the nature of the preliminary bids.

1.4 Award Criteria

The public authority has to specify in advance which criteria will be used to select the winning bid. They must respect the following principles[9]:

- Allow selecting the most economically advantageous tender;
- Only apply to the bid, not the candidate itself;
- Be fair and sufficiently precise;
- Be specified before the call for tenders;
- Must be either weighted or prioritized.

The law does not explicitly list all possible criteria, but rather proposes several categories of criteria, and sets some boundaries. Some criteria defined at the national level and used with the adapted procedure are illicit at the European level and cannot be used with the formalized procedure.

[7] https://www.service-public.fr/professionnels-entreprises/vosdroits/F23371
[8] https://ted.europa.eu/
[9] http://www.marche-public.fr/Marches-publics/Definitions/Entrees/Criteres-choix-offres.htm

In case of formalized procedure (cf. Section 1.3, each criterion must be associated to a weight, that allows assessing its importance relative to the other criteria.

It is possible to use a *single* criterion, in which case it is necessarily the *contract value*. However, this is allowed only if the contract aims at buying goods or services that are standardized, and whose quality can vary from one supplier to the other.

Otherwise, the public authority has to use several criteria, which must be related to the object of the contract or its implementation, and must include the contract value. The other possible criteria are organized in the following categories.

Quality The notion of quality covers various aspects of the bid: technical value, aesthetic and functional characteristics, availability, diversity, production and marketing conditions, guarantee of fair remuneration to producers, innovative nature, eco-friendliness, development of direct supply of agricultural products, vocational integration of disadvantaged groups, biodiversity, and animal welfare.

Delivery This includes the following aspects of the bid: delivery times, delivery conditions, customer service, technical support, supply reliability, interoperability, and operational characteristics.

Staff This category focuses on personnel-related aspects of the bid: organization, and professional qualifications and experience.

In addition to these categories, *ad hoc* criteria can be used, but they must be justified by the contract object or delivery conditions.

2 Presentation of the TED

In this section, we first describe the TED, which is our main data source (Section 2.1). We then turn to the data themselves and their structure (Section 2.2), before listing the issues that we detected (Section 2.3).

2.1 Tenders Electronic Daily

As mentioned before, the *Tenders Electronic Daily* (TED) is the online version of the supplement of the OJEU that is dedicated to the publication of the call for tenders and award notices related to public procurement. Consequently, this site hosts documents related to all the public procurement contracts whose estimated cost is above the European Threshold (see Section 1). In addition, it can also host contracts below this threshold, but such publication is not compulsory.

2.1.1 Access and Content

There are two ways to access the content publicly hosted by the TED: by querying the database through an online API[10], or by downloading the data under the form of CSV files. Each such file covers a period of one year. Note that the API provide more information than the CSV files. For now, we use these files only though, as they seem to provide all the information we need in the context of DeCoMaP. These files are not directly stored on the TED, but rather on *data.europa.eu*[11], a website dedicated to hosting the EU open data. It offers two types of notices: contract notices and contract award notices[12].

A *contract notice* (CN) is a document that provides information about an upcoming contract. A *contract award notice* (CAN) provides information on the result of the selection

[10] https://ted.europa.eu/api/v2.0/notices/search
[11] https://data.europa.eu/
[12] https://data.europa.eu/data/datasets/ted-csv?locale=en

process. The latter contains information about the customer (fields starting with `CAE`) and about the supplier (fields starting with `WIN`). This is enough to connect a contract directly to a supplier and a customer, which is why we only focus on contract award notices for now. The TED offers CSV files listing all award notices starting from 2006. However, according to the documentation available on `data.europa.eu` [7], award notices published in the TED between 2006 and 2009 are both less complete and less reliable. This documentation also describes the content of the different fields in the database.

2.1.2 Versions

The format used by the OJEU to represent the contract notices and contract award notices changes through time to fit the evolution of laws and rules. In TED, notices are represented as XML files, whose structure is specified through an XML schema using the XSD dialect. This schema changed over time, in accordance with the modifications underwent by the notice format and structure.

The CSV files available on the `data.europa.eu` website were created in 2016, after the last major change in the notices format, which took place in 2014. Therefore, these data are represented using the most recent format, which is version 2. A specific field `XSD_VERSION` explicitly states the version number of the XML schema associated with each notice or contract notice, according to the CSV used.

The notices available in the TED rely on 5 distinct minor versions of the XML schema:

- Versions 2.0.5, 2.0.6 and 2.0.7: between 2006 and 2009;
- Version 2.0.8: since 2009;
- Version 2.0.9: since 2014.

Most of the notices in the TED use versions 2.0.8 or 2.0.9.

Version 2.0.8 This version is still used for some types of forms, especially those related to the defense and security sector. It is compliant with directive 2009/81/EC[13].

Version 2.0.9 This version is compliant with the directives 2014/23/EU[14], 2014/24/EU[15], and 2014/24/EU[16]. It essentially brings two main changes to the data structure.

First, it adds 14 new variables (the complete list can be found in the *Version* column of the tables provided in Appendix C). Among them, two are mandatory, and particularly important for us:

- `WIN_NATIONALID`: national identification number of the winner.
- `CRIT_PRICE_WEIGHT`: weight associated with the price criterion.

They are important because they allow us to build various forms of networks based on these tabular data. However, due to their late inclusion, they are not filled in notices relying on older versions of the XML schema. As we will see later, working with these notices requires some work to complete the missing information.

Second, some fields previously describing the whole contract have been moved lower, to the level of the single lot. These include the fields `ID_LOT`,`ADDITIONAL_CPV`, `B_VARIANTS`, `B_OPTIONS`, `B_EU_FUNDS`, `DURATION`, `CONTRACT_START`, `CONTRACT_COMPLETION`, which are fields describing the lots. This allows to provide different information for each lot. By comparison, before this change, all the lots had to share the same information.

2.2 Dataset Description

The TED gives access to the award notice of each EU public procurement contract above the European Threshold since 2006, which corresponds to $2{,}318{,}351$ notices and $7{,}589{,}246$

[13] https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1480931705809&uri=CELEX:32009L0081
[14] https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1480931610496&uri=CELEX:32014L0023
[15] https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1480931610496&uri=CELEX:32014L0024
[16] https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1480931610496&uri=CELEX:32014L0025

lots. Data quality was improved in 2009 and the CPV typology was also revised in 2008, which is why we focus on the 2010–2019 period, as it allows us to deal with a stable set of fields. For this period, the TED contains $1{,}980{,}694$ notices, corresponding to $6{,}604{,}742$ lots. In the context of DeCoMaP though, we focus only on the French contracts, amounting to $376{,}650$ notices (20.5%), or $1{,}273{,}002$ lots (21%). In the TED database, the most represented countries are Poland, France and Germany, as shown in Figure 1.

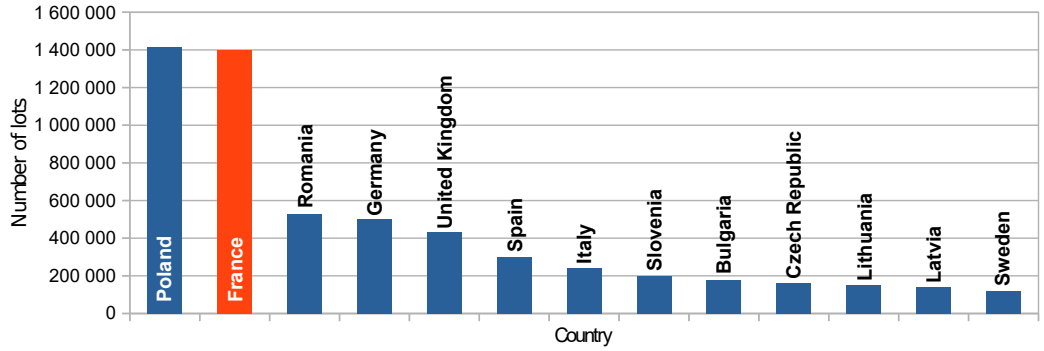


Figure 1. Number of lots published on the TED between 2010 and 2019, for EU member states with more than 100,000 lots.

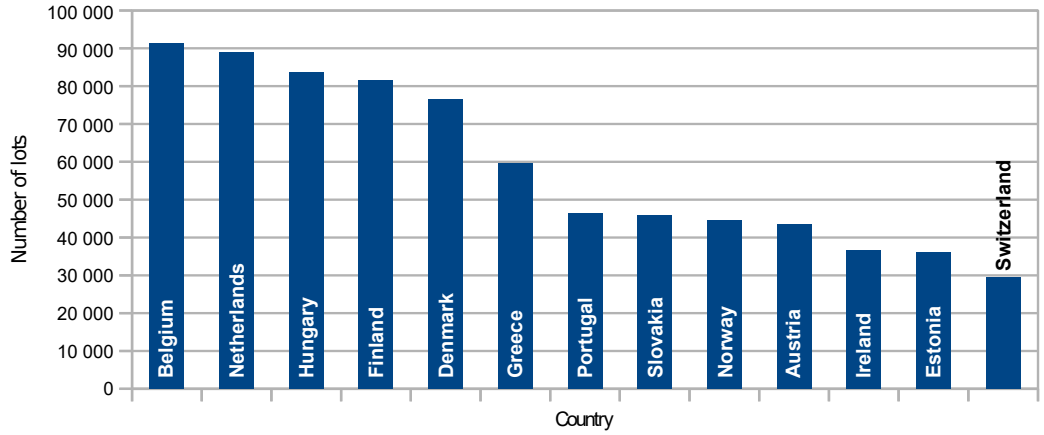


Figure 2. Number of lots published on the TED between 2010 and 2019, for EU members with fewer than 100,000 lots.

The whole TED dataset takes the form of a single logical table. This table is broken down into several CSV files, each one representing a single year. In this table, each row represents a specific *lot*, which is described through 75 distinct fields. We distinguish four categories of fields: *Notice Metadata* (Section 2.2.1); *Agent Information* (Section 2.2.2); *Lot Information* (Section 2.2.3) and *Award Information* (Section 2.2.4). The interested reader will find the complete list in Appendix C. In the rest of this section, we only focus on the fields which are the most relevant to our work, i.e. that can be used to build our network or help characterize fraud in public procurement.

2.2.1 Notice Metadata

This category gathers fields providing general information regarding the award notice. It includes:

- `ID_NOTICE_CAN`: unique identifier of the notice.
- `ID_LOT`: unique identifier of the concerned lot.
- `TED_NOTICE_URL`: URL of the notice on the TED website (page available only during 5 five years after publication).

- `YEAR`: year of publication of the call for tender notice.
- `CANCELLED`: whether the contract was canceled, and therefore not awarded.
- `CORRECTIONS`: number of corrections underwent by the contract after the publication of the call for tender.
- `INFO_ON_NON_AWARD`: if the contract was *not* awarded, indicates the reason why.

An open call for tender can be amended, and a contract can be canceled even after the end of the acceptance period: see Section 1.1 for more details.

2.2.2 Agent Information

The notion of *economical agent* refers to both the client and supplier that enter a contract at the end of the awarding process. In the TED dataset, the client is called CAE, which stands for *Contracting Authority or Entity*, and the candidate which is awarded the contract is called the *winner*.

The fields describing these agents in the dataset are the following:

- `NATIONALID`: unique identifier of the agent, specific to the concerned EU member state.
- `NAME`: name of the agent.
- `ADDRESS`: postal address, composed of the street number, street type and street name.
- `POSTAL_CODE`: zipcode of the agent.
- `TOWN`: city of the agent.

These fields are similar for both clients and suppliers, except that they are prefixed by `CAE_` for the former and `WIN_` for the latter. Public authorities have an additional field `CAE_TYPE` representing their type of public authority.

In the case of French contracts, the national identifier is the SIRET (cf. Section 2.3.6).

2.2.3 Lot Information

Fields from this category provide information regarding the lot sold through the considered contract. They include:

- `CPV`: main common procurement vocabulary code of the lot (cf. Section 1.1).
- `TYPE_OF_CONTRACT`: object of the lot, which can be *works*, *supplies* or *services*.
- `TOP_TYPE`: type of procedure used for the award.
- `CRIT_PRICE_WEIGHT`: importance weight given to the price criterion.
- `CRIT_CRITERIA`: list of criteria, except the price.
- `CRIT_WEIGHTS`: importance weights given to these criteria.
- `B_ON_BEHALF`: whether the contract involves several clients buying together.
- `B_INVOLVES_JOINT_PROCUREMENT`: whether the contract is a joint procurement.
- `B_FRA_AGREEMENT`: whether the contract is within a framework agreement.
- `B_GPA`: whether the contract is under the Government Procurement Agreement (GPA).
- `B_ACCELERATED`: whether the award procedure of the contract was accelerated.
- `OUT_OF_DIRECTIVES`: whether the award notice is published even without a contract notice.

As explained in Section 1.4, the award criteria are aspects of the supplier's bid that are considered by the CAE to select the winner. A contract lot may contain several award criteria, so each criterion has a weight to measure its relative importance. The TED assumes that the price is always a criterion, and consequently has a dedicated field to represent its weight (`CRIT_PRICE_WEIGHT`). The other criteria are represented jointly, using two fields: one lists their names (`CRIT_CRITERIA`) and the other their weights (`CRIT_WEIGHTS`).

2.2.4 Award Information

Fields from this category provide information regarding the awarding process:

- `AWARD_VALUE_EURO_FIN_1`: value of the contract as eventually agreed by the client and supplier.
- `NUMBER_OFFERS`: number of bids received by the client.
- `DT_AWARD`: date of the contract award.
- `B_CONTRACTOR_SME`: whether the contract was awarded to an SME.
- `NUMBER_TENDERS_SME`: number of tenders received from SMEs.
- `B_SUBCONTRACTED`: whether the contract was subcontracted.

2.3 Detected Problems

We have detected several issues in this dataset. This section aims at describing them so that we can propose some appropriate solutions later on.

2.3.1 Missing Award Notices

As explained in Section 2.1, TED contains both contract notices and contract award notices. Usually, a contract notice is followed by one or more contract award notices. Indeed, even in the case of an unsuccessful procedure, CAEs must publish a contract award notice providing explicitly this information.

In the CSV version of the TED, each contract notice has a field `FUTURE_CAN_ID` meant to indicate the unique id of the associated contract award notice. However, this field is not systematically filled in practice. In this case, we do not have access to any information about the results of the awarding procedure, and the winner if there is any. Moreover, some award notices have an `ID_NOTICE_CAN` that is not found in the `FUTURE_CAN_ID` field of any contract notices. They thus represent award notices that are not linked to any contract notice.

On the one hand, we identify $437{,}986$ contract notices between 2010 and 2019, which are connected to $263{,}333$ award notices. Since a contract notice can be associated with one or more award notices, there are still $310{,}770$ contract notices without award notices. These cases may indicate a contract not awarded, or not yet awarded (the data do not contain any award notices after 2020). On the other hand, we are able to link $246{,}421$ award notices to a contract notice, which implies that there remain $130{,}229$ award notices without matching contract notice. Among them, $51{,}030$ have an attribute indicating that the contract has an award notice without a contract notice.

2.3.2 Multiple Agents

As explained in Section 1, a public procurement *lot* can be awarded by several CAEs, and for several suppliers. However, in the TED dataset, each row represents a single lot, and all CAEs and suppliers are indicated jointly in their respective fields. Here is an example of lot involving several CAEs:

```
ID_NOTICE_CAN   CAE_NAME
      2018338   Centre hospitalier d'Arras---Centre hospitalier du Ternois
```

And here is an example of lot involving several suppliers:

```
ID_NOTICE_CAN   WIN_NAME
   2015283576   Montaigne --- BRGC
```

Table 1 shows the different distributions of the lots according to the number of CAEs and winners, for the whole European Union and for France in particular. It appears that even there is only one CAE and one winner in the overwhelming majority of lots, the number of lots with several agents is still significant, and should be handled properly.

In order to take advantage of these data, we need a separate representation of these agents, since we want to connect them afterwards. Therefore, we need to split these values.

Number of agents per lot	Number of lots European Union CAEs	European Union Winners	France Only CAEs	France Only Winners
1	6,085,798	5,960,759	1,376,891	1,336,369
2	112,851	203,657	15,687	42,491
3	20,623	56,128	2,838	12,645
≥ 4	10,746	10,746	6,835	10,746

Table 1. Number of CAEs per lot for French contracts

As illustrated in the previous examples, the standard TED separator between two different agents is a triple hyphen `---`. Ideally, we should always get a string of the form `Agent A --- Agent B`.

However, this is not always the case, and some data entry clerks (or *clerk* for short in the rest of the document) adopt other ways to indicate a multiplicity of entities, using for instance a slash `/`:

```
ID_NOTICE_CAN   WIN_NAME
      2010358   Scape architecture / Treuttel Garcias / Lan architecture
```

2.3.3 Name Inconsistency

As explained in Section 2.2.2, for each lot, TED provides the names of involved agents. However, this field is not normalized, in the sense that one agent can be named using different strings. This is an issue, because this makes agent identification more difficult. We can separate this problem into three sub-problems: inconsistencies in the use of typography, occurrences of different proper nouns to refer to the same agent, and inclusion of irrelevant information in the name field.

Typographic Inconsistency Names, like other string fields in TED, are not normalized: diacritics, and punctuation signs are not used consistently. This makes it impossible to directly perform exact matching between these strings.

```
ID_NOTICE_CAN   CAE_NAME
      2010334   Commune du Grau du Roi

ID_NOTICE_CAN   CAE_NAME
      2010334   Commune du Grau-du-Roi
```

Multiple Proper Nouns Sometimes, an agent can be named using different strings in the TED. A common case is the non-systematic use of acronyms, for instance:

```
ID_NOTICE_CAN   CAE_NAME
      2010334   CEA

ID_NOTICE_CAN   CAE_NAME
      2010334   Commissariat à l'énergie atomique et aux énergies alternatives
```

It is worth noticing that some names combine acronyms and full strings. This particularly the case for education and medical facilities:

```
ID_NOTICE_CAN   CAE_NAME                    Name in SIRENE
      2010332   CH de Belfort Montbéliard   Centre Hospitalier de Belfort
                                            Montbéliard
```

Name Pollution Sometimes, the name field also contains additional information related to the physical location of the agent (ex. building number), or its role in a larger structure (ex. internal department). Here is an example of the latter type:

```
ID_NOTICE_CAN   CAE_NAME
   2013265707   Réseau ferré de France - direction régionale Centre-Limousin
```

This information is not related to the agent's name itself, and makes it harder to perform a proper comparison.

2.3.4 Address Inconsistency

We detected four types of problems with the addresses. First, there is a normalization problem as for the agent's names (cf. Section 3.3.2): typography is not used consistently. Second, the TED confuses several types of addresses (postal, geographic, and geopostal). Third, the fields used to store addresses mix various aspects in an inconsistent way. Fourth, certain address fields sometimes contain irrelevant information.

Typographic Inconsistency Address and town can consequently be filled with hyphens or diacritics:

```
ID_NOTICE_CAN   CAE_TOWN
      2010142   Saint-Julien-en-Genevois

ID_NOTICE_CAN   CAE_TOWN
      2010142   St-Julien-en-Genevois
```

```
ID_NOTICE_CAN   CAE_TOWN
   2013265407   Valence d'Agen

ID_NOTICE_CAN   CAE_TOWN
   2013265407   Valence-d'Agen
```

Type Confusion A database can contain three possible types of addresses. A *geographic* address indicates the physical location using information such as building number, street number, street type, city, and country. A *postal* address is designed for the purpose of mail delivery: it contains only the information used by the postal service for delivery purposes, e.g. zipcode, post office box number. Finally, a geopostal address contains both types of information.

In the TED, all three types of addresses appear. Here is an example of geographical address:

```
ID_NOTICE_CAN   CAE_ADDRESS
   2017373033   13 place Vendôme
```

Here is an example of a postal address:

```
ID_NOTICE_CAN   CAE_ADDRESS
   2017372306   CS 20100
```

Here is an example of a geopostal address:

```
ID_NOTICE_CAN   CAE_ADDRESS
   2017373096   place Maurice Mollard; BP 348
```

The occurrence of all three types makes it difficult to compare addresses within the TED, or even to external sources, because of this lack of consistency.

Monolithic Address The `xxx_ADDRESS` field combines several parts of a geographic address, which are usually considered separately in standard databases: street Number, street type and street name. For instance, in the previous example, the field value is `13 place Vendôme`, which combines all three address parts.

The fact that these parts are combined in the TED makes it difficult to compare its addresses to those coming from other sources, as we will see later.

Address Pollution We call address pollution the presence of irrelevant information in certain address fields, in particular `xxx_TOWN`. For example, for certain agents, the CEDEX code (*Courrier d'Entreprise à Distribution EXceptionnelle* – an accelerated postal service for companies) is specified after the city name:

```
ID_NOTICE_CAN   CAE_TOWN
      2010195   Grenoble Cedex 9
```

Sometimes, the district or locality is indicated in the same field, for instance:

```
ID_NOTICE_CAN   CAE_TOWN
   2017373089   Paris La Défense
```

As for the previous error types, these mistakes make it difficult to compare addresses, both within the TED or from external sources.

2.3.5 Criteria identification

As explained in Section 2.2.3, the TED lists the award criteria and their weights, but the way this information is structured makes it difficult to use, as clerks do not always adopt the same convention to fill the fields. This is an issue for us, because award criteria are likely to constitute a discriminant information in the context of corruption or fraud prediction [2].

First, as for multiple agents, the string that appears to be the standard TED separator to distinguish multiple criteria and weights is the triple hyphen `---`, but it is not used systematically. Data entry clerks alternatively use a slash `/`, a semicolon `;`, and other characters. Here is an example of inappropriate separator for three criteria (technical value, delivery time, and price):

```
 ID_LOT   CRIT_CRITERIA
2013466   VALEUR TECHNIQUE/DELAI DE LIVRAISON/PRIX
```

Second, the price weight is sometimes mixed with the weight of the other criteria, for instance :

```
 ID_LOT   CRIT_CRITERIA             CRIT_WEIGHTS
2010169   PRIX---VALEUR TECHNIQUE   40---60
```

Here, the price criterion is listed together with the technical value, and so are its weight.

Third, the weights associated to the criteria are not normalized: the bounds are not fixed, and they can sum to any value :

```
 ID_LOT   CRIT_CRITERIA                              CRIT_WEIGHTS
2010892   Valeur technique---Prix des prestations   0;6---0;4
```

Here, the weights of the vocational integration and technical value criteria sum to 50. In order to make these values comparable from one lot to the other, we need to normalize them over the whole dataset.

Fourth, sometimes the criterion names and weights are put together in the same field, for instance :

```
    ID_LOT   CRIT_CRITERIA
2014322351   60 % sur la valeur technique---40 % sur le prix des prestations
```

Here, we have three criteria: technical value, price, and delivery time. Their weights are respectively 50, 35 and 15.

Fifth, each TED row should only contain the information related to the considered lot. However, it happens that the criteria of all the lots constituting a given contract are described in the same row, for instance:

```
ID_NOTICE_CAN   CRIT_CRITERIA
      2010227   Evaluation financière (lots 1 - 2 - 8- 9 et 10)
                ---Valeur technique (lot 1- 2 - 8- 9 et 10)
                ---Prestations de service (lot 1- 2 - 8- 9 et 10)
                ---Evaluation financière (lots 3 - 4)
                ---Valeur technique (lots 3 - 4)
                ---Prestations Evaluation financière (lots 1 - 2 - 8- 9 et 10)
                ---Valeur technique (lot 1- 2 - 8- 9 et 10)
                ---Prestations de service (lot 1- 2 - 8- 9 et 10)
                ---Evaluation financière (lots 3 - 4)
                ---Valeur technique (lots 3 - 4)
                ---Prestations de service (lots 3 - 4)
                ---Evaluation financière (lots 5 - 6)
                ---Valeur technique (lots 5 - 6)
                ---Prestations de service (lots 5 - 6)
                ---Evaluation financière (lot 7)
                ---Prestations de service (lot 7)de service (lots 3 - 4)
                ---Evaluation financière (lots 5 - 6)
                ---Valeur technique (lots 5 - 6)
                ---Prestations de service (lots 5 - 6)
                ---Evaluation financière (lot 7)
                ---Prestations de service (lot 7)
```

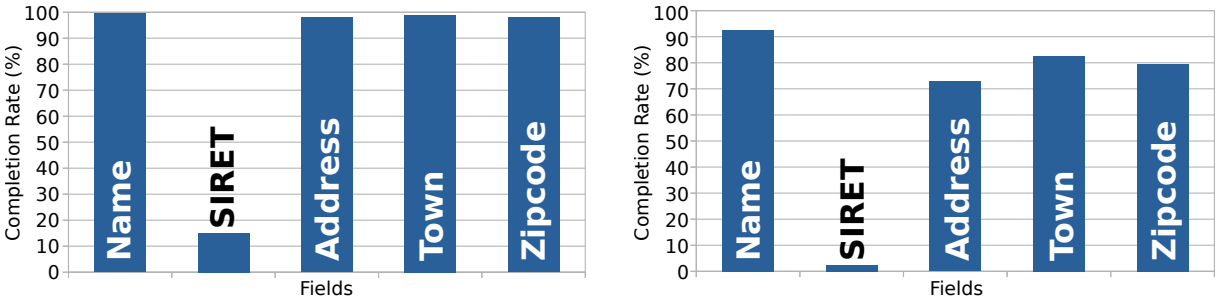
Here, the contract involves two lots, with different criteria. All of them are listed in both rows describing these two lots, instead of only listing the criteria of each concerned lot at once. The first lot uses price as its sole criterion, whereas for the second lot there are two criteria: price and candidate involvement.

In order to use this data, we need to identify which criterion applies to which lot, which requires solving all these issues.

2.3.6 Agent Identification

The TED contains a unique ID to identify each economical agent. This number is national, and can be different for each EU member state. In the case of France, it is the SIRET (*Système Informatique pour le Répertoire des Entreprises sur le Territoire* – Computer system for the national register of companies), which is a 14-digit number representing a specific facility in France (see Section 4.1 for more details about this).

However, in the TED data entry clerks rarely fill the SIRET. Figure 3 shows the completion rate for a selection of TED fields related to economical agents. The y axis shows the proportion of lots for which the field is filled. The figure represents separately the CAEs and the winners. It appears that the SIRET is filled in less than 10% of the lots overall, with an important difference between CAEs and winners: less than 20% for CAEs and less than 5% in winners.



(a) CAE fields. **(b)** Winner fields.

Figure 3. Completion rate for fields related to the identification of economical agents.

This means that we are not sure of the exact identity of the customer or supplier most of the time. Eventually, our objective is to extract various types of graphs from the TED dataset, considering CAEs and winners as nodes. Therefore, it is important to correctly identify all instances of the same economical agent. Otherwise, it is likely that the same agent will be represented as several distinct nodes, which would affect the graph structure.

To solve this problem, we need to leverage the different aspects of the information we have about the agents: name, location, and activity domain. Their location is described in fields address, city and zipcode. However, clerks do not always fill all these fields in public procurement notices, as shown in Figure 3. They are filled most of the time for CAEs, but only approximately 75% of the time for winners. The general better level of completion for CAEs compared to winners could be due to the fact that CAEs complete the contract award notice, and therefore better fill their parts.

The activity domain is more difficult to handle, because in the TED it is not described at the level of agent, but rather at the level of the lot. The `CPV` field (Common Procurement Vocabulary) contains the main CPV code associated with a lot. It gives one of the main characteristic of the contract, and is *always* filled. As mentioned in Section 1.1, each of these codes is defined as a part of a larger typology describing all subjects handled in public procurement. Although this is a lot field, we can still use it to obtain additional information on the winner, assuming that the winner's activity domain is related to this CPV code. However, in France, the activity domain of a company is represented by the `APE` (activité principale exercée). We did not find any correspondence between CPV and APE, so we had to create our own table.

2.4 Overview of the Proposed Method

Our approach to solve the problems identified in Section 2.3 is described in Figure 4. It contains 4 steps that we summarize here, and describe in detail in the rest of this document.

Split As mentioned in Section 2.1, the TED dataset is constituted of a single table broken down into several yearly CSV files, where each row represents an individual lot, with several potential CAEs and winners involved. To ease the data verification and future use, our first step consists in splitting this table into three separate new tables representing the lots, the award criteria appearing in their description, and the concerned agents.

During this step, we solve the *Multiple Agents* (Section 2.3.2) and *Criteria Identification* (Section 2.3.5) problems. We also tackle certain aspects of the *Address Inconsistency*

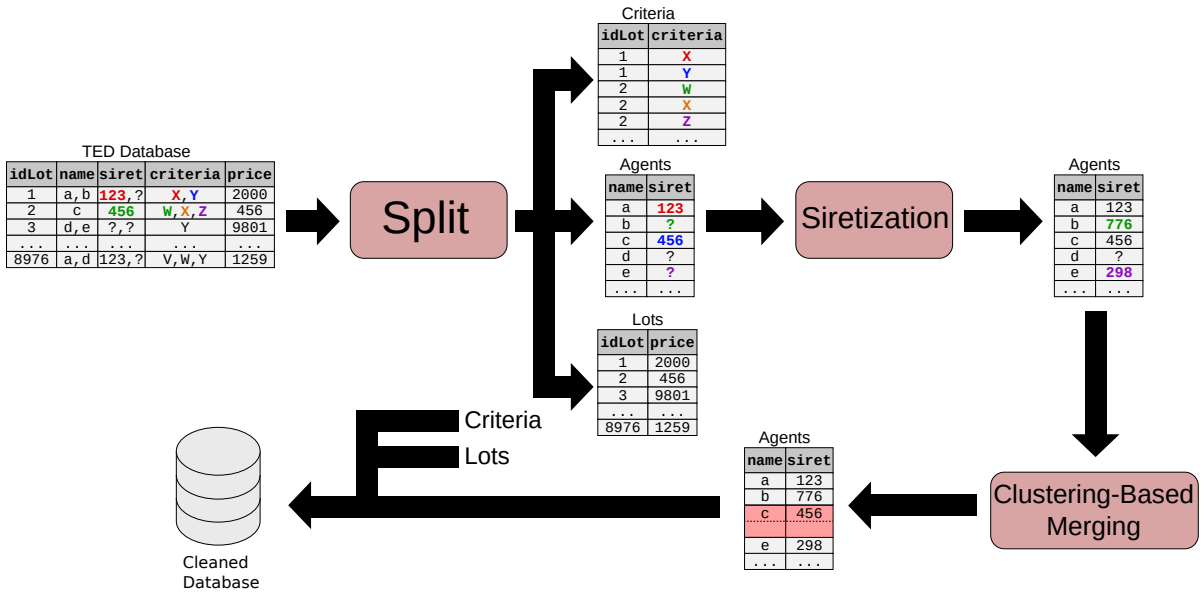


Figure 4. Overview of our proposed method to clean the TED data before graph extraction.

(Section 2.3.4) and *Name Inconsistency* (Section 2.3.3) problems. We describe this step in Section 3.

Siretization In this step, we also tackle the remaining aspects of the *Address Inconsistency* (Section 2.3.4) and *Name Inconsistency* (Section 2.3.3) problems. We also start dealing with the *Agent Identification* problem (Section 2.3.6).

The task that we call *siretization* consists in retrieving the missing SIRETs. For this purpose, we take advantage of SIRENE, an external database maintained by the French state, and listing all existing SIRETs ever. We describe this step in Section 4.

Cluster-based merging Our siretization process is not able to find a reliable SIRET for all agents, because of missing or inaccurate data. The goal of this step is to deal with the remaining cases, therefore finishing solving the *Agent Identification* problem (Section 2.3.6).

We use a fuzzy matching library called `Dedupe`, in order to group similar agent instances thanks to their address and name. Based on this process, we can get two types of clusters. If a cluster contains only SIRET-less agent instances, then we can assume these are different forms of the same entity and merge them. If a cluster contains both siretized and SIRET-less agents, then we can assume that the SIRET-less instances are instances of the siretized agents. We describe this step in Section 5.

Graph-based merging

3 Step 1: Database Initialization

The goal of this first step is to split the single original TED table into several separate tables, in order to ease both the cleaning and usage of the data. In Section 3.1, we describe the structure of our database. Then, we explain how we process the original TED data in order to split them and fill our database. In Section 3.2 we separate multiple criteria and their respective weights. In Section 3.3.1, we focus on addresses and agent names.

3.1 Database Structure

Our FOPPA database contains six tables, as described in Figure 5, and detailed in the rest of this section.

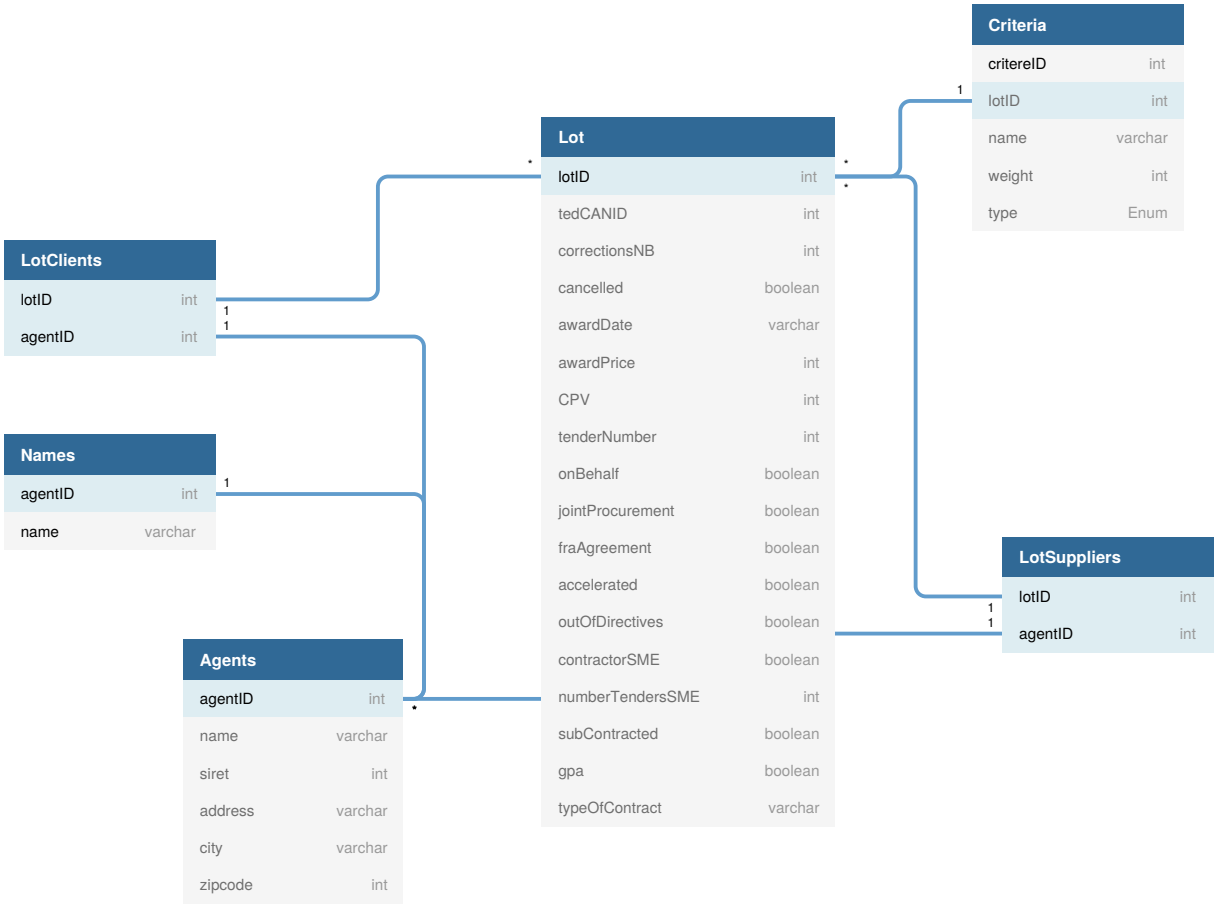


Figure 5. Structure of our database, shown as an Entity-Relation diagram.

Table `Lots` The lots are the central information in our dataset. We represent them in a dedicated table, which contains the following fields:

- `lotID`: unique identifier of the lot.
- `tedCANID`: TED identifier of the contract award notice.
- `cancelled`: boolean indicating whether the lot was cancelled.
- `correctionsNB`: number of notices providing a correction.
- `CPV`: main Common Procurement Vocabulary (CPV) code of the lot.
- `awardPrice`: value of the lot in the award notice.
- `tenderNumber`: number of supplier offers for the lot.
- `awardDate`: award date of the lot.
- `onBehalf`: boolean indicating whether the lot involves several clients buying together.
- `jointProcurement`: boolean indicating whether the lot involved a joint procurement.
- `fraAgreement`: Boolean indicating whether the lot involved a framework agreement.
- `accelerated`: Boolean indicating whether the procedure was accelerated.
- `outOfDirectives`: Boolean indicating whether a CAN was published without CN.
- `contractorSME`: Boolean indicating whether the client is a SME.
- `numberTendersSME`: number of SME offers for the lot.
- `subContracted`: Boolean indicating whether the lot was subcontracted.
- `gpa`: Boolean indicating whether the lot was associated to the Government Procurement Agreement.

- `typeOfContract`: type of the contract, one among
  - `S`: Supplies;
  - `W`: Works;
  - `U`: Utilities.

These fields directly come from the TED fields described in Section 2.2.3.

Table `Criteria` As explained in Section 2, the number of criteria used to award a lot is not predefined, and can range from one to any number. Therefore, there is a many-to-many relationship between lots and criteria. In our base, the concept of criterion is just an enumerated value, though, so there is no need for a specific table to represent the criteria themselves. Instead, we need an association table modeling the association between a lot and its criteria.

In table `Criteria`, each row associates a specific criterion to a specific lot. In addition to the `lotID`, which acts as a foreign key, the table contains 3 fields describing the criterion itself:

- `name`: name of the criterion.
- `weight`: normalized weight of the criterion, relative to all other criteria selected for the concerned lot.
- `type`: type of the criterion, which can take 6 possible values:
  - `PRIX` (price);
  - `DELAI` (deadline);
  - `TECHNIQUE` (technical terms);
  - `ENVIRONNEMENT` (environmental terms);
  - `SOCIAL`Social (social terms);
  - `AUTRES` (others).

Table `Agents` In order to avoid duplicating the information related to economical agents as in the CSV version of the TED dataset, and in order to ensure data consistency, we store agent-related information in a dedicated table. In this table `Agents`, each row represents a single and unique economical agent, which is described using the following fields:

- `agentID`: unique identifier of the agent, *in our database*.
- `name`: principal name of the agent.
- `siret`: SIRET of the agent, i.e. unique identifier of the agent in the TED database (supposedly).
- `address`: full address of the agent.
- `city`: city of the agent.
- `zipcode`: zipcode of the agent.
- `country`: country of the agent.
- `department`: French department of the agent, a code containing 2 or 3 characters.

These fields directly come from the TED fields described in Section 2.2.2s.

Table `Names` An agent can be associated with several names. We create a table in order to keep every name. This table `Names` contains two fields constituting a multiple key:

- `agentID`: identifier of the agent.
- `name`: one of the names of the agent.

Tables `LotClients` **and** `LotSuppliers` As explained in Section 2.2, there can be several economical agents acting as clients and/or as suppliers for a single lot. Therefore, we have a many-to-many relationship here. But unlike with the criteria, this time both agents and lots require dedicated tables to store all their related information. We consequently need two specific association tables to connect each lot to the relevant clients and suppliers.

Each row in table `LotClients` models the involvement of a specific economical agent as client for a specific lot. Table `LotSupplier` has the same role for suppliers. Both tables contain the same fields:

- `lotID`: identifier of the lot.
- `agentID`: identifier of the client or supplier.

3.2 Criteria Processing

The goal of this processing is to fix the *Criteria Identification* problem (Section 2.3.5). For this purpose, we perform the following operations.

Weight Cleaning Using a regex, we parse the text strings present in both TED weight fields (`CRIT_PRICE_WEIGHT` and `CRIT_WEIGHTS`), and remove everything that is not a number or the standard delimiter (`---`). The objective of this process is to remove all superfluous words. Here is an example showing the field before and after this process:

```
ID_NOTICE_CAN   CRIT_WEIGHTS
     20102608   0;45---0;35---0;2
     20102608   045---035---020
```

Criteria Splitting Once the weights are clean, we proceed with the separation of multiple criteria and the computation and/or normalization of their respective weights. There are several possible cases to handle, which we list here from the simplest to the most complicated, in terms of data processing.

If both the criteria name and price weight fields (`CRIT_CRITERIA` and `CRIT_PRICE_WEIGHT`) are empty (which should not happen, legally speaking), or if only the price weight field (`CRIT_PRICE_WEIGHT`) is filled, then there is nothing to do at all.

If both the criteria name and weight fields (`CRIT_CRITERIA` and `CRIT_WEIGHTS`) are filled, we assume that all weights are located in the weight column. We then look for separation patterns, such as the usual triple hyphens (`---`), but also the slash (`/`). Moreover, we check that the number of separators is the same in both fields.

If only the criteria name field (`CRIT_CRITERIA`) is filled, we look for the same separation patterns as before. In this case, we have identified other formattings used when filling this field, but these are too heterogeneous and each one appears very rarely. We consider these cases are not worth the effort.

Here is an example of such a separation, operated on both `CRIT_CRITERIA` and `CRIT_WEIGHTS` fields:

```
ID_NOTICE_CAN   CRIT_CRITERIA                       CRIT_WEIGHTS
      2010142   Prix---Valeur Technique---Delai     40---40---20
>>
ID_NOTICE_CAN   critName                            critWeight
      2010142   Prix                                40
      2010142   Valeur Technique                    40
      2010142   Délai                               20
```

Here is another example, this time when only `CRIT_CRITERIA` is filled:

```
ID_NOTICE_CAN   CRIT_CRITERIA                                CRIT_WEIGHTS
      2010220   Critères Technique (note sur 10) ---         -
                Critères Economiques (note sur 10)
>>
ID_NOTICE_CAN   critName                                     critWeight
      2010220   Critères Technique (note sur 10)           -
      2010220   Critères Economiques (note sur 10)         -
```

Weight Extraction When the `CRIT_CRITERIA` field is the only one filled, cannot get the weights directly, so we extract them using a regex. We then check that the number of weights found is equal to the number of criteria. If it is not the case, we remove the inconsistent weights, for example zeros. Here is an example of weight extraction:

```
ID_NOTICE_CAN   critName                                critWeight
      2010220   Critères Technique (note sur 10)        -
      2010220   Critères Economiques (note sur 10)      -
>>
ID_NOTICE_CAN   critName                                critWeight
      2010220   Critères Technique (note sur 10)        10
      2010220   Critères Economiques (note sur 10)      10
```

Weight Normalization We then normalize the weights in order to get relative values for each criterion. We apply the following formula to the old weights w_i in order to find the new weights w_i':

$$w_i' = \frac{w_i \times 100}{\sum_i w_i}. \tag{1}$$

Thanks to this, the sum of the weights of the criteria for each lot is 100, which makes it possible to make comparisons.

Here is an example of such normalization:

```
ID_NOTICE_CAN   critName                                critWeight
      2010220   Critères Technique (note sur 10)        10
      2010220   Critères Economiques (note sur 10)      10
>>
ID_NOTICE_CAN   critName                                critWeight
      2010220   Critères Technique (note sur 10)        50
      2010220   Critères Economiques (note sur 10)      50
```

If only the price weight field (`CRIT_PRICE_WEIGHT`) is filled, then we insert `Price` as a criterion in our own table, with a weight of 100%.

Criteria Classification The criteria names that appears in the TED are not normalized, which means that they are very heterogeneous. This makes it very difficult to compare contracts. To solve this issue, we define coarser categories of criteria which we store in our database, in addition to the original (free text) criterion names.

These classes are:

- `PRIX` (price);
- `DELAI` (deadline);
- `TECHNIQUE` (technical terms);
- `ENVIRONNEMENT` (environmental terms);
- `SOCIAL`Social (social terms);
- `AUTRES` (others).

We use regex to find keywords, for example `TECHNIQUE` (i.e. *technical*) or `DELAI` (i.e. *delay*), and assign the corresponding class to the cluster.

Here is an example of this process:

```
ID_NOTICE_CAN   critName                                critWeight
      2010220   Critères Technique (note sur 10)        10
      2010220   Critères Economiques (note sur 10)      10
>>
ID_NOTICE_CAN   keyword                                 critType
      2010220   Technique                               TECHNICAL
      2010220   Economiques                             PRICE
```

3.3 Agent Processing

We apply several distinct processes to agent-related data, in order to populate tables `Agents`, `LotClients` and `LotSuppliers`. We describe how we handle location information in Section 3.3.1 and agent names in Section 3.3.2.

As mentioned in Section 2.3.6, the agent SIRET, which should constitute its unique ID for the French data, is generally not filled in the TED, in practice. For this reason, we define our own ID. This requires a specific processing aiming at merging occurrences of the same agent appearing under different surface forms, which is described in Section 3.3.3.

3.3.1 Location Information

The goal of the operations described in this section is to solve the problems identified in Section 2.3.4.

ZipCode and City Normalization As explained in Section 2.3.4, certain fields contain irrelevant information (what we call *Address Pollution*). To solve this issue, we first remove the following information from the city field:

- `CEDEX`, `SP` and `CS`, which is postal information and should not be in this field;
- digits;
- punctuation.

We use regex (regular expressions) to perform this task. Here is an example of the same field before and after this process:

```
ID_NOTICE_CAN   CAE_TOWN
     20113493   MARSEILLE CEDEX 9
     20113493   MARSEILLE
```

During this step, we also partly deal with the *Typographic Inconsistency* problem identified in Section 2.3.4 (inconsistent use of hyphens and diacritics).

Second, we perform a similar task on the zipcode, by removing every non-digit character.

Third and finally, we deal with entries possessing a city name but no zipcode. We leverage a public database called `Hexaposte`[17], which contains the zipcode of each city in France. We use it to retrieve the missing zipcodes. Here is an example:

```
ID_NOTICE_CAN   CAE_TOWN   WIN_POSTAL_CODE
      2010238   PARIS      -
>>    2010238   PARIS      75000
```

Address Normalization Next, we finish dealing with the issues from Section 2.3.4 by normalizing the agents' addresses. First, we remove the different punctuation marks by using a regex, and turn everything to upper case. We also remove all extra spaces. This finishes solving the *Typographic Inconsistency* problem.

Then, we turn to the *Type Confusion* problem (the TED confuses geographic and postal addresses). We remove some words (`CEDEX`, `CS`, `bis`, etc.), especially related to postal addresses, which are not needed or useful in the rest of the process. Here is an example of such a deletion:

```
ID_NOTICE_CAN   CAE_ADDRESS
      2010211   1 PLACE ROBBERT GALLEY BP 9
>>    2010211   1 PLACE ROBBERT GALLEY
```

[17] https://www.data.gouv.fr/en/datasets/base-officielle-des-codes-postaux/

Finally, in the TED, certain addresses extend over several street numbers, which makes later comparisons more difficult and likely to results in mismatches. Therefore, we use a regex to keep only one street number per address when populating our database. Here is an example of address with multiple street numbers, before and after this processing:

```
ID_NOTICE_CAN   CAE_ADDRESS
      2010869   29-31 COURS DE LA LIBERTE
>>    2010869   29 COURS DE LA LIBERTE
```

At this stage, the *Monolithic Address* problem (constituting elements of the address all forced into the same field) is still open. We solve it later when matching the TED addresses to the SIRENE ones (Section 4.2.3).

3.3.2 Agent Names

The goal of the operations described in this section is to solve the *Multiple Agents* problem identified in Section 2.3.2 and the *Typographic Inconsistency* problem from Section 2.3.3.

Name Normalization This process concerns the agent names and aims at solving the *Typographic Inconsistency* problem from Section 2.3.3. It involves several steps. First, we remove the different punctuation marks by using a regex and turn everything to upper case. We also remove all extra spaces.

Second, we delete all the information between parentheses, which is generally irrelevant. Here is an example of a deletion:

```
ID_NOTICE_CAN   CAE_NAME
     20102390   AGENCE NATIONALE DES FREQUENCES (ANFR)
>>   20102390   AGENCE NATIONALE DES FREQUENCES
```

Multiple Name Splitting The goal of this process is to solve the *Multiple Agents* problem from Section 2.3.2, i.e. to separate several agent names involved as clients and/or suppliers in the same lot, but expressed as a single string in each concerned field: name, address, zipcode, city, SIRET. Solving this issue requires extracting the appropriate information from each field.

For this purpose, we first leverage the official delimiter, which is the triple hyphen (`---`). When this delimiter is used in the name field, it also appears in the other fields (address, zipcode, city, and possibly SIRET). It can therefore be used to split each field and retrieve the appropriate information for each concerned agent.

Second, we consider an alternate delimiter: the slash (`/`). However, we only look for the slash in winner names. Indeed, in CAEs, there are cases where a slash in the name indicates additional information and not a new agent, such as here:

```
ID_NOTICE_CAN   CAE_NAME
    201480448   CEA/Grenoble
```

When the slash is used in the winner's name, the other fields (address, zipcode, city, SIRET) are generally incomplete. Typically, only the first agent is properly described. In this case, the only thing we can do is assign this information to this agent in our database, and leave these fields blank for the other agents.

3.3.3 Agent Merging

This section aims at sketching how we solve the *Multiple Proper Nouns* and *Name Pollution* problems identified in Section 2.3.3. Our method relies on a temporary table containing

multiple forms of the same agents, that we reduce to our final `Agents` table through iterative merging.

Temporary tables In the TED, the same agent is likely to appear under various *forms*. We want to merge distinct forms corresponding to the same agent, in order to get a unique representation of each agent. For this purpose, we use two temporary data tables.

The first is `AgentsTemp`, which initially contains all the data describing the agents.

- `idAgentBase`, unique identifier of each entity entry in the TED.
- `nameAgent`: principal name of the agent.
- `siretAgent` : SIRET of the agent.
- `addressAgent` : address of the agent.
- `cityAgent` : city of the agent.
- `zipcodeAgent`: zipcode of the agent.
- `sameAgent`: identifiers of the other forms of this agent.

It has the same fields as `Agents`, except the primary key, which has a different name (`idAgentBase` instead of `idAgent`), and the additional field `sameAgent`, which connects the various forms of the same agent.

During our merging process, this temporary table is gradually reduced, with fewer and fewer entries in the database, but more complete `sameAgent` fields. At the end of the process, each agent should have a single form in our table, which is then copied in the `Agent` table.

The second temporary table, `AssociationsTemp`, models the association between each lot and the involved clients and suppliers in `AgentsTemp`. Its purpose is to later fill the tables `LotClients` and `LotSuppliers` with the appropriate agent ids. This table contains the following fields:

- `lotID`: identifier of the lot.
- `idAgentBase`: the identifier of the entity entry in the TED.
- `Type`: The type of the agent, which can be CAE or WIN.

Merging Process During our process, we group agent forms based on their SIRET or cluster. We explain later, in Sections 4 and 5, exactly how these SIRET and clusters are obtained. For now, we describe the generic part of this processing.

When several forms are grouped together, we keep the most probable name, which is the one that is most frequent among these forms. The other names are stored in the `Names` table. For the address, if we find a SIRET, we keep the address found in SIRENE. Otherwise, we apply the previous method for each field.

Here is an example of merging several forms of the same agent:

```
WIN_NAME                          WIN_NATIONALID      WIN_ADDRESS
Eiffage
Eiffage Energie Thermie EST       34002322500055      1 rue Mendes France
Eiffage Energie Thermie Est       34002322500055      1 rue Mendes France
Eiffage Energie Thermie Grand Est 34002322500055      1 rue Mendes France
Eiffage Energies                  34002322500055      1 rue Mendes France
Eiffage Thermie                   34002322500055      1 rue Mendes France
Eiffage Thermie EST               34002322500055      1 rue Mendes France
Eiffage Thermie Est               34002322500055      1 rue Mendes France
Eiffage Thermie Est SAS           34002322500055      1 rue Mendes France
Eiffage energie thermie Grand Est 34002322500055      1 rue Mendes France
Eiffage thermie                   34002322500055      1 rue Mendes France
Eiffage énergie                   34002322500055      1 rue Mendes France

WIN_NAME                          WIN_NATIONALID      WIN_ADDRESS
EIFFAGE THERMIE EST               34002322500055      1 RUE MENDES FRANCE
```

4 Step 2: Siretization

As explained in Section 2.3.6, the SIRET is used as a unique ID to uniquely identify economical agents in the French TED dataset. However, this information is missing in most entries. In this step, our objective is to identify as many agents as possible, and fill these missing values. In order to fulfill this task, we need an external data source, since the information of interest is missing from the TED. We use the SIRENE database, which is maintained by the French state. It lists all French companies, and describes them using a variety of fields. We introduce this important tool in Section 4.1.

To retrieve the SIRET from SIRENE, we use the individual information available in the dataset, i.e. the name and the address of the agents. But, as explained is Section 2.3, these fields themselves are not always filled: sometimes there is just the name, sometimes the city is present too, and sometimes the full address, which is composed of the street number, street type, and street name. To solve our issue, we propose several processing steps, that we describe in Section 4.2. Finally, we use a part of our data to assess the performance of our method in Section 4.3.

4.1 SIRENE Database

The SIRENE database[18] (*Système National d'Identification et du Répertoire des Entreprises et de leurs Etablissements* – National identification system for commercial entities and their facilities) lists all economical agents participating in public procurement, in France. The database was created in 1973[19], but the use of SIRETs became compulsory only in 1997[20]. SIRENE is a large base, containing about 28 million entries. It covers each year since 1973, and includes not only agents that are currently active, but also agents that are no longer active. It is publicly available online since 2017[21].

In this section, we first discuss a specificity of SIRENE: it distinguishes between two levels of economical agents (Section 4.1.1). We then describe the structure of this database (Section 4.1.2). We conclude with a presentation of the processing we applied to its data, in order to make them suitable to our needs (Section 4.1.3).

4.1.1 Entities vs. Facilities

It is important to stress that SIRENE distinguishes two levels of economical agents: entities vs. facilities. *Entities* (or *Unités*, i.e. units, in the SIRENE terminology) are companies, government agencies, department, charity, institutions (legal entity) or people (natural person) that have a legal existence and the ability to enter into agreements or contracts. *Facilities* (or *Établissements* in the SIRENE terminology) are geographically located units where all or part of the entity economic activity is carried out. Agents from the TED correspond to facilities: we want to identify their SIRETs.

Each entity is identified through a unique 9-digit number called the SIREN (*Système d'Identification du Répertoire des Entreprises* – Identification system of the entity register), whereas for a facility it is a 14-digit number called the SIRET (*Système d'Identification du Répertoire des Etablissements* – Identification system of the facility register). The first 9 digits of the SIRET correspond to the SIREN of the associated entity, while the last 5 digits are called the NIC (*Numéro Interne de Classement* – Internal classification number) and are specific to each facility. Two facilities linked to the same entity share the same SIREN, but have a

[18] https://www.sirene.fr/
[19] https://www.legifrance.gouv.fr/loda/id/LEGITEXT000006062081/
[20] https://www.legifrance.gouv.fr/loda/id/JORFTEXT000000201066/
[21] https://www.sirene.fr/sirene/public/static/open-data

different NIC, and therefore a different SIRETs. If an entity closes a facility and reopens it later at the same location, it gets a different NIC.

Here is an example of two facilities related to the same entity:

```
CAE_NAME                CAE_TOWN            SIRET
ELECTRICITE DE FRANCE   DIJON       55208131788047
ELECTRICITE DE FRANCE   CRETEIL    55208131788054
```

Before 2005, the prefix of the SIREN number of a *public sector* entity was significant: the first two digits represented its legal category, and the following two characters represented the department code of its head office, for entities with a territorial competence. Here, the notion of *department* refers to a French administrative subdivision, corresponding to the NUTS3 level in the European typology. Table 2 lists these codes and their meaning. The *Legal Category* column refers to a code defined by the INSEE[22], the French institute for statistics, and identifying precisely the type of entity. It is available in the SIRENE database.

SIREN Prefix	Legal Category	Description
10/11	7111–7113	State Administration
12	7120	Central department of a ministry
16	7160	Decentralized department of a ministry
17	7171–7179	Decentralized department of a Region
18	7381–7490	Other public institution
19	7383-7384	Scientific institution or college
21	7210,7312–7314	Municipality
22	7220–7229	Department
23	7230	Region
24	7341–7349	Community of Communes
25	7351–7356	Intercommunal household
26	7361–7366	CCAS and hospitals
27	7371	Public housing office
28	7372–7379	Public administrative establishment

Table 2. Categories of public entities distinguished in SIREN ids.

However, since November 2005, only the first two SIREN digits are significant, in two cases:

- Code 13: state administration and agencies with *national* competence.
- Code 20: entities with *territorial* competence.

4.1.2 Structure of the Database

SIRENE is accessible via 3 methods: first, a dedicated website allows a human access; second, it can be accessed programmatically through an online API; and third, it is possible to download the database as CSV files, in order to use them locally. Like for the TED, we adopt the last method, because it allows us to have more control over the way economical agents are searched in these data.

The CSV version of the database consists of four parts:

- `StockUniteLegale_utf8.csv`: a CSV file containing all the entities (unités) (22M entries), be them open or closed, with the latest available information, including the SIREN, usual denomination and acronym.
- `StockEtablissement_utf8.csv`: a CSV file containing all facilities (établissements) (28M entries), be them open or closed, with the latest available information, including the SIRET, zipcode, city, address and trading name. It should be noted that in SIRENE, an address is represented by 3 fields :

[22] https://www.insee.fr/en/accueil

  - `typeVoieEtablissement`: street number.
  - `numeroVoieEtablissement`: type of road.
  - `libelleVoieEtablissement`: street name.
- `StockEtablissementHistorique_utf8.csv`: a CSV file containing the historical modifications of the facilities, including their opening and closing dates.
- `StockUniteHistorique_utf8.csv`: a CSV file containing the historical modifications of the entities, with the previous names of each entity.

Using these four files, we create a temporary database containing four tables, as described in Figure 6. Its goal is only to cross-reference the agents from the TED with the facilities from SIRENE, in order to fill the missing SIRETs. It is not meant to stay in our database after the completion of this task.

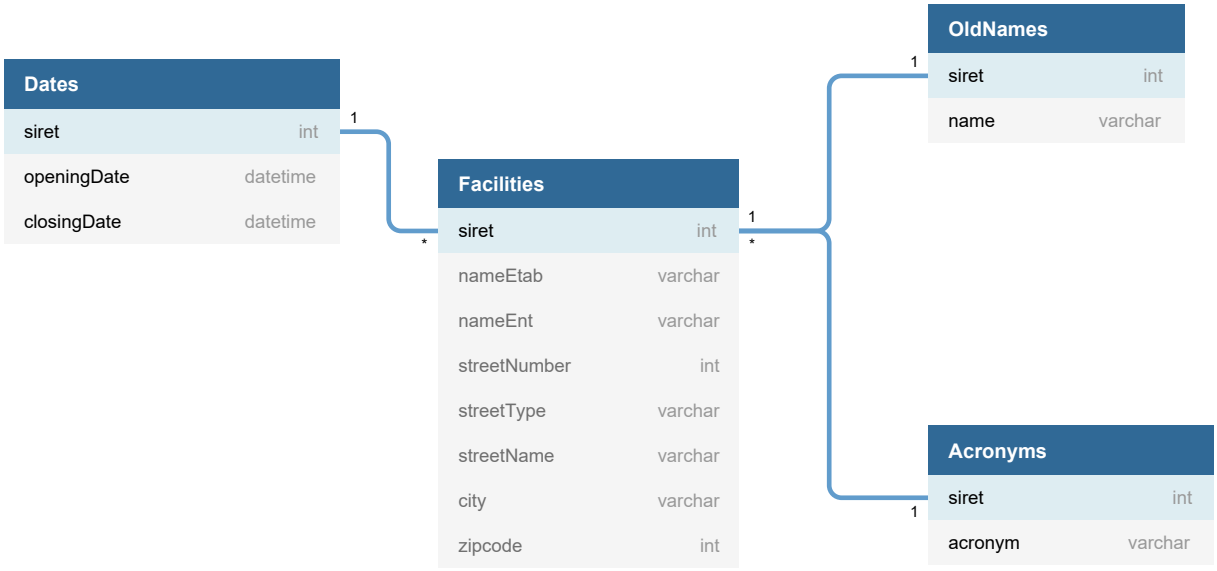


Figure 6. Structure of the SIRENE database, shown as an Entity-Relation diagram.

Table `Facilities` It contains the facilities (établissements), described using the following fields:

- `siret`: SIRET, i.e. unique ID of the facility.
- `nameEtab`: name of the facility.
- `nameEnt`: name of the associated entity.
- `streetNumber`: street number.
- `streetType`: type of the street.
- `streetName`: name of the street.
- `town`: city.
- `zipcode`: zipcode.

Table `Dates` One given facility can open and close several times during its existence. For this reason, it is not possible to store opening and closing dates directly in the Facilities table: we use the `Dates` table for this purpose. It contains the following fields:

- `siret`: SIRET of a facility.
- `openingDate`: opening date of this facility.
- `closingDate`: closing date of this facility.

The closing date is missing when the facility is still open.

Table `OldNames` An entity can change its name during its existence. File **StockUniteLegale_utf8.csv** only provides the latest name, whereas file `StockUniteHistorique_utf8.csv` contains the older ones.

We proceed similarly in our database: table `Facilities` contains the latest name (field `nameEnt`), whereas the previous ones are stored in table `OldNames`. The latter contains the following fields:

- `siret` SIRET of a facility.
- `name`: one of the previous names of this facility.

Table `Acronyms` Some entity names take the form of acronyms. In order to leverage them later when comparing agent names, we gather all these specific names in a different table called `Acronyms`, and containing the following fields:

- `siret`: SIRET of a facility.
- `acronym`: acronym of this facility.

4.1.3 Preparation of the Data

Overall, the SIRENE data appear to be of good quality, and does not require much preparation. The only issue that we detected concerns the names of its entities and facilities. Indeed, as mentioned before, we want to cross-reference the SIRET-less agents from the TED with the facilities present in SIRENE based on their names and addresses. However, the facility names are not always filled in SIRENE, and when they are, they are not always the most appropriate field for our task, as SIRENE may contain several names.

For each entity in SIRENE, one name is possibly stored in the following fields:

- `denominationUniteLegale`: name in case of legal person.
- `denominationUsuelleUniteLegale`: name commonly used by the public.
- `nomUniteLegale`: name in case of a natural person.
- `prenomUniteLegale`: first name in case of a natural person.
- `sigleUniteLegale`: acronym of the facility name.

For each facility in SIRENE, we have a single name stored in the following field:

- `enseigneEtablissement`: name of the facility.

Moreover, as mentioned before, an entity may change its name over time. For instance:

```
    SIREN   UNITE NAME
247400161   SIVOM MORILLON SAMOENS SIXT VERCHAIX
247400161   SIVOM EAU ASSAINISSEMENT MOR/SAM/VER/SIX
247400161   SI DES MONTAGNES DU GRIFFE
```

To create the table `Facility`, we link the facilities in SIRENE to their entities, in order to retain for each facility both the company name and the facility name. We extract opening dates, acronyms and older names of the historical CSV to create the 3 other tables.

4.2 Matching Algorithm

In this section, we describe our proposed method to match a SIRET-less economical agent from the TED to a facility from SIRENE, and therefore obtain the agent's SIRET. As mentioned before, we dub this operation *siretization*.

Our algorithm is described in Figure 7. Each green block represents a subset of facilities from the SIRENE database. The first subset is initialized by selecting facilities which are compatible with the information provided by the considered lot description from the TED dataset. We reduce this set of potential candidates by filtering them depending on the other available fields, through three phases:

1. *Date & Domain filtering phase*: use SQL queries in order to find valid candidate facilities in SIRENE, using the TED agent's activity domain, opening dates and department.
2. *Name filtering phase*: perform an approximate matching based on the names of these valid facilities in order to refine the set of potential candidates.

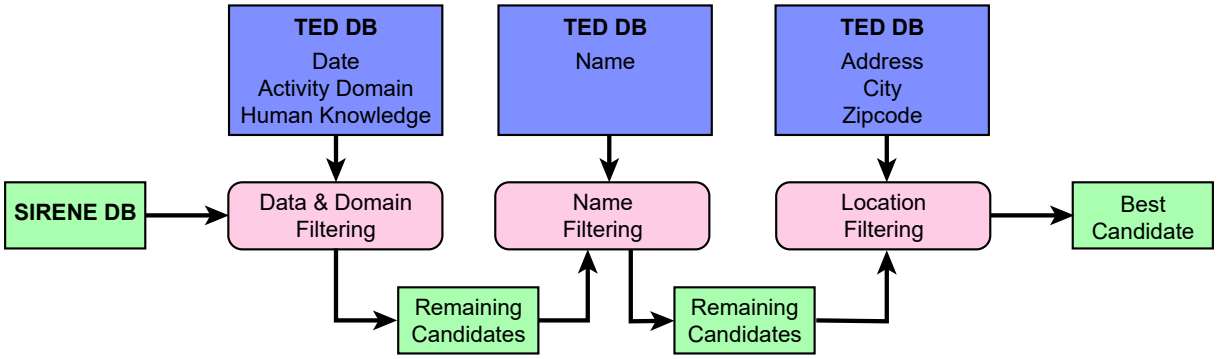


Figure 7. Successive filtering phases of the siretization process. The square boxes represent data, and the round ones processing steps.

3. *Location filtering phase*: perform an approximate matching based on the address, city and zipcode of each potential candidate, in order to find the most likely ones.

At then end, the process outputs the single SIRENE candidate best matching the TED agent. It is possible that the process stops before that point, if no suitable candidate is found during one of the filtering phases.

We could directly try to match the TED agent's name to the whole SIRENE database, but this has several drawbacks. First, this is computationally expensive, as SIRENE contains millions of entries. Second, this would lead to numerous errors, as many agents are likely to have similar names while differing on other characteristics (especially their location). For this reason, the first phase (Section 4.2.1) aims at reducing the number of candidates before performing the name matching in the second phase (Section 4.2.2), in order to select good candidates. Finally, the third phase (Section 4.2.3 aims to rank these candidates in order to select the best one.

4.2.1 Date & Domain Filtering

Each filtering phase focuses on a specific type of information describing the agent in the TED. The first uses temporal and activity-related information, as well as a part of the geographical information and possibly certain aspects of the name:

- Department (first 2 digits of the zipcode);
- Activity domain;
- Opening dates;
- Name.

We first filter by department, using only the first two digits of the zipcode field (*POSTAL_CODE*). Based on our observation of the TED data, it is the most reliable geographical information, and it allows us to greatly reduce the number of candidates. Similarly, we only retain the SIRENE facilities which are related to the activity domain of the targeted TED agent, and which are active at the date of the considered contract.

In addition, we use human knowledge to identify situations allowing to narrow the candidate set even further. The names of certain facilities contain predefined terms that characterize their general nature or activity domain. For example, there are only a few ways to refer to a hospital in France, depending on its role and importance:

```
Common term                         Acronym associated
Centre Hospitalier Regional         CHR
Centre Hospitalier Departemental    CHD
Centre Hospitalier Universitaire    CHU
Centre Hospitalier General          CHG
```

We leverage these terms to constrain the set of candidates even further. In the previous example, this means only searching among the hospitals contained in SIRENE. This allows to significantly decrease the number of candidates.

4.2.2 Name Filtering

The next phase exclusively focuses on the TED agents and SIRENE facility names. We refine the potential SIRENE candidate set obtained at the previous phase, by retaining only the remaining facilities whose name is close enough to the TED agent's name.

For this purpose, we perform an approximate comparison between the TED and SIRENE names, through a method based on the Levenshtein distance [4]. We use the Python library `Fuzzywuzzy`[23], which proposes 4 main string comparison methods:

- `ratio`: simple Levenshtein distance, which is normalized by dividing by the length of the string.
- `partial_ratio`: comparison between the shortest name and all the substrings of the same length found in the longer name.
- `token.sort_ratio`: both names are tokenized, these tokens are sorted alphabetically, then concatenated, before computing the Levenshtein distance on both resulting strings.
- token.set_ratio: same operation as `sort_ratio`, but the common tokens are taken out.

Each of these functions returns a score between 0 (completely different) and 100 (perfectly identical). In the rest of this document, we call this score *similarity*.

As mentioned in the Section 4.1.2, we have at most 4 possible types of names to characterize a SIRENE facility: facility name, entity name, previous entity names, acronyms. We use different similarity functions depending on the types of the available names.

Acronyms We distinguish between two cases: either the TED name field only contains an acronym, or it contains an acronym and some additional text (full name, administrative subdivision, service, department, etc.). Based on empirical estimation, we assume that names shorter than 7 characters generally correspond to the first case (sole acronyms). We handle both cases differently, similarity-wise:

- Sole acronym: we use the `ratio` function, and keep only results with a maximal score, i.e. 100.
- Acronym and other text: we use the `partial_ratio` function, and keep only results with a maximal score.

A slight difference between two acronyms is absolutely not a guarantee of proximity between two companies. Here is an example showing two similar acronyms referring to completely different companies:

SIRET	ACRONYM
48760448000029	EDG
55208131766522	EDF

This is the reason why, in this case, we perform *exact* comparison by retaining only maximal similarity cases.

Other names We use the same function `token_set_ratio` for all other types of names. The only difference lies in the threshold that we set for keeping candidates or not:

- For the cases where we used human knowledge (hospitals, department etc.) at the previous stage (Section 4.2.1), the threshold must be high, since each of the remaining candidates is likely to have a similar name or at least some words in common. We chose an acceptance threshold of 90, which gives suitable results according to our experiments.

[23] https://github.com/seatgeek/thefuzz

- For other cases, we found that the best acceptance threshold is around 70.

Overall Result Based on various approximate comparisons, our approach handles the *Name Pollution* problem (agent names containing irrelevant information) identified in Section 2.3.3. When a facility has several names in SIRENE, we treat each one separately using the above methods. We then keep the highest score as the result of the comparison.

4.2.3 Location Filtering

The last filtering phase takes advantage of the rest of the geographical information, in order to filter the candidate facilities remaining after the previous phase (name filtering):

- City;
- Complete address, i.e. street number, street type and street name.

There are two situations that complicate the comparison of the TED and SIRENE addresses. First, in TED, general address information sometimes appears in the `city` field, an issue we call *Address Pollution* in Section 2.3.4. Here is an example:

ID_NOTICE_CAN	CAE_TOWN
2016156574	LA DEFENSE

In this case, `LA DEFENSE` is not a town but a business district in Paris. Consequently, only matching the city will not return any result.

Second, TED does not always provide all 3 pieces of address information: the number may be missing, or the type of street may be different, etc. However, a single error on one of these three fields does not necessarily invalidate the whole address.

Therefore, in order to perform this comparison, we concatenate all the fields in one string. This means that on the TED side, we merge `address` and `city`; and on the SIRENE side, we merge `streetnumber`, `streettype`, `streetname`, and `city`. This allows making the most of the available information. In the previous example, our method is able to factor the district in the comparison, as it appears in the SIRENE `address` field. Our approach also allows taking care of the *Type Confusion* (mixing geographic and postal addresses) and *Monolithic Address* (combinig various parts of the address in the same field) issues identified in Section 2.3.4.

Based on the concatenated string, we compute a score for each remaining candidate using function `token_set_ratio`. We then take the average between this score and the one obtained at the previous phase (name filtering). Our goal is to boost candidates whose names are very similar to the target's. The candidate with the highest average score is selected as the final result.

4.3 Performance Assessment

In this section, we assess the performance of our siretization method by comparing its output with two distinct ground truths. On the one hand, we leverage TED entries whose SIRET is filled out in the TED (Section 4.3.1). However, we observed that the agents concerned by these cases are mainly clients, which suggests a potential bias, agent-wise. This is why, on the other hand, we consider a random sample of entries without SIRET, which we siretize manually (Section 4.3.2).

4.3.1 Pre-existing SIRETs

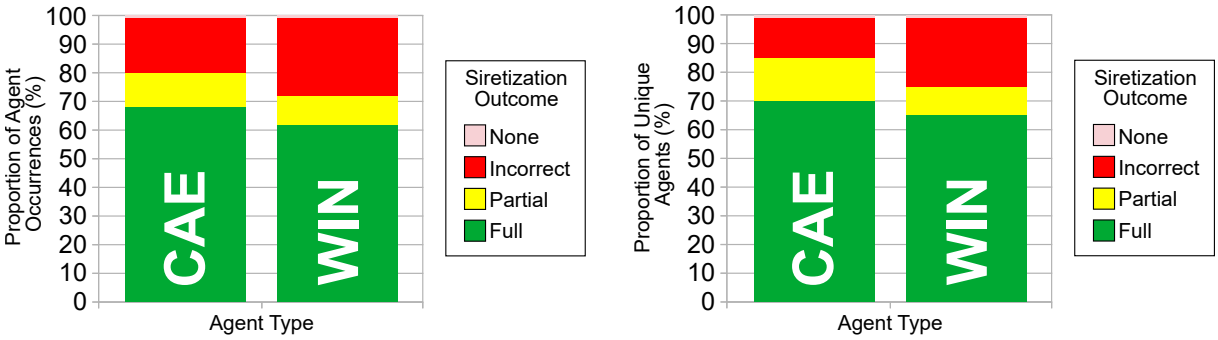
The TED contains $165{,}035$ client *occurrences* (by opposition to unique clients) and $22{,}569$ supplier occurrences, with a valid SIRET. We remove these SIRETs from the database and apply our method, to recover them.

In order to assess the performance of our method, we consider 4 different outcomes:

- *Full SIRET*: the method correctly retrieves the SIRET, i.e. all 14 digits. The matching is then successful.

- *Partial SIRET*: the method only retrieves the SIREN part of the code, i.e. the first 9 digits. Put differently, the SIRENE entity is correct, but the method fails to identify the facility.
- *Incorrect SIRET*: the method selects an incorrect candidate and returns a completely incorrect SIRET.
- *No SIRET*: the method fails to identify any suitable candidate, and returns no SIRET at all.

Figures 8 represents the results of our evaluation, for the SIRETs originally present in the dataset. The x-axis represents the agents type: the left-hand bar focuses on the clients and the right-hand one on the suppliers. The colors represent the four outcomes described before: *Full SIRET* (green), *Partial SIRET* (yellow), *Incorrect SIRET* (red), and *No SIRET* (pink). The y-axis represents the percentage of agent occurrences for each outcome.



(a) Results for agent occurrences.

(b) Results for unique agents.

Figure 8. Distribution of the four possible outcomes of the siretization process for the *predefined* SIRETs.

In the case of already siretized data, the complete success, i.e. the complete identification of the SIRET represents 70% of the client occurrences and 65% of the suppliers occurrences. For clients, we consider that a partial SIRET is also a good result, since procurement management is often centralized at the main facility. Summing up the two, we get close to 90% of success. The result on the suppliers is lower, which is explained by the fact that the awards notices are filled in by the clients. They are more likely to make mistakes or miss some information when describing the suppliers.

Among these client occurrences, the TED contains $5{,}358$ unique clients, and $14{,}181$ unique suppliers. Figure 8b represents the result of our evaluation for unique agents. The performance is lower than when considering occurrences, because of the greater prevalence of small agents, i.e. agents that rarely appear in the data. These agents tend to exhibit more problems in terms of both completion and reliability of the provided information. They are consequently more difficult to identify and siretize.

4.3.2 Manually Annotated SIRETs

To constitute the second ground truth, we first randomly sample 500 agents from the SIRET-less entries of the TED that possess both a city and a name. This sample does not contain multiple occurrences of the same agent, because of its small size. Second, we take advantage of these two fields (name and city) to be able to *manually* retrieve the missing SIRETs of all 500 agents.

This method allows solving the bias present in the predefined SIRET dataset (Section 4.3.1), i.e. the overrepresentation of CAEs. The evaluation method is the same as in Section 4.3.1. Figure 9 represents the obtained results.

Compared to the results obtained for the predefined SIRETs, these performances are better than for unique agents (Figure 8b) and worse than for agent occurrences (Figure 8a).

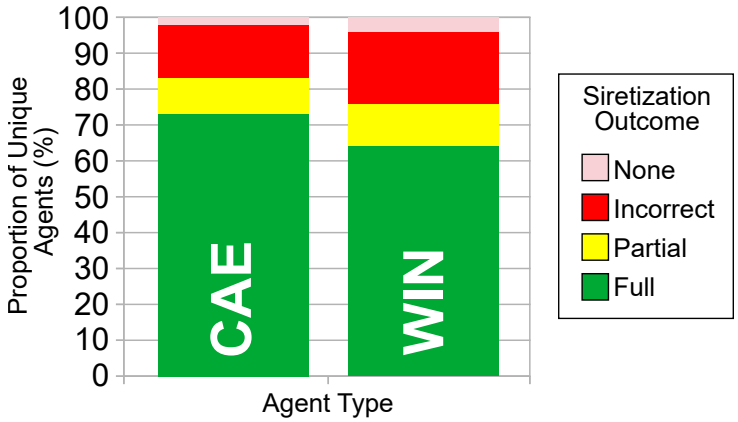


Figure 9. Distribution of the four possible outcomes of the siretization process for the *annotated* SIRETs.

As we randomly select agent occurrences from the TED to constitute the annotated dataset, there is a higher chance to get large agents, since they are more frequent in the database: this could explain this observation.

There is a larger number of agents for which the algorithm is not able to return a SIRET. This is because these agents originally have many missing fields in the TED and/or a poorly written name (compare to the one present in SIRENE). We do find better performances for the customers than for the suppliers, as previously.

5 Step 3: Clustering-Based Merging

As explained in Section 4.3, our siretization process fails to retrieve a reliable SIRET for certain agent occurrences. We assume that a part of these occurrences actually represent the same entities as other successfully siretized occurrences, or even other SIRET-less occurrences. The goal of this step is to group these occurrences under the same entries in our database.

For this purpose, we leverage the Dedupe library, which we describe in Section 5.1. We use this library to perform a cluster analysis of the TED agents, as explained in Section 5.2. We then have to process the clusters produced by Dedupe in order to decide which agent to merge, as described in Section 5.3. Finally, we use a part of our data to assess the performance of this step in Section 5.4.

5.1 Description of Dedupe

Dedupe[24] is a Python library which performs fuzzy matching, record deduplication and entity resolution. Its algorithm is based on three main steps: compute the *record similarity*, use *blocking* to handle large datasets, and perform *cluster analysis* to uncover groups of similar records. Dedupe uses active learning to estimate the best parameter values during all of these steps.

Record similarity In order to compare two strings, Dedupe uses the *Affine Gap* distance [8], a variation of the *Hamming distance* [3]. The Hamming distance is the number of different letters located at the same position in both strings. The affine gap distance allows using gaps between letters (with a penalty), which provides a more flexible matching.

Dedupe compares two records field-by-field, i.e. by considering each field separately, before combining the resulting distance values. It assigns a weight to each field, corresponding to different levels of importance during this comparison. These fields also allow normalizing

[24] https://github.com/Dedupeio/Dedupe

the overall score in order to get a probability value. These weights are data-dependent, and learned during the active learning phase.

Blocking After assigning the weights, one could theoretically compute the distance between each pair of records. However, it is not possible to do so in practice, as it would be computationally too costly. To solve this problem, Dedupe uses a system called *blocking*: the data are divided in groups of records sharing some common patterns. A pattern can be, for instance, having the same value for a specific field, or the same first characters. Each record can be located in one or more so-called blocks.

A blocking rule focuses on a specific subset of fields, on which it defines a set of constraints (strict equality, but also more flexible comparisons). Each rule defines one block, and records respecting several rules at once belong to the different corresponding blocks.

Once the blocks are created, Dedupe only compares records within the same block, in order to avoid comparisons between records that are too different. The rules for creating blocks are data-dependent, and Dedupe learns them during the training phase.

Clustering The last step consists in forming clusters containing similar records. This task is complicated by the fact that Dedupe does not have access to the similarity of certain pairs of records, if they do not belong to the same block. In order to solve this issue, Dedupe uses a *hierarchical clustering with centroid linkage* [5]. The resulting clusters contain groups of records considered as duplicates. In order to perform the clustering, Dedupe leverages a user-defined *cophenetic threshold* [6], i.e. the minimal similarity value for two records to be placed in the same cluster.

Active Learning Active learning requires the user to provide the tool with annotations on specific cases identified as relevant. These are modeled as a set of pairs that:

- are duplicates for Dedupe, but not in the same blocking group;
- are not duplicates for Dedupe, but in the same blocking group.

Dedupe provides a pair of this set to the user, who indicate if they are the same agent or two different agents. Thanks to this new labeled example, Dedupe updates the blocking rules, the weights of the algorithm and the set of pairs. Dedupe proposes new pairs to the user, until he or she decides to stop the process.

5.2 Application to our data

After the siretization phase, we can distinguish two types of agents:

- Siretized agents: each unique SIRET is associated to a single surface form, thanks to the merging step described in Section 3.3.3.
- SIRET-less agents: these can be one of the following three cases:
  - Another surface form of an already siretized agent that our process did recognize correctly;
  - A surface form of an agent that appears under other SIRET-less forms;
  - An agent different from all other agents present in the database (siretized or SIRET-less).

We compare each agent using the non-SIRET fields in our database, i.e.:

- `name`.
- `address`.
- `city`.
- `zipcode`.

Active Learning Phase To start, we perform the active learning phase on 500 pairs. We manually identify the pairs selected by Dedupe, which correspond to 78 positive pairs (different forms of the same agent) and 422 negative pairs (not the same agent).

Here are some examples of the blocking rules used by Dedupe:

- Same first 5 characters on the name field.
- Phonetic matching on the address field.
- Same integer on the address field.
- Same six-gram on the city field.

Clustering Phase The next step is to perform the cluster analysis. We select a conservative cophenetic threshold, because some entries with name and city could be associated despite a difference of city, and thus necessarily of agents. A threshold of 0.8 gives suitable results according to our experiments. After this processing, Dedupe outputs a CSV with 2 additional fields to each agent: a cluster number and a confidence score. The latter is a measure of similarity of the agent of interest, in relation to the other agents in the same cluster.

5.3 Postprocessing

Once we have the Dedupe clusters, we must process them in order to decide which agent occurrences must be merged in our database. In the following, we consider all possible situations and the corresponding actions.

Singleton Cluster A singleton contains only one agent. Dedupe did not find any other agent sufficiently similar to put them together.

If the agent has a SIRET, then we assume that there is no other form of the same agent possessing a different SIRET, and that there is no SIRET-less agent matching it. This SIRET may be incorrect, but at this step, we assume that it is correct. It may be revised at the next step, in case of merging with another siretized agent.

If the agent constituting the singleton is SIRET-less, then the case is a failure, as our process could not identify its SIRET. The next step of our pipeline may succeed later in this task. The agent is assigned a unique id, internal to our FOPPA.

Multiple SIRETs It is possible that Dedupe puts several siretized agent occurrences in the same cluster. In this case, we have what we call a SIRET *conflict*, since all these occurrences correspond to the same agent according to Dedupe, yet they are considered as distinct agents according to our database.

We could either consider that our siretization was incorrect or that Dedupe improperly considers two distinct agents as duplicates. We choose to favor the former assumption, because our previous experiments show that our siretization process can sometimes produce incorrect SIRETs. Typically, when two occurrences of the same agent are poorly filled in the TED, with small disparities between them, the siretization process does not lead to the same result when matching with SIRENE. In addition, as mentioned before, we use a conservative cophenetic threshold with Dedupe.

Consequently, we merge the concerned agent occurrences, using the same strategy as in Section 3.3.3: we keep the most frequent value for each field, including the SIRET. In the case of the SIRET, we also consider the number of occurrences of each agent in the original dataset to determine which unique agent is majority in the cluster. The rationale behind this strategy is to favor more frequent agents, as their information tend to be more reliable in the TED. If the cluster contains SIRET-less agents, they are also merged during the process.

Other Cases If the cluster contains several agents without any SIRET, then we assume that they are all different occurrences of the same agent. We combine them all, to get a single entry in our database, identified by its own unique internal id. If one of the agents has a SIRET, then we also use it to identify the combined entry.

5.4 Performance Assessment

In this section, we assess the performance of the clustering step. We first present some general statistics regarding the size of the clusters identified by Dedupe (Section 5.4.1). Then,

we propose two methods to assess the amount of false positives (Section 5.4.2) and false negatives (Section 5.4.3), respectively. The *false positives* are agent occurrences placed in the same cluster by Dedupe, when they actually correspond to several distinct unique agents, and should therefore be located in different clusters. The false negatives are agent occurrences placed in different clusters by Dedupe, when they actually correspond to the same unique agent, and should therefore be located in a single cluster.

5.4.1 Cluster Sizes

The clustering process distributes the 273,526 agent occurrences over 252,910 clusters. These are small, with an average size of 1.08 agent occurrences by cluster. Table 3 shows the full distribution of the cluster sizes, and it appears that most clusters are singletons (94%).

Cluster size	1	2	3	4	5	6	7	8	9	10+	Total
Count	237,994	12,203	1,824	490	166	66	37	34	12	83	252,910
Proportion	94.11%	4.82%	0.72%	0.19%	0.07%	0.03%	0.01%	0.01%	0.01%	0.03%	100%

Table 3. Distribution of the number of agent occurrences per cluster.

As explained in Section 5.3, singleton clusters do not require any additional processing during the post-processing: if they have a SIRET, then it is assumed correct (for now), and if they do not, it means that Dedupe could not find one. The remaining 14,916 clusters contain several agent occurrences, possibly corresponding to a single or several ids (SIRET or SIREN).

5.4.2 False Positives

False positives correspond to SIRET or SIREN conflicts, as defined in Section 5.3. Table 4 represents the distribution of clusters according to their numbers of distinct ids (SIRETs and SIRENs). By comparison, Table 3 focuses on agent occurrences, not ids.

Number of distinct ids	0	1	2	3	4	5+	Total
Frequency for SIRETs	48,384	191,415	11,129	1,497	328	157	252,910
Frequency for SIRENs	48,384	193,328	9,637	1,194	257	110	252,910

Table 4. Distribution of the number of distinct ids by cluster, in terms of SIRET and SIREN.

There are 241,712 (96%) and 239,799 (95%) clusters with no conflict (i.e. they contain 0 or 1 id), in terms of SIRENs and SIRETs, respectively. That leaves us with a total of 13,111 and 11,198 conflicted clusters. These numbers respectively amount to 88% and 75% of the 14,916 clusters containing several agent occurrences (cf. Section 5.4.1). Among them, 1,913 clusters are conflicted according to SIRETs, but not when focusing only on SIRENs.

There can be two reasons for these conflicts: either the siretization process is incorrect and the concerned occurrences should have the same SIRET, or the clustering step is incorrect and those are indeed different agents that should be kept separated. The reliability of the siretization step is already assessed in Section 4.3: here, we want to focus on the latter case.

In order to investigate the performance of the clustering step, we compute the same statistics as in Table 4, but while focusing only on the SIRETs and SIRENs *that were originally provided by the TED* (i.e. excluding those resulting from our siretization step). There are 18,435 TED SIRETs and 16,288 TED SIRENs. Our assumption here is that the SIRETs and SIRENs originating from the TED are certainly correct, and should therefore be placed in distinct clusters. As our siretization involves merging all agent occurrences possessing the same SIRET, each TED SIRET appears once and only once at the clustering step.

Table 5 shows the distribution of clusters according to the number of TED ids they contain. We get a total of 16,988 clusters containing at least one such SIRET. According to the table, 86%

of the TED SIRETs are correctly placed in singleton clusters, whereas the rest are incorrectly mixed with other TED SIRETs.

Number of distinct TED ids	1	2	3	4	5+	Total
Frequency for TED SIRETs	15,971	1,013	139	32	13	16,988
Frequency for TED SIRENs	16,276	646	52	11	3	16,988

Table 5. Distribution of the number of original TED ids by cluster, in terms of SIRETs and SIRENs.

When characterizing clusters in terms of their numbers of distinct SIRENs instead of SIRETs, the performance is slightly higher: 88%. Moreover, 2% of the unique TED SIRETs confused by Dedupe correspond to facilities belonging to the same entity (according to the SIRENE terminology), i.e. they have the same SIREN. After a manual verification, we conclude that these cases are most likely due to clerks indicating a SIRET associated to another geographical address (a different facility related to the same entity), or to a previous facility. It is therefore possible to get several different SIRETs for the same facility, which constitutes an error. This, in turn, causes Dedupe to create clusters with homogeneous SIRENS but heterogeneous SIRETs.

These results show the performance of our clustering step in terms of clusters. But ultimately, we are interested in the quality of the agent SIRETs. In terms of false positives, we can distinguish three different outcomes. They depend on the SIRET assigned to each agent based on its cluster, according to the post-processing described in Section 5.3:

- *Full SIRET*: the SIRET is correct. This can match two situations: either the method puts the agent in a singleton cluster, or it puts it in a multi-SIRET cluster in which its SIRET is majority.
- *Partial SIRET*: the SIRET is incorrect, but the SIREN is correct. This happens when the method puts the agent in a multi-SIRET cluster in which the majority SIRET is different from the agent's, but with a common SIREN.
- *Incorrect*: neither the SIRET or the SIREN are correct. This situation corresponds to the case where the method puts the agent in a multi-SIRET cluster whose majority SIRET has nothing in common with the agent's.

Table 6 summarizes this aspect of the performance. A large portion of the clusters contain only a single TED id and therefore, most agents are associated with the correct SIRET. Only 5% of the considered unique agents end up with a completely incorrect id, or 8% with a partially incorrect id.

Number of distinct TED ids	Full	Partial	Incorrect	Total
Count	16,963	630	842	18,435
Proportion	92%	3%	5%	100%

Table 6. Distribution of the three possible outcomes of the clustering process for the original TED SIRETs and SIRENs.

5.4.3 False Negatives

False negatives correspond to agent occurrences incorrectly placed in different clusters by Dedupe. In order to assess this type of error, we cannot use the same data as when studying the false positives in Section 5.4.2, because the forms of these agent occurrences do not exhibit enough diversity. Instead, we adopt a specific procedure to constitute a more appropriate dataset.

First, we randomly sample the agent occurrences of the original TED dataset, in order to constitute our ground truth. We perform this sampling under the following constraints. Each

such occurrence must correspond to an agent appearing *several times* in the original TED data, and under *different forms* in this sample. Moreover, each sampled agent must have a SIRET[25]. Second, we ignore these agent occurrences during the siretization step, which is only applied to the rest of the data. Third, we conduct the Dedupe-based clustering phase on the whole dataset, including the sample. Finally, we assess the false negatives produced during this last step by studying how the sampled agent occurrences that represent the same unique agent are distributed over the clusters identified by Dedupe.

Our sample contains $5{,}020$ occurrences, that correspond to 377 unique agents (i.e. there are 377 different SIRETs in the sample), for an average of 13.31 occurrences by agent. Each SIRET appears between 1 and 538 times in the sample. To assess how these occurrences are distributed over the Dedupe cluster, we compute two measures.

Concentration Ratio The first is what we call the *Concentration Ratio* $CR(a)$, which is defined for an agent a of interest. It is the maximal proportion of occurrences of this agent in a single cluster, relative to the total number of occurrences of this same agent in the sample:

$$CR(a) = \max_{C \in \mathcal{C}} \frac{|C \cap A|}{|A|}, \tag{2}$$

where $\mathcal{C} = \{C_1,...,C_k\}$ is the partition constituted of k clusters C_i detected by Dedupe, A is the set of all occurrences of a in the sample, and $|...|$ denotes the cardinality of a set. A concentration ratio close to one indicates that the occurrences of the same agent are located in a single cluster, and thus that the agent was well clustered by Dedupe. On the contrary, a low ratio shows that these occurrences are distributed over a number of clusters.

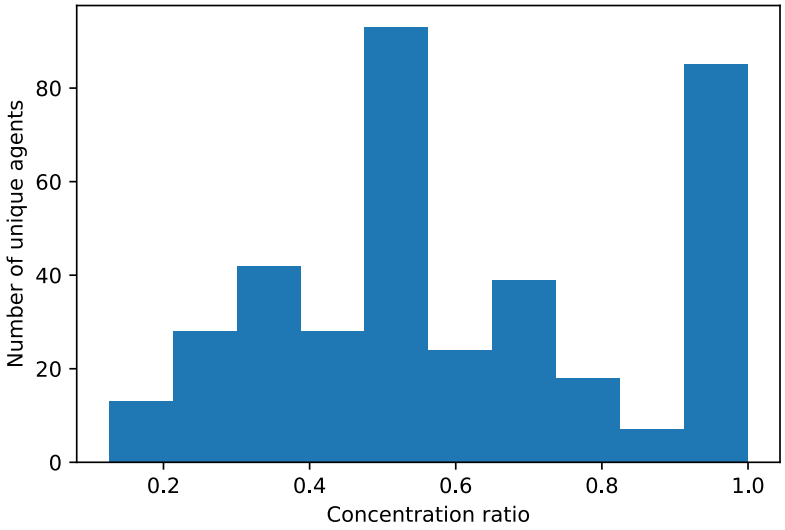


Figure 10. Distribution of the concentration ratio over unique agents.

Figure 10 shows the distribution of the concentration ratio over unique agents. The x-axis represents the concentration ratio, and the y-axis represents the number of agents. The mean concentration ratio is 0.6, which means that Dedupe puts more than half of the occurrences of an agent in the same cluster, in average. Dedupe perfectly clusters 243 agent occurrences (5%), representing 83 unique agents (22%). These are clusters with 2 or 3 occurrences: it is apparently hard to gather many occurrences of the same agent in a single cluster. The others agents are less concentrated, with a majority of clusters containing half of their occurrences.

[25] It can be a TED SIRET, or a SIRET estimated at the siretization step.

In order to define an overall performance measure, we sum the concentration ratio over all TED SIRETs, using their frequencies as weights. We get a value of 0.43, which is consistent with our previous observations.

Singleton Ratio Among the occurrences that are not gathered in the same cluster, for a given agent, we consider differently those each forming a singleton cluster, and those forming a cluster with occurrences of other agents. Indeed, the former correspond to false negatives, whereas the latter are false positives. Since we focus on the former in this section, we propose the *Singleton Ratio* $SR(a)$ to characterize them. Like the *Concentration Ratio*, it is computed for an agent of interest a. It corresponds to the ratio of its number of occurrences forming singleton clusters, to its total number of occurrences in the sample:

$$SR(a) = \frac{|\{C \subset \mathcal{C} : C \subset A \wedge |C| = 1\}|}{|A|}. \tag{3}$$

A singleton ratio close to 1 indicates that all the agent occurrences are distributed in their own cluster. On the contrary, a ratio close to zero shows that the occurrences belong to larger clusters (possibly with other occurrences of the same agent, or not).

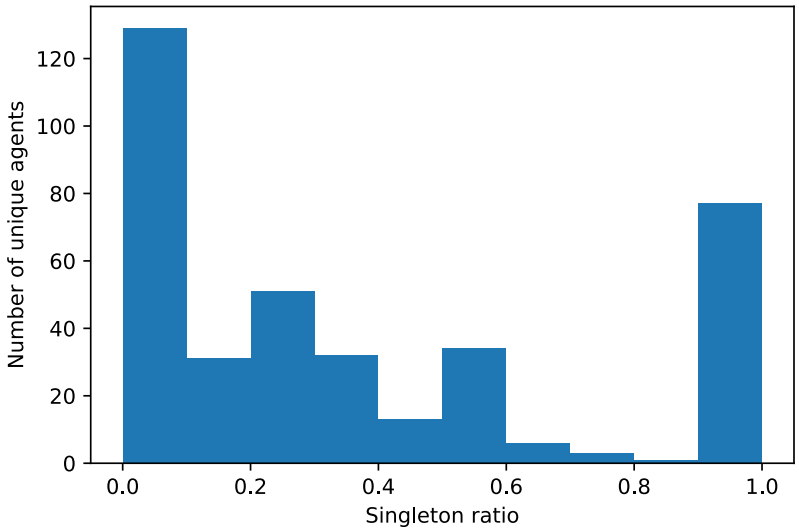


Figure 11. Distribution of the singleton ratio over unique agents.

Figure 11 shows the distribution of the singleton ratio over unique agents. The x-axis represents the singleton ratio and the y-axis the number of unique agents. On the one hand, there are 77 unique agents (20%) with a ratio of 1. This means that all related occurrences have been located in separate clusters. This is the case for occurrences that are very different, for example occurrences not sharing the same name. This result confirms the relevance of our siretization process: some occurrences can only be gathered by finding the correct SIRET number. On the other hand, the ratio of the rest of the agents is much lower. The mean singleton ratio for these other agents (i.e. without considering ratios equal to 1) is 0.18. This means that Dedupe splits the set of occurrences linked to a SIRET into a limited number of subgroups.

As with the CR before, we compute an overall measure by summing the singleton ratio over all agents, using their frequencies as weights. We get a value of 0.52 indicating a relatively high level of dispersion of the agent occurrences over the clusters.

Concluding Remarks As mentioned in Section 5.3, our post-processing includes the merging of agent occurrences located in the same cluster, even if they do not have the same SIRET.

For this purpose, we keep the most frequent SIRET, i.e. the one linked to the greatest number of occurrences.

Considering a given unique agent at the end of the clustering step, some of its occurrences can be assigned the correct SIRET, but others could receive only a partially correct id (same SIREN), or no id at all. Considering all possible outcomes makes it difficult to assess the performance of this step in a meaningful way. For this reason, we adopt a simplified view by focusing only on how the *absolute majority* of the agent's occurrences are treated. We distinguish the following five possible situations:

- *Full SIRET*: most of the agent's occurrences belong to the same cluster, whose majority SIRET matches the agent's. Most occurrences of this agent consequently get their correct SIRET.
- *Partial SIRET*: most of the agent's occurrences get a SIRET compatible with the agent's SIREN. This happens either when the occurrences are concentrated in the same cluster, but are a minority, or when the occurrences are scattered over several clusters.
- *Incorrect SIRET*: most of the agent's occurrences get a SIRET incompatible with the agent's SIREN. The situations leading to this case are similar to the previous one, except with completely different SIRETs.
- *No SIRET*: most of the agent's occurrences do not receive any SIRET at all.
- *No Majority*: there is no absolute majority for any of the four above situations. All of them may occur for the considered agent, but none dominates.

Table 7 summarizes this aspect of the performance. More than half of the unique agents (54%) see most of their occurrences left without any SIRET at all. This is because these occurrences are mostly located in singleton or SIRET-less small clusters. For 37% of the unique agents, most of their occurrences receive a SIRET. Focusing only on these cases, this SIRET is correct for 75% of the unique agents.

Number of unique agent	Full	Partial	Incorrect	None	No Decision	Total
Count	105	9	28	204	31	377
Proportion	28%	2%	7%	54%	9%	100%

Table 7. Distribution of the four possible outcomes of the clustering process.

To summarize these results: when our process is able to assign a SIRET to a unique agent, it is frequently correct; however our process is not able to identify a SIRET for most agents. This last comment should be modulated though, due to the data we used to perform this assessment. By construction, these data contain many occurrences of the same agent; however, in practice, this is unlikely, as it would require the agent occurrences to take a number of very different forms. The role of the siretization step is to merge the distinct forms of the same unique agent that are similar enough. After this step, one agent should have only one form, or a few very different ones.

6 Conclusion

To conclude, our process aims to transform agent occurrences into unique agents. Table 8 represents the number of unique agents during each step of the process. The first column represents the number of unique agents after a first gathering of occurrences with the same address, city and name fields. The second column represents the number of unique agents after the siretization step, and the third column represents the number of unique agents after the clustering step.

In this process, the siretization step is the most important step in order to lower the number of unique agents. However, this number of agents is reduced by another 15% thanks

Before the siretization	After the siretization	After the clustering
889,692	273,525	252,910

Table 8. Unique agents for each step.

to the clustering part. We must indicate that the siretization and the clustering steps are complementary:

- the siretization step consider historical data. Therefore, it is possible to gather occurrences representing the same agent, but filled with dissimilar information.
- the clustering step is more cautious, and only match occurrences that are very close semantically. However, this step will gather occurrences that are poorly filled, which give very variable results via the siretization step are correctly.

References

[1] BOAMP. *Avant de répondre à un marché public*. Bulletin Officiel des Marchés Publics. 2020. URL: https://www.boamp.fr/Espace-entreprises/Comment-repondre-a-un-marche-public/Questions-de-reglementation/Avant-de-repondre-a-un-marche-public/Sommaire (visited on 10/07/2021).

[2] M. Fazekas and I. J. Tóth. "From corruption to state capture: A new analytical framework with empirical applications from Hungary". In: *Political Research Quarterly* 69.2 (2016), pp. 320–334. DOI: 10.1177/1065912916639137.

[3] R. W. Hamming. "Error Detecting and Error Correcting Codes". In: *Bell System Technical Journal* 29.2 (1950), pp. 147–160. DOI: 10.1002/j.1538-7305.1950.tb00463.x.

[4] L. V. I. "Binary codes capable of correcting deletions, insertions, and reversals". In: *Soviet physics. Doklady* 10 (1965), pp. 707–710.

[5] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. DOI: 10.1017/cbo9780511809071.

[6] R. R. Sokal and J. F. Rohlh. "The comparison of dendrograms by objective methods". In: *TAXON* 11 (1962), pp. 33–40. DOI: 10.2307/1217208.

[7] Tenders Electronic Daily. *TED CSV Open Data*. Tech. rep. Tenders Electronic Daily, 2021. URL: https://data.europfa.eu/euodp/en/data/storage/f/2022-02-14T122429/TED%5C%28csv%5C%29_data_information_v3.4.pdf.

[8] M. Vingron and M. S. Waterman. "Sequence alignment and penalty choice". In: *Journal of Molecular Biology* 235.1 (1994), pp. 1–12. DOI: 10.1016/s0022-2836(05)80006-3.

A Database Changelog

Table 9 describes the different versions of the FOPPA database.

Version	Date	Description
`1.0.0`	26/01/2022	First version of the FOPPA database.
`1.0.1`	28/03/2022	Five changes: 1) Added Boolean fields to `Lot` table; 2) Fixed issues related to criterion weights; 3) Fixed normalized issues related to missing data; 4) Deleted agents related only to non-awarded lots; 5) Normalized dates.
`1.0.2`	22/04/2022	Added two new fields: `country` (country of the agents); `department` (additional information for overseas departments and Corsica).
`1.0.3`	31/05/2022	Correction of non-siretized agent's names and geographical information.

Table 9. History of the FOPPA database versions.

B Procedure-Related Information

As explained in Section 1, when a contract is above the European threshold of the concerned activity domain, it is necessary to follow a formalized procedure. Table 10 shows the evolution of the European thresholds over time.

Client	Sector	Type of contract	Threshold (€)
01/01/2004–31/12/2004			
Central public authority	All	Goods, Services	162,000
Local public authority	All	Goods, Services	249,000
Public entity	All	Goods, Services	499,000
Public authority/entity	All	Works, Concessions	6,242,000
01/01/2005–31/12/2005			
Central public authority	All	Goods, Services	154,000
Local public authority	All	Goods, Services	236,000
Public entity	All	Goods, Services	473,000
Public authority/entity	All	Works, Concessions	5,923,000
01/01/2006–31/12/2007			
Central public authority	All	Goods, Services	137,000
Local public authority	All	Goods, Services	211,000
Public entity	All	Goods, Services	422,000
Public authority/entity	All	Works, Concessions	5,278,000
01/01/2008–31/12/2009			
Central public authority	All	Goods, Services	133,000
Local public authority	All	Goods, Services	206,000
Public entity	All	Goods, Services	412,000
Public authority/entity	All	Works, Concessions	5,150,000
01/01/2010–31/12/2011			
Central public authority	All	Goods, Services	125,000
Local public authority	All	Goods, Services	193,000
Public entity	All	Goods, Services	387,000
Public authority/entity	All	Works, Concessions	4,845,000
01/01/2012–31/12/2013			
Central public authority	All	Goods, Services	130,000
Local public authority	All	Goods, Services	200,000
Public entity	All	Goods, Services	400,000
Public authority/entity	All	Works, Concessions	5,000,000
01/01/2014–31/12/2015			
Central public authority	All	Goods, Services	134,000
Local public authority	All	Goods, Services	207,000
Public entity	All	Goods, Services	414,000
Public authority/entity	All	Works, Concessions	5,186,000
01/01/2016–31/12/2017			
Central public authority	All	Goods, Services	135,000
Local public authority	All	Goods, Services	209,000
Public entity	All	Goods, Services	418,000
Public authority/entity	All	Works, Concessions	5,225,000
01/01/2018–31/12/2019			
Central public authority	Normal	Goods, Services	144,000
Central public authority	Special	Goods, Services	221,000
Local public authority	All	Goods, Services	221,000
Public entity	All	Goods, Services	443,000
Public authority/entity	All	Works, Concessions	5,548,000
01/01/2020–31/12/2021			
Central public authority	Normal	Goods, Services	139,000
Central public authority	Special	Goods, Services	214,000
Local public authority	All	Goods, Services	214,000
Public entity	All	Goods, Services	428,000
Public authority/entity	All	Works, Concessions	5,350,000

Table 10. Evolution of the European thresholds. The term *Special* refers to derogatory activity sectors.

Here are the resources used to constitute this table:

2004. http://www.marche-public.fr/Marches-publics/Textes/Directives/2004-18-CE/Montant-seuils-marches-publics.htm

2005. https://www.lemoniteur.fr/article/seuils-d-application-en-matiere-de-procedures-de-passation-des-marches-modification-des-directives-2004-17-ce-et-2004-18-ce-du-parlement-europeen-et-du-conseil.1885024

2007. https://www.lemoniteur.fr/article/seuils-d-application-en-matiere-de-procedures-de-passation-des-marches-au-1er-janvier-2006-modification-des-directives-2004-17-ce-et-2004-18-ce.729864

2009. https://www.lemoniteur.fr/article/seuils-europeens-au-1er-janvier-2008-pour-la-passation-des-marches-publics.1737704

2011. https://www.lemoniteur.fr/article/marches-publics-de-nouveaux-seuils-au-1er-janvier-2010.589449

2013. https://www.lemoniteur.fr/article/marches-publics-de-nouveaux-seuils-europeens-au-1er-janvier-2012.1050484

2015. http://www.marche-public.fr/contrats-publics/DAJ-maj-seuils-2016.htm

2017. https://www.boamp.fr/Espace-acheteurs/Actualites/Archives/Nouveaux-seuils-applicables-aux-marches-publics

2019. http://www.marche-public.fr/Marches-publics/Definitions/Entrees/Seuil.htm

2021. https://www.economie.gouv.fr/daj/marches-publics-nouveaux-seuils-europeens-applicables-au-1er-janvier-2020

C Fields of the TED dataset

This section gives the comprehensive list of all fields present in the TED CSV files used to initialize our database with CANs. We break down this list by categories, as indicated in the official TED documentation [7].

Certain fields are directly extracted from the formed filled by the CAEs, whereas others are computed based on other fields. The latter are indicated with an asterisk (*).

C.1 Notice Metadata

Table 11 presents the TED fields related to the general information of the notice.

Name	Data Type	Description	Version
ID_NOTICE_CAN	Integer	Unique ID of the contract award notice	all
TED_NOTICE_URL	String	URL of the notice on the TED Website	all
YEAR	Date	Year of publication of the notice	all
ID_TYPE	Integer	Code representing which directive type the notice falls under	all
DT_DISPATCH	Date	Date when the notice was sent to the TED for publication	all
XSD_VERSION*	R20X.SX	Version of the XML Schema definition	2.0.5
CANCELLED*	Boolean	Whether the notice was canceled (1) or not (0)	all
CORRECTIONS*	Integer	Number of later correction notices	all

Table 11. General TED fields related to the notice.

C.2 CAE Identification

Table 12 presents the TED fields focusing on the client(s). Some of these fields take a value among several predefined ones, which are listed below.

Name	Data Type	Description	Version
B_MULTIPLE_CAE*	Boolean	Whether the notice involves several CAEs	2.0.9
CAE_NAME	String	Name(s) of the CAE(s)	all
CAE_NATIONALID	String	National registration number(s) of the CAE(s)	all
CAE_ADDRESS	String	Postal address(es) of the CAE(s)	all
CAE_TOWN	String	City(s) of the CAE(s)	all
CAE_POSTAL_CODE	String	Zipcode(s) of the CAE(s)	all
CAE_GPA_ANNEX*	Enum	WTO Classe(s) of the CAE(s) (only for 2014–2016)	all
ISO_COUNTRY_CODE	String	ISO code for the country of the first CAE	all
ISO_COUNTRY_CODE_GPA*	String	ISO code for the *legal* country of the first CAE (only in 2014–2016)	all
B_MULTIPLE_COUNTRY*	Boolean	Whether the first CAE is related to several countries	2.0.9
ISO_COUNTRY_CODE_ALL	String	List of all other ISO country codes	2.0.9
CAE_TYPE*	Enum	Type of the contracting authority (ministry, regional, local...)	all
EU_INST_CODE	Enum	Subtype, if the CAE is an EU institution	2.0.9
MAIN_ACTIVITY	Enum	Main Activity of the CAE(s)	all

Table 12. TED fields related to the client.

WTO GPA Field `CAE_GPA_ANNEX` leverages the classification defined by the WTO Government Procurement Agreement (GPA), as detailed online[26].

CAE Type Field `CAE_TYPE` can contain the following values [7]:

- `1`: Ministry or any other national or federal authority, including their regional of local subdivisions;
- `3`: Regional or local authority;
- `4`: Utilities sectors;
- `5`: European Union institution/agency;
- `5A`: Other international organization;

[26] http://www.wto.org/english/tratop_e/gproc_e/appendices_e.htm#ec

- `6`: Body governed by public law;
- `8`: Other;
- `N`: National or federal Agency / Office;
- `R`: Regional or local Agency / Office;
- `Z`: Not specified.

EU Institution Code If the CAE is an EU institution (CAE Type `5`), then field `EU_INST_TYPE` indicates its precise type [7]:

- `AG`: Agencies;
- `BC`: European Central Bank;
- `BI`: European Investment Bank;
- `BR`: European Bank for Reconstruction and Development;
- `CA`: European Court of Auditors;
- `CJ`: Court of Justice of the European Union;
- `CL`: Council of the European Union;
- `CR`: European Committee of the Regions;
- `EA`: European External Action Service;
- `EC`: European Commission;
- `ES`: European Economic and Social Committee;
- `FI`: European Investment Fund;
- `OB`: European Patent Office;
- `OP`: Publications office of the European Union;
- `PA`: European Parliament.

Main Activity Field `MAIN_ACTIVITY` represents the area of activity of the CAE. It relies on the Classification of the Functions of Government (COFOG)[27], which we reproduce here:

- `General public services`: Executive and legislative organs, financial and fiscal affairs, external affairs; foreign economic aid; general services; basic research; R&D related to general public services; general public services n.e.c.; public debt transactions, transfers of a general character between different levels of government.
- `Defence`: Military defence; civil defence; foreign military aid, R&D related to defence; defence n.e.c. (not elsewhere classified).
- `Public order and safety`: Police services; fire-protection services; law courts; prisons; R&D related to public order and safety; public order and safety n.e.c.
- `Economic affairs`: General economic, commercial and labour affairs; agriculture, forestry; fishing and hunting; fuel and energy; mining, manufacturing and construction; transport; communication; other industries, R&D related to economic affairs; economic affairs n.e.c.
- `Environmental protection`: Waste management; water waste management; pollution abatement; protection of biodiversity and landscape; R&D related to environmental protection.
- `Housing and community amenities`: Housing development; community development; water supply; street lighting; R&D related to housing and community amenities; housing and community amenities n.e.c.
- `Health`: Medical products, appliances and equipment; outpatient services; hospital services; public health services; R&D related to health; health n.e.c.
- `Recreation, culture and religion`: Recreational and sporting services; cultural services; broadcasting and publishing services; religious and other community services, R&D related to recreation, culture and religion; recreation; culture and religion n.e.c.
- `Education`: Pre-primary, primary, secondary and tertiary education, post-secondary non-tertiary education, education non definable by level, subsidiary services to edu-

[27] https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Classification_of_the_functions_of_government_(COFOG)

cation, R&D; n.e.c.
- `Social protection`: Sickness and disability; old age; survivors; family and children; unemployment; housing; R&D; social protection and social exclusion n.e.c.

C.3 Notice and Lot Level Variables

Table 13 shows the fields describing whole contracts and lots. Fields taking values in an enumerated collection are detailed below.

Name	Data Type	Description	Version
B_ON_BEHALF	Boolean	Whether the contract involves several buyers	all
B_INVOLVES_JOINT_PROCUREMENT	Boolean	Whether the contract is a joint procurement	2.0.9
B_AWARDED_BY_CENTRAL_BODY	Boolean	Whether the CAE is a central purchasing body	2.0.9
TYPE_OF_CONTRACT	Enum	Contract related to works, supplies or services	all
TAL_LOCATION_NUTS	Enum	NUTS code for the main location of work	all
B_FRA_AGREEMENT	Boolean	Notice *declared* as related to a framework agreement (FA)	all
FRA_ESTIMATED*	Enum	Notice *estimated* as related to a FA	all
B_FRA_CONTRACT*	Boolean	Notice *estimated* as related to contracts within a FA	all
B_DYN_PURCH_SYST	Boolean	Notice involving a dynamic purchasing system	all
CPV	Enum	Main common procurement vocabulary code (2008 version)	all
MAIN_CPV_CODE_GPA*	Enum	Cleaned version of the main CPV	all
ADDITIONAL_CPVS	Enum	Additional CPV codes	all
B_GPA	Boolean	Contract covered by the Government Procurement Agreement	all
GPA_COVERAGE*	Enum	Detailed information about GPA coverage (only for 2014–2016)	all
ID_LOT	Integer	Unique id of the Lot	2.0.9
LOTS_NUMBER*	Integer	Number of lots in the contract (since 2009)	all
VALUE_EURO	Float	Pre-tax CAN value (€)	all
VALUE_EURO_FIN_1*	Float	Pre-tax CAN value (€), automatically estimated from other fields	all
VALUE_EURO_FIN_2*	Float	Pre-tax CAN value (€), manually estimated	all
B_EU_FUNDS	Boolean	Whether the contract is related to a project funded by the EU	all
TOP_TYPE	Enum	Type of procedure	all
B_ACCELERATED	Boolean	Whether the awarding procedure was accelerated	all
OUT_OF_DIRECTIVES	boolean	CAN published even though there was no CN	all
CRIT_CODE	Enum	Main award criterion	all
CRIT_PRICE_WEIGHT*	Float	Weight of the price criterion	2.0.9
CRIT_CRITERIA	String	Additional award criteria	all
CRIT_WEIGHTS	Float	Weights of the additional criteria	all
B_ELECTRONIC_AUCTION	Boolean	Whether an electronic auction was conducted	all
NUMBER_AWARDS*	Integer	Number of different winners for the lot	all

Table 13. TED fields related to the notices and lots.

On Behalf In field `B_ON_BEHALF`, the involvement of several clients can be due to a joint procurement or to the client being a central purchasing body. This is specified in fields `B_INVOLVES_JOINT_PROCUREMENT` and `B_AWARDED_BY_CENTRAL_BODY`, respectively.

Type of Contract . Field `TYPE_OF_CONTRACT` can be one of the following:
- `W`: Works;
- `U`: Supplies;
- `S`: Services.

Main Location Field `TAL_LOCATION_NUTS` shows the main location of work, place of delivery or of performance [7]. It is a NUTS code (*Nomenclature des Unités territoriales statistiques* – Nomenclature of Territorial Units for Statistics)[28].

Relation to Framework Agreement Field `FRA_ESTIMATED` indicates the (possible) relation automatically detected between the notice and a framework agreement [7]:
- `K`: keyword "framework" found in the title or description of the notice;
- `A`: multiple awards were given per one lot;
- `C`: most of the notices which following this notice are marked as framework agreement.

[28] https://ec.europa.eu/eurostat/web/nuts/background

GPA Coverage Field `GPA_COVERAGE` indicates how the contract is cover (or not) by the Government Procurement Agreement [7]:

- `1`: covered by GPA;
- `2`: entity not covered by GPA;
- `3`: entity covered, but contract not covered by GPA;
- `4`: below-thresholds contract;
- `5`: contracting entity is not an EU public entity.

Value Fields `VALUE_EURO_FIN_1` is an estimation of the pre-tax CAN value for the case where field `VALUE_EURO` is empty. The estimation method is provided in Appendix I of [7]. Field `VALUE_EURO_FIN_2` is most often equal to `VALUE_EURO_FIN_1`, but can include an additional manual correction.

Type of Procedure Field `TOP_TYPE` shows the type of procedure used to award the contract [7]:

- `AWP`: award without prior publication of a contract notice;
- `COD`: competitive dialogue;
- `NOC/NOP`: negotiated without a call for competition;
- `NIC/NIP`: "negotiated with a call for competition;
- `OPE`: open procedure;
- `RES`: restricted procedure;
- `INP`: innovative partnership.

Award Criteria Field `CRIT_CODE` indicates the criteria considered during the awarding procedure [7]:

- `L`: lowest price;
- `M`: most economically advantageous tender.

C.4 Award Metadata

Table 14 shows the fields describing awards. Fields taking values in an enumerated collection are detailed below.

Name	Type	Description	Version
`ID_AWARD`	Integer	Unique id for the contract award	all
`ID_LOT_AWARDED`	Integer	Unique id of the concerned lot	all
`INFO_ON_NON_AWARD`	Enum	Reasons why the contract was not awarded	all
`INFO_UNPUBLISHED`	Boolean	Whether some confidential information was not published	all

Table 14. TED fields related to the awards.

Contract Not Awarded Field `INFO_ON_NON_AWARD` is empty if the contract was awarded. Otherwise, it indicates why it was not awarded [7]:

- `PROCUREMENT_UNSUCCESSFUL`: no tenders or requests to participate were received, or all were rejected;
- `PROCUREMENT_DISCONTINUED`: other reasons (discontinuation of procedure).

C.5 Winning Bidder Identification

Table 15 presents the fields related to the winner(s) of the awarding process. If the contract is awarded to several winners, only the first one is supposed to be described by these fields [7].

C.6 Other CA level variables

Table 16 presents the remaining fields, related to the contract award.

Name	Type	Description	Version
B_AWARDED_TO_A_GROUP	Boolean	Whether the contract was awarded to several winners	2.0.9
WIN_NAME	String	Official name of the winner	all
WIN_NATIONALID	String	National registration number of the winner	2.0.9
WIN_ADDRESS	String	Postal address of the winner	all
WIN_TOWN	String	City of the winner	all
WIN_POSTAL_CODE	String	Zipcode of the winner	all
WIN_COUNTRY_CODE	String	ISO country code of the winner	all
B_CONTRACTOR_SME	Boolean	Whether the winner is an SME	2.0.9

Table 15. TED fields related to the winner.

Name	Type	Description	Version
CONTRACT_NUMBER	Integer	Unique id of the contract	all
TITLE	String	Title of the contract	all
NUMBER_OFFERS	Integer	Total number of tenders received	all
NUMBER_TENDERS_SME	Integer	Number of tenders from SMEs	2.0.9
NUMBER_TENDERS_OTHER_EU	Integer	Number of tenders from other EU states	2.0.9
NUMBER_TENDERS_NON_EU	Integer	Number of tenders from non-EU states	2.0.9
NUMBER_OFFERS_ELECTR	Integer	Number of offers received electronically	all
AWARD_EST_VALUE_EURO	Float	Estimated pre-tax CA value (€)	all
AWARD_VALUE_EURO	Float	Effective pre-tax CA Value, or lowest bid (€)	all
AWARD_VALUE_EURO_FIN_1*	Float	Pre-tax CA value (€), estimated based on other fields	all
B_SUBCONTRACTED	Boolean	Whether the contract is likely to be subcontracted	all
DT_AWARD	Date	Date of contract award	all

Table 16. TED fields related to the CA level variables.

Award Value Field AWARD_VALUE_EURO_FIN_1 is an estimation provided when AWARD_VALUE_EURO is empty. The estimation method is the same as for field VALUE_EURO_FIN_1, as described in [7].

D Lexicon

In this section, we give a short definition of the main concepts related to French public procurement, TED, and more generally the DeCoMaP project. The French translation of these expressions is given (in italics) when it appears frequently in the data or documentation.

Acceptance period / ***Période d'acceptation*.** Number of calendar days (after the publication of the notice) available to the Government before for awarding a contract.

Adapted Procedure / ***Marché à procédure adaptée (MAPA).*** Procedure used to award a contract whose estimated value is below the European threshold (see also *Formalized Procedure*).

Agent. Economical entity able to enter into a contract, either as a Client or a Supplier.

French official bulletin of public procurement notices / ***Bulletin Officiel des Annonces des Marchés Publics (BOAMP).*** National outlet used to publish French public procurement contract notices and contract award notices whose estimated value is above a certain national threshold (itself lower than the European threshold).

Contract Authority or Entity (CAE). Agent acting as the client in a public procurement contract.

Central purchasing bodies / ***Centrale d'achat*.** Contracting authority that make contracts on behalf of a CAE.

Contract Award Notice (CAN) / ***Avis d'attribution*.** Document describing the result of the awarding of a contract.

Common Procurement Vocabulary (CPV) / ***Vocabulaire commun pour les marchés*.** European classification system aiming at describing in a normalized way the domain of the product or service that is considered in a public procurement.

Contract Notice / ***Avis de marché*.** Description of a tender opportunity in the public market.

Data Entry Clerk / ***Opérateur de saisie*.** CAE staff in charge of entering the public procurement data into a computer system.

Dynamic Purchasing System / ***Système d'achat dynamique*.** Electronic system used in public procurement, where a supplier can join any time.

Entity / ***Entité*.** In the SIRENE terminology, a high level economical agent, not tied to any geographical zone, and likely to cover one or several lower level economical agents called facilities.

European threshold / ***Seuil européen*.** Depending on whether its estimated value is below or above this threshold, a public procurement must follow the adapted or formalized procedures, respectively.

Facility / ***Établissement*.** In the SIRENE terminology, a low level economical agent, attached to a SIRENE entity, and localized at a specific geographical point.

Formalized Procedure / ***Marché à procédure formalisée*.** Procedure used to award a contract whose estimated value is *above* the European threshold (see also *Adapted Procedure*).

Framework Agreement / ***Accord Cadre*.** Specific type of agreement between some clients and suppliers, allowing to have one or more contracts during a predefined period.

Government Procurement Agreement / ***Accord sur les marchés publics*.** Agreement under the World Trade Organization (WTO), aiming at regulating public procurement.

Joint procurement / ***Groupement conjoint*.** Combined procurement between two or more CAE or suppliers.

Lot. Stand-alone unit of a public procurement, that is assigned separately from the other lots attached to the same contract.

Official Journal of the European Union (OJEU) / ***Journal Officiel de l'Union Européenne (JOUE).*** Outlet to publish contract notices and contract award notices of public procurement contracts with a value above the European threshold.

Open procedure / ***Procédure ouverte.*** Awarding procedure allowing each supplier to submit a bid.

Public entity / ***Entité publique.*** Office or department under the supervision of a local or state government.

Public procurement / ***Marché public.*** Contract concluded for valuable consideration between a public or private buyer and a public or private economic operator.

Restricted procedure / ***Procédure restreinte.*** Awarding procedure in which any supplier can ask to participate, but the client chooses who can submit an offer.

Small and medium-sized enterprises (SME) / ***Petites et moyennes entreprises (PME).*** Companies which employing up to 250 employees.

SIREN code / ***Code SIREN.*** Unique nine digits number used to identify a company or organization in France. SIREN stands for *Système d'Identification du Répertoire des ENtreprises* (Identification system of the entity register).

SIRENE database / ***Base SIRENE (Système national d'Identification du Répertoire des ENtreprises et de leurs Etablissements).*** French database managed by the INSEE (French national institute for statistics) that assigns SIRENs to entities and SIRETs to facilities.

SIRET code / ***Code SIRET.*** Unique 14 digits number containing used to identify a facility in France. It contains the SIREN of the corresponding entity, followed by five digits specific to each facility attached to this entity. SIRET stands for *Système d'Identification du Répertoire des Etablissements* (Identification system of the facility register).

Tenders Electronic Diary (TED). Online version of the OJEU, dedicated to public procurement notices.

Winners / ***Gagnant.*** Economical agent acting as a supplier and receiving a lot award in a public procurement.