Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**

Máster en Ciencia de Datos

Trabajo Fin de Máster

Clasificador de Códigos de Licitaciones Públicas: Caso de Estudio España y México

Autor(a): Alvaro Ramírez Sixtos

Madrid, Julio, 2022

Este Trabajo Fin de Máster se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid.

Trabajo Fin de Máster
Máster en ***Ciencia de Datos***

Título: **Clasificador de Códigos de Licitaciones Públicas: Caso de Estudio España y México**
Julio, 2022

Autor(a): **Alvaro Ramírez Sixtos**

Tutor

Óscar Corcho García

ETSI Informáticos
Departamento de Inteligencia Artificial
Universidad Politécnica de Madrid

Resumen

Las compras gubernamentales representan un 14% del PIB en la Unión Europea y un 5% del PIB en México. En España, los procedimientos de contratación utilizan el estándar de clasificación de artículos CPV (Common Procurement Vocabulary), mientras que en México se emplea el Clasificador Único de las Contrataciones Públicas (CUCoP). Ambos estándares tienen como objetivo la unificación de las referencias utilizadas por los órganos de contratación para describir el objeto del contrato en una licitación. Dichos vocabularios son representados en una estructura jerárquica arborescente de códigos que van de niveles generales a más específicos. El esquema CPV cuenta con cerca de 10,000 términos distintos y CUCoP con más de 15,000, esto hace que en la práctica la asignación de claves sea una tarea compleja por parte de la entidad encargada de la administración de la compra. En este trabajo, se presenta un modelo de clasificación multiclase para cada esquema de clasificación. Además, se emplean técnicas de procesamiento de lenguaje natural para la limpieza y normalización de los textos y métodos que transforman las descripciones textuales a vectores numéricos para entrenar modelos de aprendizaje supervisado. Los modelos emplean como dato de entrada la descripción textual del objeto del contrato en español y clasifican los textos de la siguiente manera: El modelo CPV clasifica un texto en las 45 categorías presentes del nivel más general del esquema CPV, mientras que el modelo CUCoP clasifica el texto en las 30 categorías distintas del segundo nivel jerárquico más general del esquema CUCoP. Los resultados obtenidos son prometedores, ya que los modelos implementados obtienen un *accuracy* de 84% para CPV y un 91% en el caso de CUCoP.

Abstract

Public procurement means 14% of GDP in the European Union and 5% of GDP in Mexico. In Spain, procurement procedures use the standard CPV (Common Procurement Vocabulary), while in Mexico, the Unified Classifier for Public Procurement (CUCoP, Clasificador Único de las Contratacioes Públicas in Spanish) is used. Both standards aim at unifying the references used by contracting authorities to describe the subject of procurement contracts. These vocabularies are based on a tree structure comprising codes that range from general to more specific ones. CPV has about 10,000 different terms and CUCoP has more than 15,000. In practice, the assignment of codes is a complex task for the entity managing the procurement. In this work, we have created a multiclass classification model for each classification scheme. In addition, we used natural language processing techniques for cleaning and normalization of text. We also implemented methods that transform the textual descriptions into numerical vectors in order to train supervised machine learning models. The models use as input the textual description of the subject of procurement contracts in Spanish and classify the texts in the following way: The CPV model classifies a text in the 45 top-level CPV categories, while the CUCoP model classifies a text in the 30 second-level CUCoP categories. Our results obtained are promising since the implemented models obtained an accuracy of 84% for CPV and 91% for CUCoP.

Tabla de contenidos

1 Introducción

Las compras gubernamentales representan un gasto importante en la adquisición de bienes y servicios para el sector público, tan solo en la Unión Europea aproximadamente un 14% del Producto Interno Bruto (PIB) se emplea para este propósito, es decir, alrededor de 2 trillones de euros anuales [1] [2]. En países como México, se estima que alrededor del 22-25% del presupuesto que se destina a las dependencias y entidades del gobierno se utiliza en las contrataciones públicas, un gasto que asciende al 5% del PIB de ese país [3] [4] [5] [6].

Sin embargo, es importante considerar que si existe corrupción en los procesos de contratación el gasto se eleva desde un 20% hasta un 25% [7]. A fin de mitigar esto, los procesos son regulados por legislaciones para permitir que los recursos se inviertan de manera eficiente, de esta manera se garantiza la alta calidad de la prestación de servicios o bienes adquiridos y que se cumplan los objetivos públicos que se hayan establecido. Por ejemplo, en México los procedimientos de contratación pública y adjudicación de contratos se han de llevar a cabo de manera electrónica en el sistema central de contrataciones públicas (CompraNet), estos procesos están regulados bajo el marco jurídico de la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público (LAASSP) y la Ley de Obras Públicas y Servicios Relacionados con las Mismas (LOPSRM) [6]. En España, las compras electrónicas se realizan a través de La Plataforma de Contratación del Sector Público de acuerdo con lo establecido por la Ley de Contratos del Sector Público [8].

Adicional a las regulaciones de los gobiernos, por un lado, con el objetivo de mitigar y prevenir la corrupción en los procesos de contratación y a su vez para mantener los principios de transparencia y rendición de cuentas, los gobiernos como parte de los planes de acción del llamado Gobierno Abierto [9] [10] [11], emplean una estrategia denominada Contrataciones Abiertas [12]. Estrategia que consiste en la publicación y uso de la información abierta y accesible de las contrataciones públicas. Los datos de los procedimientos de contratación son publicados en Portales Abiertos a fin de involucrar a la ciudadanía y al sector privado en el ciclo completo de contratación pública. Puesto que la contratación pública se basa en la competencia abierta para ofrecer el mejor valor por el dinero público.

Teniendo en cuenta que, el acceso a los datos de los procedimientos de contratación es fundamental en las contrataciones públicas. Por una parte, se pretende dar igualdad de oportunidades al sector privado y, por otro lado, garantiza la transparencia del gobierno. Sin embargo, identificar el bien o servicio solicitado de entre todos los procedimientos de contratación que se publican no sería sencillo sin la ayuda de los esquemas de clasificación de artículos [13] [14] que pretenden identificar inequívocamente el bien o servicio requerido por las instituciones públicas. Normalizar los términos que emplean los órganos de contratación al describir el objeto del contrato por medio del uso de códigos contribuye al avance en hacer más eficaz y transparente la contratación pública [14].

Debido a la importancia de los esquemas de clasificación de artículos, en este trabajo se presenta un enfoque que permite a la entidad encargada en la administración de la compra facilitar la labor de asignación del código correspondiente a partir del objeto del contrato de un procedimiento de contratación.

1.1 Planteamiento del problema

A cada proceso de contratación pública se le asigna un código para clasificarlos e identificarlos de un correspondiente esquema de clasificación, de esta manera se organizan y pueden ser ordenados y descubiertos fácilmente por todos involucrados en el proceso de contratación, principalmente por el sector privado. Su importancia, reincide en que permite flexibilizar y facilitar los procedimientos en materia de licitaciones públicas. Por ejemplo, para identificar fácilmente el producto o servicio que una entidad pública requiere contratar y que una compañía que ofrece dicho bien pueda participar en el proceso.

El esquema de clasificación de artículos puede variar de acuerdo con el país donde se emplea. En la Unión Europea, el Vocabulario Común de Adquisiciones (CPV o Common Procurement Vocabulary en inglés) es un estándar que se ha adoptado desde 2008 [14]. Mientras que en México se emplea el Clasificador Único de las contrataciones Públicas (CUCoP) [13], ambos estándares empleados han tenido una amplia aceptación en los procesos de contratación.

Sin embargo, en la práctica, asignar un código a un proceso de contratación no es una tarea sencilla de llevar a cabo y representa varios desafíos. Uno de ellos es que las clasificaciones empleadas en la UE y México son muy extensas ya que cada esquema de clasificación presenta casi 10,000 y 15,000 términos respectivamente. Con tantos elementos, a veces es complicado seleccionar el adecuado y por ello no todas las licitaciones públicas presentan el correcto clasificador.

Por otro lado, y debido a la naturaleza de la estructura de los esquemas de clasificación. Al estar organizados en una estructura jerárquica donde los primeros niveles son términos muy generales, los responsables de asignar el código seleccionan un código que es muy genérico. Lo que conlleva a desaprovechar el potencial de los esquemas de clasificación y no aportan valor al procedimiento.

1.2 Objetivos del proyecto

1.2.1 Objetivos generales

Para solventar los desafíos que se han mencionado en el planteamiento del problema, en este trabajo se presenta el uso de algoritmos de inteligencia artificial como: MultinomialNB, *Support Vector Machines (SVM)* y SGDClassifier para la clasificación automática de un código CPV o CUCoP correspondiente al objeto del contrato de un proceso de contratación público. El alcance de la propuesta asume que se cuenta con la descripción textual del objeto del contrato en idioma español, además se conoce el origen de la licitación, ya sea de España o México, a fin de asignar un único código CPV o CUCoP a la licitación.

En resumen, como principal objetivo de este proyecto es generar un modelo de clasificación de códigos de las licitaciones públicas para cada estándar empleado.

1.2.2 Objetivos específicos

Es complejo llevar a cabo el desarrollo de todo un trabajo teniendo presente un solo objetivo, por lo que para lograr el cumplimiento de éste es necesario establecer objetivos más específicos que puedan ser desarrollados y ejecutados fácilmente. Los objetivos específicos surgen a partir del objetivo principal y son los siguientes:

- Crear un corpus de textos referentes a licitaciones públicas en español obtenido de portales abiertos para códigos CPV y CUCoP. El dato más importante por recabar de una licitación es el objeto del contrato, la descripción textual del bien o servicio que se está solicitando, a su vez, es necesario contar con al menos un código asignado del esquema de clasificación al que pertenezca el procedimiento de contratación.
- Realizar una exploración del corpus, los procedimientos de contratación recogidos de España y México, para identificar la distribución de los códigos asignados y poder establecer una propuesta del nivel jerárquico que se pretende alcanzar para la asignación de los códigos.
- Realizar un preprocesamiento y limpieza del corpus recolectado a fin de que pueda ser utilizado como entrada principal para los modelos de aprendizaje automático que se desarrollen.
- Entrenar modelos clasificación utilizando técnicas de aprendizaje automático y diferentes métodos para codificar la descripción textual de los objetos del contratado que se presentan en el corpus recabado.
- Evaluar y comparar el desempeño de los modelos de clasificación implementados.

1.3 Metodología

Con la finalidad de lograr las metas establecidas por el objetivo principal y específicos es necesario seguir un conjunto de procedimientos que faciliten esas tareas. Por tal motivo, la metodología que se emplea en este proyecto consiste en cuatro fases: Adquisición y Conocimiento de datos, Preparación de datos, Modelado y Evaluación y Resultados. Etapas que han sido extraídas y adaptadas de una de las metodologías más importantes y conocidas en el área de la ciencia de datos, CRISP-DM [15]. Las fases de la metodología CRISP-DM se representan en la ilustración 1 y a continuación se menciona como se han adaptado las etapas a los intereses de este proyecto.

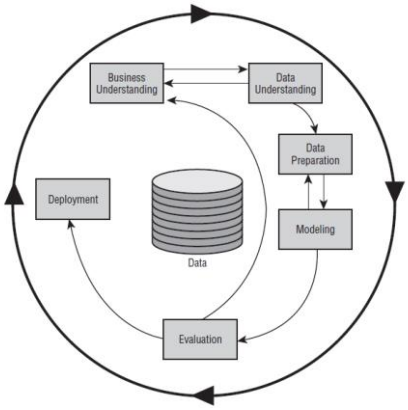

Ilustración 1: Metodología CRISP-DM. Imagen extraída de https://bismart-blog.tumblr.com/post/27199606503/crisp-dm-model-cross-industry-standard-process

En la metodología original, la primera fase denominada conocimiento del negocio, es una de las más importantes, puesto que antes de desarrollar un modelo, se debe entender a la perfección los objetivos del negocio que se pretenden cumplir y a su vez va ligada con la segunda etapa llamada conocimiento de los datos. Dicha fase comprende la familiarización con los datos para descubrir en ellos conocimiento valioso que pueda ser empleado para resolver el problema planteado inicialmente. En este trabajo, se han unido ambas fases en una única y se ha denominado Adquisición y Conocimiento de datos.

La finalidad principal de la etapa de Adquisición y Conocimiento de los datos es la recolección de un corpus que concentre los procedimientos de contratación de España y México con su respectivo esquema de clasificación asociado. Es importante mencionar que el conocimiento de cómo se publican y estructuran los registros en los portales abiertos es imprescindible en esta etapa. Mientras se entiende el estándar y cómo se publican las licitaciones en los portales abiertos, se recuperan los datos que son de vital importancia para este trabajo, como ya se ha mencionado anteriormente, el objeto del contrato y el código CPV o CUCoP.

En la fase de preparación de los datos, se realizan las técnicas necesarias para construir el conjunto final de los datos, mismos que serán empleados como entrada principal para las herramientas de modelado y de aprendizaje automático. Esta fase se mantiene de la metodología original.

La tercera fase, preparación de los datos, está vinculada con la etapa de modelado, puesto que aquí se aplicarán los métodos y técnicas de modelado que sean acordes al problema a resolver, será necesario aplicar requerimientos específicos a los datos dependiendo de la técnica de modelado empleada, por tal

motivo es pertinente regresar a la fase de preparación y realizar ajustes. Al igual que la fase anterior, esta fase también se mantiene de la metodología CRISP-DM.

En la cuarta y última fase de Evaluación y Resultados, se ha de valorar y comparar el desempeño de los modelos que se han construido, es importante evaluar a fondo los pasos que se han seguido para crearlos. Esto permite proporcionar calidad suficiente, desde una perspectiva de análisis de datos, al modelo final que en teoría resuelve el problema. La evaluación del modelo o modelos no es la etapa final de este proyecto, puesto que se han de mostrar los resultados del porqué se han seleccionado dichos modelos.

2 Estado del Arte

2.1 Contrataciones Públicas y Gobierno Abierto

Para poder concebir este proyecto, es necesario comprender las bases del conocimiento sobre las que se ha construido, para ello en los siguientes apartados se ahonda en definiciones y conceptos que se consideran importantes para que el lector pueda visualizar en su totalidad el panorama general de este trabajo.

En el ámbito del Sector Público, las contrataciones públicas o compras gubernamentales se refieren al proceso por el cual el gobierno o autoridades adquieren bienes, obras y servicios de compañías. De acuerdo con La Organización para la Cooperación y el Desarrollo Económicos (OCDE) la contratación pública se refiere a la compra por parte de los gobiernos y las empresas estatales de bienes, servicios y obras [16]. Las compras gubernamentales de los países miembros de la OCDE, entre los que se incluyen España y México, destinan el 12% del PIB para estos fines y varían desde 4.9% en México hasta el 19.5 en Países Bajos [17] [3] [7] [4].

Es inevitable que al hablar de compras públicas no se mencione el término de licitación pública, acorde con la Secretaría de la Función Pública (SPF), institución que tiene la facultad de coordinar las políticas públicas en materia de control interno y evaluación de la gestión gubernamental en México, esta dependencia define una licitación pública como un procedimiento de contratación en que a través de una declaración unilateral de voluntad contenida en una convocatoria pública, el Estado se obliga a celebrar un contrato para la adquisición de un bien o servicio -incluida obra pública-, con aquél interesado que cumpliendo determinados requisitos prefijados en la convocatoria por el ente público de que se trate, ofrezca al Estado las mejores condiciones de contratación. Dicho procedimiento se encuentra abierto a todos aquellos interesados que reúnan los requisitos previstos, de ahí que la licitación pública sea un procedimiento cuya esencia se encuentra en la competencia [18].

Se podría pensar que una licitación pública solamente se desempeña para satisfacer las necesidades del Sector Público y se limita a la adquisición de bienes y servicios. Sin embargo, es una herramienta muy importante y tiene que estar regulada por leyes para maximizar el valor del dinero que es destinado por el Estado para alcanzar los objetivos y fines de las instituciones gubernamentales y de la sociedad. La Comisión Europea, a través de legislaciones, garantiza el cumplimiento de tres principios fundamentales:

- Principio de Igualdad de trato.
- Principio de no discriminación.
- Principio de transparencia.

Es mediante el principio de transparencia, que se establecen normas y directrices para publicar periódicamente información de los procedimientos de contratación públicos a fin de garantizar y asegurar la transparencia de las licitaciones relacionadas con las Administraciones Públicas. Como consecuencia de proveer información libre y de acceso gratuito se incrementa el mercado de la transparencia, disminuyen los costos de transacción y facilita la rendición de cuentas por parte del gobierno.

La cultura de gobernanza que incita a los gobiernos a promover los principios de integridad, transparencia y rendición de cuentas es denominado Gobierno Abierto [12] y se han establecido organizaciones multilaterales integrados por

reformadores de las Administraciones públicas, entre ellas, La Alianza para el Gobierno Abierto (OGP por sus siglas en inglés) [11] para conseguir dicha cultura. Además, se han establecido planes de acción [9] [10] entre los miembros de la organización OGP en las que se establecen compromisos para avanzar en temas de Gobierno Abierto. Como es el caso de México y España quienes son países miembros fundadores de la alianza OGP en 2011 y desde entonces han mejorado sus políticas para avanzar en este aspecto.

2.2 Contrataciones Abiertas y Datos Abiertos

Las contrataciones abiertas surgen de la cultura de Gobierno Abierto, fomentando la transparencia y rendición de cuentas en materia de las contrataciones públicas. A través de dicha cultura, los gobiernos han puesto en marcha la creación de portales de Datos Abiertos.

La Alianza para el Gobierno Abierto también promueve la apertura de las contrataciones públicas. Adicional a esta, surge igualmente la Alianza para las Contrataciones Abiertas (OCP por sus siglas en inglés), una colaboración entre distintos gobiernos cuya misión es impulsar una norma global que busca contrataciones públicas mejores y abiertas.

2.2.1 ¿Qué es la Contratación Abierta?

La Alianza para las Contrataciones Abiertas (OCP) define a las Contrataciones Abiertas como la divulgación y uso de información abierta, accesible y oportuna sobre las contrataciones del gobierno, para lograr que los ciudadanos y las empresas puedan participar, con el fin de identificar problemas y solucionarlos [19].

La Contratación Abierta cobra sentido teniendo en cuenta que las contrataciones públicas son clave en las actividades económicas de las administraciones públicas y están particularmente expuestas a una mala gestión, fraude y corrupción [12]. No es de extrañar que las contrataciones públicas sean la actividad del gobierno con el mayor riesgo de corrupción. Por ejemplo, en la Unión Europea se estima que en costos directos de corrupción se gasta 120 mil millones de euros anuales, el 1% del PIB [20]. Al tener costos significativos, la Contratación Abierta es muy útil para saber que el uso de los recursos de los gobiernos se emplea de manera íntegra y conforme a objetivos oficiales establecidos.

Además de combatir la corrupción, otro beneficio que ofrece la apertura de los datos relativos a las contrataciones públicas es ofrecer una competencia más justa e igualdad de condiciones para el sector privado, empresas que sin importar su tamaño o volumen puedan ofrecer sus bienes o servicios al gobierno [12]. Esto garantiza suministrar a los ciudadanos servicios de calidad al obtener precios justos y razonables de los proveedores.

2.2.2 Portales de Datos Abiertos

Se podría decir que los Portales de Datos Abiertos forman parte fundamental para la apertura de datos, estas son plataformas digitales donde se almacena y comparte bases de Datos Abiertos. Se entiende como Datos Abiertos los datos que pueden ser utilizados, reutilizados y redistribuidos libremente por cualquier persona, y que se encuentran sujetos, cuando más, al requerimiento de atribución y de compartirse de la misma manera en que aparecen. Definición

extraída de la Fundación Conocimiento Abierto (Open Knowledge Foundation, OKF) [21].

En este sentido, los gobiernos brindan acceso a los distintos datos que generan y administran para que los datos sean utilizados, reutilizados y redistribuidos. En España los Portales de Datos Abiertos cobran interés en el contexto de la iniciativa Aporta [22], que surge en 2009 con la finalidad de promocionar la apertura de la información pública y desarrollo de servicios avanzados basados en datos. El Ministerio de Asuntos Económicos y Transformación Digital es el encargado de promover esta iniciativa. Mientras que, en México, como parte de los compromisos establecidos en la Alianza para el Gobierno Abierto [23] y la iniciativa de Datos Abiertos [24], la información pública del gobierno se publica a través de la plataforma oficial del Gobierno de la República, datos.gob.mx [25], los datos se incorporan en dicho sitio web de manera gradual.

Cabe mencionar que, a nivel global varios países también han adoptado la cultura que incita a promover sus datos abiertos. Por ejemplo, Data.gov [26] de Estados Unidos y Data.gob.uk [27] de Inglaterra por mencionar algunos.

En lo que concierne a este proyecto, el interés principal es en obtener acceso a información relativa de las contrataciones públicas en portales de Datos Abiertos que las administraciones públicas de México y España han puesto a disposición de la ciudadanía.

2.2.3 Plataforma de Contrataciones Abiertas en España

La distribución de las licitaciones públicas en España se lleva a cabo mediante el portal oficial de contrataciones públicas denominado Plataforma de Contratación del Sector Público (PLACSP) [28]. Dicha plataforma es el principal punto de acceso a la información sobre la actividad de las contrataciones públicas del Sector Público en España y es puesta a disposición de todos los órganos de contratación del sector público para permitir publicar y dar a conocer los resultados de sus convocatorias de licitación. De este modo, facilita el proceso de contratación electrónica, ya que actúa como medio de comunicación entre los organismos públicos y el sector privado. Además de proporcionar y distribuir los datos referentes a las convocatorias de licitación, en la plataforma también se puede encontrar la documentación de todo el proceso de contratación.

Asimismo, al ser una plataforma de titularidad del Ministerio de Hacienda y Función Pública, dicha institución en cumplimiento con las obligaciones de publicidad que establece la Ley de Contratos del Sector Público [8] a través del Portal de datos abiertos del Ministerio de Hacienda [29] apertura datos para su reutilización relativa a las licitaciones públicas de la plataforma PLACSP. En ese portal se pueden descargar ficheros de datos comprimidos por año con las actualizaciones producidas desde el 1 de enero de 2012 o por meses del año en curso de las licitaciones publicadas en PLACSP.

Además de proporcionar la información de las licitaciones, el Ministerio de Hacienda proporciona herramientas como OpenPLACSP [30], con licencia de software libre EUPL 1.2 [31], para facilitar la transformación de los datos abiertos en documentos de hojas de cálculo con los principales datos, objeto del contrato y código CPV, de las licitaciones.

Otros sitios donde también se puede encontrar licitaciones públicas en español que utilicen el estándar CPV, son las plataformas:

- TED: Licitaciones Electrónicas Diarias o (Tenders Electronic Daily - TED en inglés) portal que es considerado como la piedra angular de la contratación pública en Europa [32].
- Data.europa.eu: El portal oficial de datos abiertos europeos [33].

2.2.4 Plataforma de Contrataciones Abiertas en México

En México, CompraNet [34] es la plataforma electrónica de información pública gubernamental en materia de contrataciones públicas y de uso obligatorio por la Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público (LAASSP) y la Ley de Obras Públicas y Servicios Relacionados con las Mismas (LOPSRM) [6]. Dicho sistema electrónico es operado por la Unidad Política de Contrataciones Públicas (UPCP) de la Secretaría de la Función Pública. El objetivo principal es permitir a las instituciones públicas realizar procedimientos de contratación de manera electrónica. Todos los involucrados en los procedimientos de contratación, en especial los proveedores o contratistas pueden acceder a las licitaciones públicas y enviar sus proposiciones por ese medio de manera segura. Tan solo en 2017, en CompraNet, se registró un total de 208,386 contratos que equivalen a un gasto de más de 547 mil millones de pesos [3].

Como fin de promover la iniciativa de Datos Abiertos, en CompraNet se puede consultar datos de los procedimientos de contratación que realizan las instituciones públicas que reciben recursos públicos con el objetivo de transparentar las contrataciones públicas. Adicional a ese sistema electrónico, y como compromiso establecido en la Alianza para las Contrataciones Abiertas MX [23] en 2017 se presentó la plataforma de Contrataciones Abiertas [35] donde se puede encontrar datos relevantes de los procedimientos de contratación como: quién contrató; a quién y para qué; las fechas y montos de los contratos adquiridos.

2.3 Estándar para Contrataciones Abiertas

Cabe mencionar que la apertura de la información de datos relativos a los procedimientos de contratación es crucial para apegarse a la cultura de las contrataciones abiertas. Sin embargo, es necesario que dichos datos sean publicados de manera estructurada y estandarizada a fin de ofrecer mayor calidad y puedan ser reutilizados por terceros fácilmente. Los gobiernos han adoptado estándares que se ajustan a sus necesidades para la publicación de sus datos.

2.3.1 Proyecto CODICE

El proyecto CODICE [36], Componentes y Documentos Interoperables para la Contratación Electrónica, es un proyecto que se desarrolló para garantizar un estándar en la creación de estructuras para los sistemas informáticos en el proceso de las contrataciones electrónicas. Este proyecto fue desarrollado por el Ministerio de Economía y Hacienda. CODICE es empleado para publicar los procedimientos de contratación por la plataforma de contrataciones abiertas en España. La versión más actual corresponde a CODICE 2.06.

De acuerdo con la guía de implementación publicada por el Ministerio de Hacienda [37], CODICE dispone de una estructura de arquitectura de

documentos y componentes que facilita a las aplicaciones la reutilización de código para generar documentos con información común.

A continuación, se muestran los principales documentos que son generados en los procedimientos de contratación electrónica empleando el estándar CODICE 2.0.

Tabla 1: Extracto de la tabla de guía de implementación de documentos CODICE 2.0

Documento CODICE	Nombre	Descripción
PriorInformationNotice	Anuncio de información previa	Anuncio por el que da publicidad a la intención de adjudicar contratos de obras, servicios o suministros durante los próximos 12 meses.
ContractNotice	Anuncio de licitación	Anuncio por el que da publicidad a la convocatoria de la licitación.
Call For Tenders	Pliegos	Documentación en la que se establece todas las condiciones de la licitación, el objeto y condiciones del contrato, las condiciones de participación, y las condiciones de adjudicación.
TendererQualification	Documentación administrativa	Documentación presentada por el licitador para acreditar el cumplimiento de los requisitos de participación en el procedimiento.
Guarantee	Garantía	Documento que permite constituir una garantía provisional o definitiva en forma de aval o contrato de seguro de caución.
Tender	Oferta	Documentación que contiene las condiciones ofertadas por el licitador.
AwardingNotification	Notificación de adjudicación	Notificación que el órgano de contratación emite a los participantes en el procedimiento para informarles sobre el resultado del mismo.

2.3.2 Estándar EDCA

De acuerdo con la Alianza para las Contrataciones Abiertas (OCP), el Estándar de Datos para las Contrataciones Abiertas (OCDS/EDCA), es un estándar de datos abierto, gratuito y no protegido por derechos de propiedad intelectual para la contratación pública y que ha sido implementado por más de 30 gobiernos en todo el mundo. Dicho estándar describe cómo publicar datos y documentos de las etapas del proceso de contratación.

En México se adoptó el Estándar de Datos de Contrataciones Abiertas (EDCA) para todos los contratos modelo de adquisiciones gubernamentales en los niveles central y local en el portal de Contrataciones Abiertas [35]. Según la OCP, la Ciudad de México es la primera ciudad del mundo en publicar información de los procedimientos de contratación pública en sus distintas etapas: planificación, licitación, adjudicación e implementación mediante el estándar EDCA. Sin embargo, en CompraNET, la información actualmente publicada no se proporciona en el estándar como lo requiere el EDCA [6].

OCDS describe una forma para modelar y publicar datos de manera estandarizada para el proceso completo de las contrataciones públicas. A nivel general, la información es representada en estructuras de datos comunes. Estos datos son mapeados en los siguientes esquemas principales [38]:

- **Metadatos de la entrega**: información contextual sobre cada entrega de datos.
- **Partes involucradas**: información sobre las organizaciones y los participantes en el proceso de contratación.
- **Planeación**: información sobre los objetivos, presupuestos y proyectos a los que se refiere un proceso de contratación.
- **Licitación**: información sobre la forma en que tendrá lugar la licitación o como se ha realizado.
- **Adjudicación**: información sobre las adjudicaciones otorgadas como parte de un proceso de contratación.
- **Contrato**: información sobre contratos firmados como parte de un proceso de contratación.
- **Implementación**: información sobre el progreso de cada contrato hasta su finalización.

2.4 Esquema de clasificación de Artículos

Si bien los bienes, servicios y obras que requiere una administración pública son diversos y podrían generar confusión entre los participantes de las contrataciones públicas. Por ello, es necesario el uso de un esquema de clasificación de artículos que se desempeñe como referencia de los bienes y servicios a adquirir.

2.4.1 Esquema CPV - Vocabulario Común de Adquisiciones

El Vocabulario Común de Adquisiciones (CPV por sus siglas en inglés) es un estándar de uso obligatorio adoptado por la Comisión de la Comunidad Europea y consiste en un vocabulario principal para definir el tema de un contrato y un vocabulario complementario para agregar información cualitativa adicional [14].

El estándar CPV, se basa en una estructura de árbol que comprende códigos de hasta 9 dígitos (código de 8 dígitos más uno de control) asociado con una redacción que describe el tipo de bienes, obras o servicios que forman el tema del contrato. El código de control puede omitirse.

El código numérico incluye ocho dígitos y se subdivide en:

- **Divisiones**, identificadas por los dos primeros dígitos del código (XX000000).
- **Grupos**, identificados por los tres primeros dígitos del código (XXX00000)
- **Clases**, identificadas por los cuatro primeros dígitos del código (XXXX0000)
- **Categorías**, identificadas por los cinco primeros dígitos del código (XXXXX000-Y).

A modo de ejemplo, el código 90911100, servicios de limpieza de viviendas. Pertenece a la división 90000000, servicios de salud y asistencia social; al grupo 90900000, servicios sanitarios y de limpieza; a la clase 90910000, servicios de

limpieza y la categoría 90911000, servicios de limpieza de viviendas, edificios y ventanas. Tal como se muestra en la siguiente ilustración.

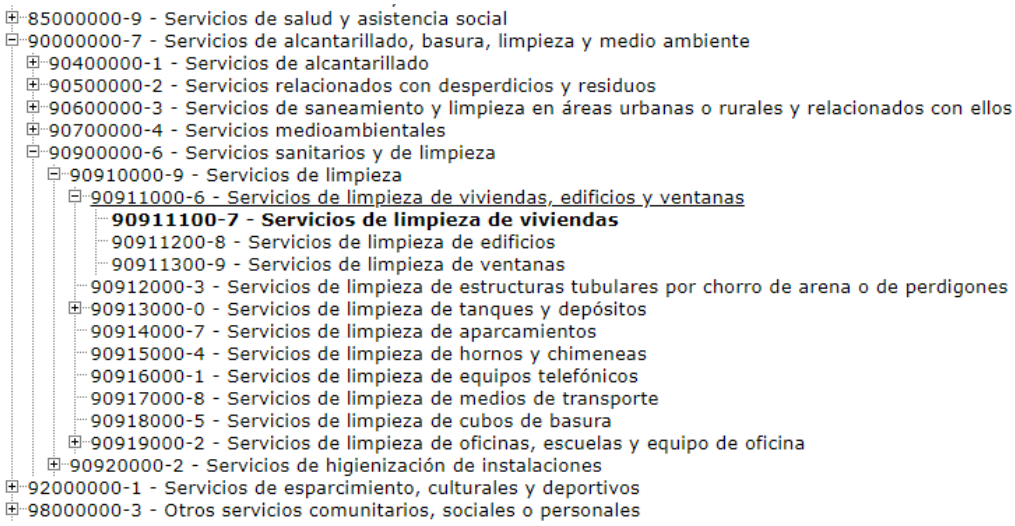

Ilustración 2: Estructura jerárquica del código 90911100. http://www.cpv.enem.pl/es/90911100-7

2.4.2 CUCOP – Clasificador Único de Contrataciones Públicas

El Clasificador Único de las Contrataciones Públicas es un esquema que solo se emplea en México para la clasificación de bienes, trabajos, servicios y rentas [39]. Este esquema se basa en los códigos creados por la Secretaría de Hacienda y Crédito Público: los clasificadores por objeto del gasto (COGs [40]) para estructurar el gasto público en México [6]. El objetivo principal de este esquema de clasificación consiste en unificar y estandarizar los criterios técnicos que emplean las entidades de contratación en la descripción del objeto del contrato de los procedimientos de contratación.

Al igual que el esquema CPV, CUCoP presenta una estructura jerárquica y comprende códigos de 8 dígitos que se distribuye en los siguientes niveles de desagregación.

- Capitulo. Código compuesto por 4 dígitos, corresponde al capítulo del COG.
- Concepto. Código compuesto por 4 dígitos, corresponde al concepto del COG.
- Partida Genérica. Código compuesto por 4 dígitos, corresponde a la partida genérica del COG.
- Partida Específica. Código compuesto por 5 dígitos, corresponde a la partida específica del COG.

Una clave CUCoP está compuesta por un código de 5 dígitos que representa el nivel de partida específica, pero solamente se toman los primero 4 dígitos de ésta y 3 dígitos adicionales como números consecutivos. A modo de ejemplo se emplea la clave 21100001, abrecartas, para mostrar las partes fundamentales que lo integran.

21100 001: El primer dígito codifica el capítulo. En este caso corresponde al capítulo 2000, materiales y suministros.

21100 001: Los dos primeros dígitos codifican el concepto. 21 corresponde al concepto 2100, materiales de administración, emisión de documentos y artículos oficiales.

21100001: Los tres primeros dígitos representan el nivel de partida genérica. En este caso, la partida genérica corresponde a 2110, materiales y equipos menores de oficina.

21100001: Los primeros cuatro dígitos codifican la partida específica sin su último digito, en este caso corresponde a la partida específica 21101, materiales y útiles de oficina. Los 4 dígitos siguientes (0001) corresponden al número consecutivo, en este caso es el primer artículo de esta categoría.

En la siguiente ilustración se muestra de manera visual las categorías a las que está asignada dicha clave CUCoP.

CLAVE CUCoP	PARTIDA ESPECÍFICA	CLAVE CUCoP +	DESCRIPCIÓN	NIVEL
2000		2000	Materiales y suministros	1
2100	2000	2100	Materiales de administración, emisión de documentos y artículos oficiales	2
2110	2100	2110	Materiales, útiles y equipos menores de oficina	3
21101	2110	21101	Materiales y útiles de oficina	4
21100001	21101	21101-0001	Abrecartas	5

Ilustración 3: Estructura jerárquica de la clave CUCoP 21100001

Cabe mencionar que, en los últimos años se ha planteado un ajuste al esquema CUCoP con la finalidad de representar completamente la categoría del nivel de partida especifica. De esta manera, se ha propuesto un nuevo esquema denominado CUCoP+ (léase como cucop plus) que utiliza una estructura de 9 dígitos, donde los primeros 5 dígitos corresponden a la partida específica y los siguientes 4 al número consecutivo. En el caso de la clave CUCoP 21100001 su correspondiente clave CUCoP+ sería 21101-0001. Este nuevo esquema todavía no ha sido adoptado en las contrataciones públicas del gobierno de México, pero se espera que pronto lo sea.

2.5 Trabajos relacionados con este proyecto

A continuación, se describen trabajos relacionados respecto a modelos de clasificación de códigos CPV.

2.5.1 MKaan/multilingual-cpv-sector-classifier

El trabajo desarrollado por Kaan Görgün, es un modelo de clasificación multi-idioma [41] que emplea como conjunto de datos de entrenamiento los objetos del contrato de los procedimientos de contratación que son publicados en la plataforma TED [32] y los clasifica en las 45 divisiones presentes en CPV. El modelo es una versión ajustada del modelo pre-entrenado BERT-base-multilingual-cased [42] y en general, obtiene un F1-score de 0.686.

Para entrenar el modelo, el autor ha empleado un total de 744,360 registros que ha dividido en 80% para *training* y 20% para *testing*. Dichos registros corresponden a licitaciones públicas de la Unión Europea del periodo 2011 al 2018.

Al ser un modelo multi-idioma, el texto de entrada puede ser ingresado en cualquiera de los distintos 104 idiomas que soporta el modelo base BERT. Incluyendo el idioma español. Sin embargo, el autor hace mención que el modelo solamente ha sido evaluado en 22 idiomas y no presenta información del desempeño en los idiomas que no están incluidos. El desempeño del modelo en idioma español obtiene un F1-Score de 0.64 utilizando un conjunto total de 7,483 registros para su evaluación.

Dicho modelo clasifica en las 45 categorías presentes en el estándar CPV del primer nivel jerárquico, sin embargo, la clasificación es realizada con la descripción textual y no con el código CPV asociado. Aunque no sería complicado convertir de uno a otro.

El modelo se encuentra disponible en línea [41] para que cualquier interesado pueda hacer uso de este. Se ha realizado una prueba empleando un objeto del contrato real para comprobar el desempeño del modelo.

Tabla 2: Ejemplo de objeto del contrato de un procedimiento de contratación y su clasificación CPV

Código CPV	90919200
División	Servicios de alcantarillado, basura, limpieza y medio ambiente
Objeto del contrato	Servicio de limpieza de las Oficinas de Extranjeros de la Avenida Constitución 106-108 y 116 de Valencia, dependientes de la Delegación del Gobierno en la Comunidad Valenciana.

El resultado arroja con un 0.99 de certeza que la clasificación corresponde a *Sewage-, refuse-, cleaning-, and environmental services,* que en español es Servicios de alcantarillado, basura, limpieza y medio ambiente siendo la categoría correcta a la que pertenece el objeto del contrato.

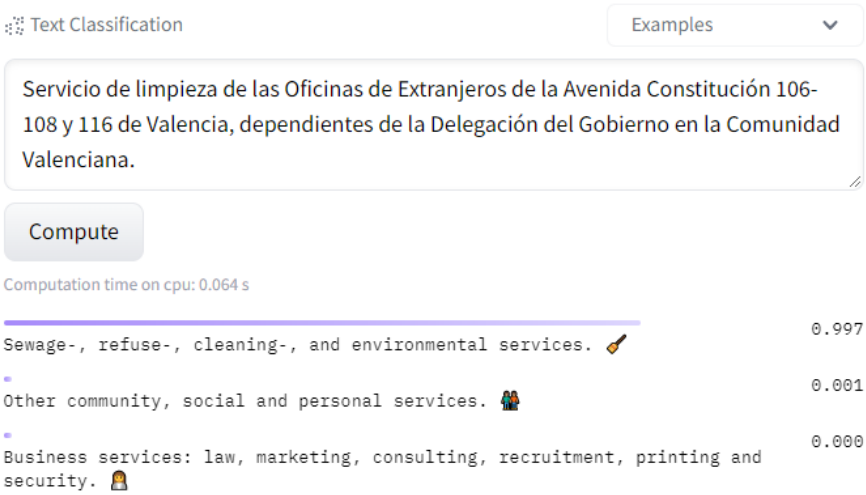

Ilustración 4: Prueba del modelo MKaan/ multilingual-cpv-sector-classifier utilizando un objeto del contrato en español

2.5.2 Multi-label Text Classification for Public Procurement in Spanish

Es un artículo científico [43] publicado por el Grupo de Ingeniería Ontológica de la Universidad Politécnica de Madrid en la 38ª conferencia de la sociedad española para el procesamiento del lenguaje natural. En dicho artículo, se presentan modelos de clasificación multi-etiqueta para clasificar objetos del contrato en las 45 categorías principales del estándar CPV, los modelos fueron entrenados utilizando las descripciones textuales en idioma español de los procedimientos de contratación del 2019 que el Ministerio de Hacienda pública en su portal de datos abiertos. Los autores mencionan que su enfoque fácilmente puede adaptarse a otros idiomas.

Para el modelado, los autores emplean diferentes algoritmos clásicos de clasificación como Naïve Bayes, KNN, DecisionTree, RandomForest y SVM. Siendo este último el modelo que alcanza un F1-score de 0.69 sobrepasando el modelo propuesto por Kaan Görgün que alcanza un 0.64.

Además, este grupo de expertos realizaron modelos afinando la versión del modelo MarIA [44], un modelo basado en RoBERTa transformed-based [45] que ha sido entrenado en un corpus de la Biblioteca Nacional de España. El mejor de estos modelos alcanzo un F1-score de 0.80, superando los anteriores modelos.

2.6 Procesamiento de Lenguaje Natural

En este apartado se muestran las principales técnicas para la limpieza de conjuntos de datos basados en textos. Este paso es crucial para omitir caracteres y palabras innecesarias a los modelos de clasificación.

2.6.1.1 Lowercasing

Dado que los documentos constan de muchas oraciones, las palabras escritas en mayúsculas pueden generar problemas al clasificar documentos grandes. El enfoque más común para lidiar con mayúsculas consistente en transformar cada letra a minúsculas. Esta técnica convierte todas las palabras en el texto a una nueva representación, lo que podría causar un problema importante en la interpretación de algunas palabras. Por ejemplo, la palabra USA se convierte en usa que podría ser la conjunción del verbo usar en tercera del singular.

2.6.1.2 Tokenization

La *tokenization* es un método de preprocesamiento que divide un texto completo en palabras, frases, símbolos u otros elementos significativos que lo componen llamados *tokens* [46]. El objetivo principal de este paso es la obtener las palabras que conforman una sentencia.

2.6.1.3 Noise removal

La gran mayoría de textos en conjuntos de datos contienen muchos caracteres innecesarios tales como signos de puntuación y caracteres especiales. Estos símbolos son importantes para el entendimiento del mensaje para los humanos, sin embargo, en el caso de algoritmos de clasificación podrían ser innecesarios.

2.6.1.4 Stop-word removal

Los documentos de texto incluyen palabras que no contiene significado importante que sean útiles para los algoritmos de clasificación, tales como: a, acerca, para, con, desde, contra, etc. La técnica más común para liderar con dichas palabras es eliminarlas de los textos y documentos [47].

2.6.1.5 Lemmatization

La lematización es un proceso que reemplaza el sufijo de una palabra por otra diferente o elimina el sufijo de una palabra completamente para obtener la forma básica de la palabra denominada lema.

2.6.2 Técnicas de Vectorización de Texto

Generalmente los documentos y textos se concentran en conjuntos de datos no estructurados. Sin embargo, para que estos datos puedan ser procesados por computadores se tiene que codificar a vectores numéricos. Las técnicas comunes para la vectorización o extracción de características son las que se mencionan a continuación [48].

2.6.2.1 CountVectorizer

Conocido también como bolsa de palabras o bag-of-words en inglés, es la técnica más simple para la extracción de características (*feature extraction*). Este método se basa en contar el número de palabras en cada documento y asignarlas a una representación vectorial.

2.6.2.2 TF-IDF

K, Sparck Jones [49] propuso la matriz inversa de documentos (IDF) como método para ser usado en conjunto con la matriz de frecuencia de términos (TF) para disminuir el efecto de las palabras más comunes en el corpus. IDF asigna un peso más grande a las palabras con términos de baja frecuencia en el documento. La combinación entre TF e IDF es conocida como TF-IDF (del inglés Term frequency – Inverse document frequency).

2.6.2.3 Word2Vect

La representación de palabra a vector (Word2Vect) fue introducida por T. Mikolov [50] como una mejora en la arquitectura de vectorización de palabras. La técnica Word2Vect utiliza redes de neuronas superficiales con dos capas ocultas, bag-of-words continuas y el modelo skip-gram para crear un vector de mayor dimensión para cada palabra. El modelo skip-gram, se utiliza para mantener la información sintáctica y semántica de las oraciones para los algoritmos de aprendizaje automático.

2.6.2.4 Sentence embedings SBERT

La representación de las sentencias es mapeada en vectores de números reales [51] utilizando la oración completa en la vectorización. Esta técnica de codificación representa oraciones completas y mantiene la información semántica en vectores. Esto ayuda a la máquina a comprender el contexto, la intención y otros matices en todo el texto.

2.6.3 Algoritmos de Inteligencia Artificial utilizados

En este apartado se presentan los algoritmos de aprendizaje automático que se emplearan en el modelado de este proyecto.

2.6.3.1 MultinomialNB

Con los clasificadores bayesianos se intenta construir modelos probabilísticos basado en el modelado de las características subyacentes de las palabras en las clases. La idea general es clasificar un texto basado en la probabilidad posterior de que los documentos pertenezcan a las diferentes clases en función de la presencia de palabras en los documentos [52].

En el caso del modelo Multinomial, se captura las frecuencias de los términos en un documento representando un documento con un Bag of Words. Los documentos de cada clase se pueden modelar como muestras extraídas de una distribución de palabras multinomial. Como resultado, la probabilidad condicional de un documento dada una clase es simplemente un producto de la probabilidad de cada palabra observada en la clase correspondiente.

2.6.3.2 SVM

Los clasificadores SVM intentan dividir el espacio de datos con el uso de delimitadores lineales o no lineales entre las diferentes clases. La clave de tales clasificadores es determinar los límites óptimos entre las diferentes clases y utilizarlos con fines de clasificación [52].

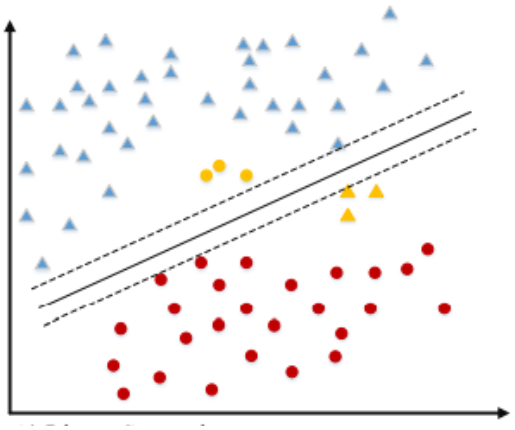

Ilustración 5: SVM Lineal. Imagen obtenida del algoritmo SVM en [48]

2.6.3.3 SGDClassifier

Es un algoritmo de clasificación [53] que emplea clasificadores lineales como SVM, Regresión logística, etc con entrenamiento SGD (Stochastic gradient descent). SGD se ha aplicado con éxito a problemas de aprendizaje automático de gran escala que se encuentran a menudo en la clasificación de textos. Dado que los datos son dispersos, los clasificadores de este módulo escalan fácilmente a problemas con más de 10^5 ejemplos de entrenamiento y más de 10^5 características.

3 Desarrollo

En este capítulo se presenta el flujo de trabajo que se ha llevado a cabo para cada una de las fases establecidas en la metodología a fin de completar los objetivos del presente trabajo.

En la primera sección, se detallan las actividades realizadas para recopilar los registros de datos referentes a licitaciones públicas de México y España. Se describen las principales fuentes para la obtención de los datos de las contrataciones públicas. Además, se refiere las estrategias empleadas para almacenar los registros de las licitaciones y los esquemas de clasificación de bienes. Asimismo, se representan gráficamente los datos para descubrir información que podría ser relevante para este trabajo. Posteriormente, se presentan los métodos empleados para la limpieza y procesamiento de los datos. Finalmente, en el último apartado se muestra modelos entrenados a partir de los datos y el desempeño de cada uno de ellos.

Los conjuntos de datos, el código fuente y *notebooks* empleados en este proyecto se encuentran alojados en el repositorio principal localizado en la siguiente liga: alvaroame/TFM: Trabajo de Fin de Máster (MSc in Data Science) (github.com)

3.1 Adquisición de los datos

Teniendo en cuenta que, el objetivo principal es realizar modelos de clasificación de códigos de licitaciones públicas, para ello es necesario obtener una cantidad considerable de registros que sirvan como entrada para entrenar algoritmos de clasificación. Debido a que, en México y España, en las licitaciones públicas se emplean distintos esquemas de clasificación de artículos, CUCoP en México y CPV en España. Por lo anterior, se pretende realizar un modelo para cada esquema.

Por otro lado, y al objeto de este proyecto, los principales datos y mínimos necesarios que se pretende recolectar por cada licitación pública son: el objeto del contrato, que es la descripción del bien, obra o servicio que la entidad pública solicita o requiere y el código del esquema de clasificador empleado en la licitación, CUCoP o CPV.

Al tener que generar dos conjuntos de datos, uno para cada esquema de clasificación, se opta por recolectar los registros de las fuentes oficiales de portales abiertos de México y España, a continuación, se menciona el proceso que se ha empleado para la adquisición de los datos.

3.1.1 Recolección de datos de España

La recolección de los datos necesarios de las licitaciones públicas de España se ha de realizar en el Portal de Datos Abiertos del Ministerio de Hacienda. En ese sitio, se publican anual y mensualmente, para los meses del año en curso, los registros de las licitaciones públicas que se ingresan en la Plataforma de Contratación del Sector Público. Estos registros son publicados en ficheros comprimidos que pueden ser descargados gratuitamente y ser examinados posteriormente sin conexión a Internet.

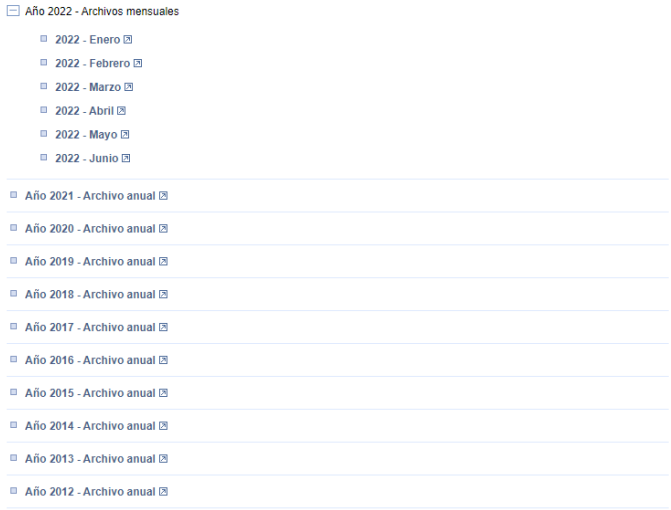

Ilustración 6: Ficheros comprimidos de licitaciones públicas en PLACSP.

Al descomprimir cada fichero, estos presentan un conjunto de archivos con extensión ATOM y por lo tanto se encuentran en formato XML. Se podría decir que, al estar en un formato estructurado, los archivos son de fácil lectura por ordenador con un programa informático y a su vez permite la extracción de datos fácilmente. Sin embargo, la lectura de estos archivos no es trivial y tiene que ser de manera organizada. Cada archivo, además de contar con registros de licitaciones proporciona información referente al siguiente archivo a leer, es decir cada archivo se encuentra enlazado con el siguiente.

Nombre	Fecha de modificación	Tipo	Tamaño
licitacionesPerfilesContratanteCompleto3.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,542 KB
licitacionesPerfilesContratanteCompleto3_20160513_151003.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,396 KB
licitacionesPerfilesContratanteCompleto3_20160513_151003_1.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,274 KB
licitacionesPerfilesContratanteCompleto3_20160513_160738.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,346 KB
licitacionesPerfilesContratanteCompleto3_20160513_160738_1.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,345 KB
licitacionesPerfilesContratanteCompleto3_20160513_160738_2.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,687 KB
licitacionesPerfilesContratanteCompleto3_20160513_170827.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,643 KB
licitacionesPerfilesContratanteCompleto3_20160513_170827_1.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,278 KB
licitacionesPerfilesContratanteCompleto3_20160513_181010.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,120 KB
licitacionesPerfilesContratanteCompleto3_20160513_181010_1.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,693 KB
licitacionesPerfilesContratanteCompleto3_20160513_181010_2.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,771 KB
licitacionesPerfilesContratanteCompleto3_20160513_191104.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,530 KB
licitacionesPerfilesContratanteCompleto3_20160513_191104_1.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,152 KB
licitacionesPerfilesContratanteCompleto3_20160513_191104_2.atom	28/02/2018 09:49 a. m.	Archivo ATOM	6,652 KB
licitacionesPerfilesContratanteCompleto3_20160513_200857.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,137 KB
licitacionesPerfilesContratanteCompleto3_20160513_200857_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,369 KB
licitacionesPerfilesContratanteCompleto3_20160513_200857_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,792 KB
licitacionesPerfilesContratanteCompleto3_20160513_211046.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,223 KB
licitacionesPerfilesContratanteCompleto3_20160513_211046_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,642 KB
licitacionesPerfilesContratanteCompleto3_20160513_211046_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,553 KB
licitacionesPerfilesContratanteCompleto3_20160513_221409.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,359 KB
licitacionesPerfilesContratanteCompleto3_20160513_221409_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	5,954 KB
licitacionesPerfilesContratanteCompleto3_20160513_231406.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,170 KB
licitacionesPerfilesContratanteCompleto3_20160513_231406_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,468 KB
licitacionesPerfilesContratanteCompleto3_20160513_231406_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,904 KB
licitacionesPerfilesContratanteCompleto3_20160514_152957.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,302 KB
licitacionesPerfilesContratanteCompleto3_20160514_152957_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,294 KB
licitacionesPerfilesContratanteCompleto3_20160514_152957_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,131 KB
licitacionesPerfilesContratanteCompleto3_20160514_152957_3.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,337 KB
licitacionesPerfilesContratanteCompleto3_20160514_163319.atom	28/02/2018 09:50 a. m.	Archivo ATOM	5,768 KB
licitacionesPerfilesContratanteCompleto3_20160514_163319_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,025 KB
licitacionesPerfilesContratanteCompleto3_20160514_163319_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,368 KB
licitacionesPerfilesContratanteCompleto3_20160514_163319_3.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,137 KB
licitacionesPerfilesContratanteCompleto3_20160514_172731.atom	28/02/2018 09:50 a. m.	Archivo ATOM	5,702 KB
licitacionesPerfilesContratanteCompleto3_20160514_172731_1.atom	28/02/2018 09:50 a. m.	Archivo ATOM	5,785 KB
licitacionesPerfilesContratanteCompleto3_20160514_172731_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	5,900 KB
licitacionesPerfilesContratanteCompleto3_20160514_172731_3.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,827 KB
licitacionesPerfilesContratanteCompleto3_20160514_182837_2.atom	28/02/2018 09:50 a. m.	Archivo ATOM	6,065 KB

Ilustración 7: Estructura de un fichero descomprimido

De acuerdo con la plataforma, el fichero principal con el que se debe comenzar a leer es: *licitacionesPerfilesContratanteCompleto3.atom.* El contenido de todos los archivos es similar a ese y además son generados utilizando el estándar CODICE 2.0. Conforme al estándar, el nodo ***entry*** hace referencia a nuevas licitaciones y presenta seis nodos hijos de entre los que nos interesa el nodo ***<cac-place-ext:ContractFolderStatus>*** que es donde se encuentra todos la información detallada de la licitación.

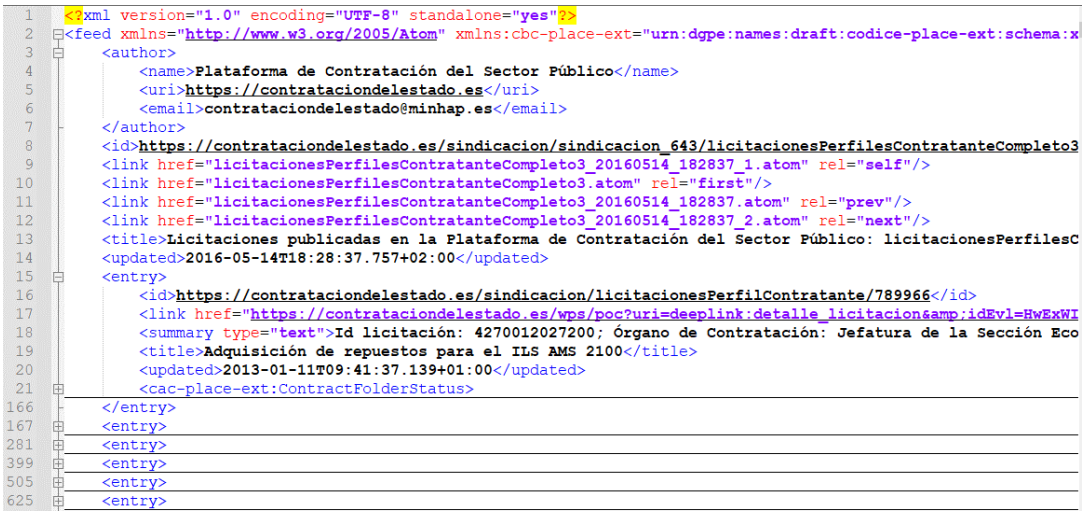

Ilustración 8: Contenido del archivo principal: licitacionesPerfilesContratanteCompleto3.atom

Dentro del contenido del nodo ***<cac-place-ext:ContractFolderStatus>*** se encuentra la información relativa a la licitación, además del estado posible del anuncio. De los nodos hijos, el más importante, puesto que contiene los datos requeridos para los objetivos de este proyecto, es el nodo ***<cac:ProcurementProject>*** que define un proyecto de compra y contiene toda la información definida en el proyecto original. El objeto del contrato se encuentra en el elemento ***<cbc:Name>*** y el código CPV asignado en ***<cac:RequiredCommodityClassification>.***

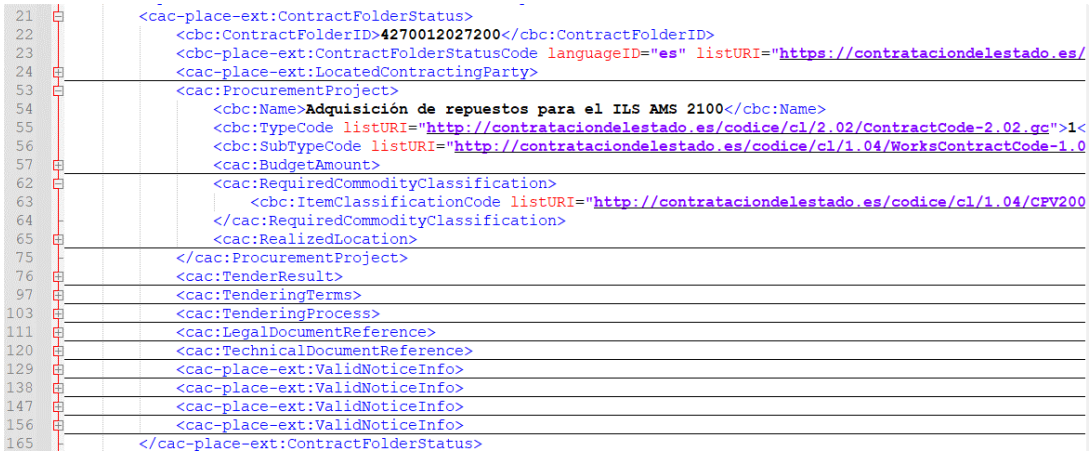

Ilustración 9: Contenido del nodo <cac-place-ext:ContractFolderStatus>

Hasta este punto se podría construir un programa informático que procese cada archivo y extraiga al menos los datos del objeto del contrato y el código CPV. No obstante, se tiene que almacenar toda esa información en algún sitio. Para este propósito, se ha creado una base de datos SQL para guardar los datos más

importantes de las licitaciones. El objetivo principal de la creación de una base de datos es mantener la integridad entre las licitaciones y su código CPV asociado, por ejemplo, si se intenta agregar un registro con un código CPV incorrecto, la base de datos por la propiedad de Consistencia del modelo ACID no lo permitiría y de este modo los registros que sí son almacenados son exactos e íntegros. En la siguiente ilustración se muestra el diagrama entidad-relación de la base de datos diseñada para almacenar los registros de las licitaciones que tengan asociado un código CPV. El diseño está compuesto por una tabla principal ***licitacion_es*** con una referencia foránea con la tabla ***CPV.***

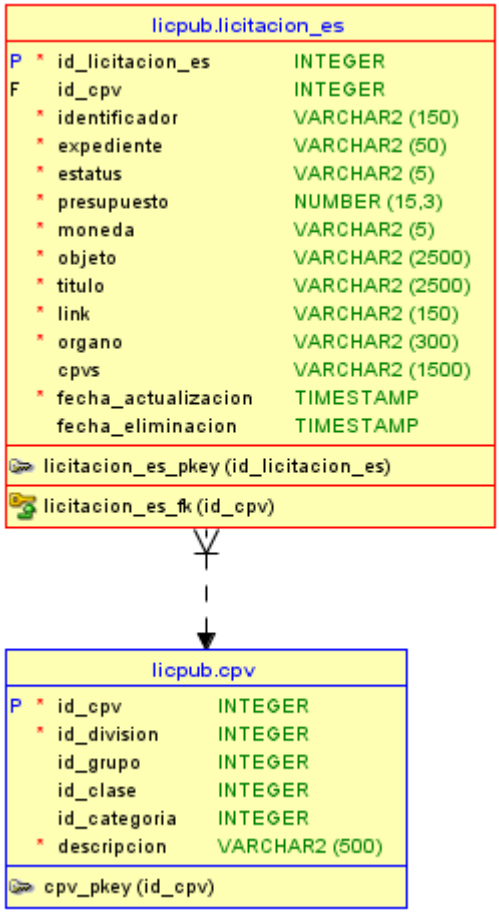

Ilustración 10: Diagrama de la base de datos licitacion_es

La tabla ***licitacion_es*** está pensada para almacenar los registros de las licitaciones públicas. Los atributos que se almacenan se describen a continuación, así como el mapeo con el estándar CODICE 2.0 de donde se extraen, además se muestra un ejemplo, en color rojo se representa el valor extraído. Todos los atributos son elementos o propiedades contenidas en el nodo <**entry>,** nodo que contiene todos los datos de una licitación.

- **id_licitacion_es**: Es el identificador único de la licitación, no se repite. El elemento asociado a este atributo del estándar CODICE 2.0 es el valor numérico del elemento **<id>** presente en el nodo **<entry>.**

```
<entry>
<id>https://contrataciondelestado.es/sindicacion/licitacionesPerfilCon
tratante/789966</id>
```

- **id_cpv:** Es el código CPV empleado para la clasificación del bien o servicio. Se extrae del elemento ***<cbc:ItemClassificationCode>*** que se encuentra en el nodo ***<cac:RequiredCommodityClassification>*** que a su vez se encuentra en el nodo ***<cac:ProcurementProject>***.

```
<entry>
 <cac-place-ext:ContractFolderStatus>
  <cac:ProcurementProject>
    <cac:RequiredCommodityClassification>
     <cbc:ItemClassificationCode
listURI="http://contrataciondelestado.es/codice/cl/1.04/CPV2007-
1.04.gc">32500000
   </cbc:ItemClassificationCode>
```

- **Identificador:** Es el identificador único de la licitación. El elemento asociado a este atributo del estándar CODICE 2.0 es el valor completo del elemento **<id>**.

```
<entry>
<id>https://contrataciondelestado.es/sindicacion/licitacionesPerfilCon
tratante/789966</id>
```

- **Expediente:** Es el número de expediente que corresponde a la licitación. Según el estándar CODICE 2.0 es aplicable únicamente a documentos de licitación electrónica. En el mapeo con los archivos XML este atributo se extrae del elemento ***<cbc:ContractFolderID>*** que se encuentra en el nodo ***<cac-place-ext:ContractFolderStatus>.***

```
<entry>
 <cac-place-ext:ContractFolderStatus>
   <cbc:ContractFolderID>4270012027200</cbc:ContractFolderID>
```

- **Estatus:** Es el código referente al estado del expediente, es un código empleado para relacionar el expediente a la fase en la que se encuentra la licitación. El valor se extrae del elemento ***<cbc-place-ext:ContractFolderStatusCode>*** presente en el nodo ***<cac-place-ext:ContractFolderStatus>.***

```
<entry>
 <cac-place-ext:ContractFolderStatus>
   <cbc-place-ext:ContractFolderStatusCode              languageID="es"
listURI="https://contrataciondelestado.es/codice/cl/2.04/SyndicationCo
ntractFolderStatusCode-2.04.gc">RES</cbc-place-
ext:ContractFolderStatusCode>
```

- **Presupuesto:** Es el valor total estimado del presupuesto para el contrato. Este dato se obtiene del valor del elemento ***<cbc:TaxExclusiveAmount***>

del nodo ***<cac:BudgetAmount>*** del nodo padre ***<cac:ProcurementProject>.***

```
<entry>
 <cac-place-ext:ContractFolderStatus>
  <cac:ProcurementProject>
   <cac:BudgetAmount>
    <cbc:TaxExclusiveAmount
currencyID="EUR">50377.14</cbc:TaxExclusiveAmount>
```

- **Moneda:** Es el código de la moneda empleada en el presupuesto. Este valor se obtiene del atributo ***currencyID*** del elemento ***<cbc:TaxExclusiveAmount***>.

```
<entry>
 <cac-place-ext:ContractFolderStatus>
  <cac:ProcurementProject>
   <cac:BudgetAmount>
    <cbc:TaxExclusiveAmount
currencyID="EUR">50377.14</cbc:TaxExclusiveAmount>
```

- **Objeto:** Es el nombre del proyecto de compra u objeto del contrato. El dato se obtiene del valor del elemento ***<cbc:Name>*** que se encuentra dentro del nodo ***<cac:ProcurementProject>.***

```
<entry>
 <cac-place-ext:ContractFolderStatus>
  <cac:ProcurementProject>
    <cbc:Name>Adquisición de repuestos para el ILS AMS 2100</cbc:Name>
```

- **Título:** Define el título del contrato. El dato es extraído del elemento ***<title>*** de cada licitación.

```
<entry>
  <title>Adquisición de repuestos para el ILS AMS 2100</title>
```

- **Link:** Es la referencia al detalle del expediente en formato URL a la Plataforma de Contratación Pública del Estado. Dato extraído del elemento ***<link>*** de cada licitación.

```
<entry>
  <link
href="https://contrataciondelestado.es/wps/poc?uri=deeplink:detalle_li
citacion&amp;idEvl=HwExWI%2FHuQQQK2TEfXGy%2BA%3D%3D"/>
```

- **Órgano.** Es la institución u órgano a cargo de la contratación pública. El valor asociado en el estándar CODICE 2.0 es ***<cbc:Name>*** dentro de los nodos ***<cac:PartyName>, <cac:Party>, <cac-place-ext:LocatedContractingParty>*** y ***<cac-place-ext:ContractFolderStatus>.***

```
<entry>
 <cac-place-ext:ContractFolderStatus>
  <cac-place-ext:LocatedContractingParty>
   <cac:Party>
    <cac:PartyName>
    <cbc:Name>Jefatura de la Sección Económico-Administrativa 27 - Base
Aérea de Getafe</cbc:Name>
  </cac:PartyName>
```

- **CPVs:** En caso de que la licitación contenga más de un código CPV asignado, se almacenan aquí separados por punto y coma. En cualquier otro caso se presenta el valor nulo.

  ```
  Null
  ```

- **Fecha_actualizacion:** Es la fecha de la última actualización del contrato. El valor es extraído del nodo ***<updated>*** de cada ***<entry>.***

  ```
  <entry>
    <updated>2013-01-11T09:41:37.139+01:00</updated>
  ```

- **Fecha_eliminacion:** Este dato está pensado para almacenar la fecha en que se ha eliminado la licitación.

La tabla CPV contiene todos los registros de los códigos CPVs descritos en el estándar del Vocabulario Común de Contratos Públicos de la Unión Europea. Al estar estructurados de manera jerárquica, se ha realizado un diseño para almacenarlos en distintas tablas, cada tabla representa el nivel de código en la jerarquía correspondiente. Primero, en el nivel más alto se encuentra la división, luego el grupo, la clase, la categoría y finalmente el CPV.

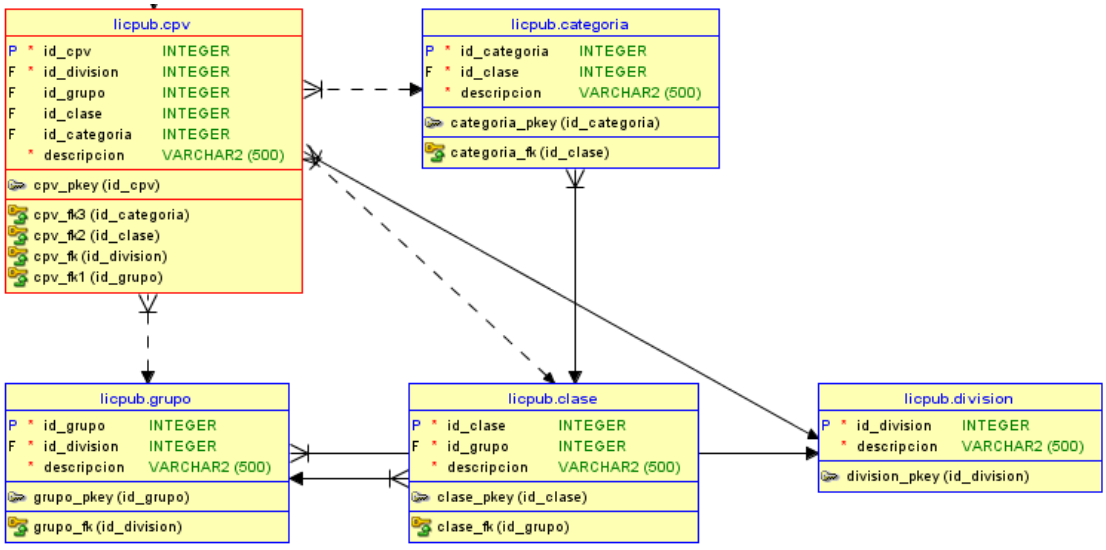

Ilustración 11: Tabla de códigos CPV

Debido a la restricción de llave foránea de la tabla ***licitacion_es*** con la tabla ***CPV***, primero es necesario poblar la tabla ***CPV*** con los códigos del vocabulario antes de guardar los registros de las licitaciones. Para ingresar los términos del vocabulario, se ha procesado el archivo que se encuentra en el siguiente enlace: ***http://contrataciondelestado.es/codice/cl/1.04/CPV2007-1.04.gc***. El documento, referenciado en el estándar CODICE 2.0, contiene el listado de todos los códigos CPV en formato XML. Cada término del vocabulario se encuentra en el nodo ***row***, y este está conformado por tres elementos: **code**, **name** y **nombre**. El dato ***name*** y ***code*** son el mismo valor y representan el código numérico del bien o servicio; la descripción del término se encuentra en el elemento ***nombre***. La estructura XML para un código es la siguiente:

```
<Row>
   <Value ColumnRef="code">
      <SimpleValue>03000000</SimpleValue>
   </Value>
   <Value ColumnRef="name">
      <SimpleValue>03000000</SimpleValue>
   </Value>
   <Value ColumnRef="nombre">
      <SimpleValue>Productos de la agricultura, ganadería, pesca,
silvicultura y productos afines.</SimpleValue>
   </Value>
</Row>
```

A partir de la estructura XML anterior, se realiza una transformación de forma, haciendo uso de expresiones regulares, para convertir los códigos a un formato de valores separados por tabuladores, tal como se muestra a continuación.

```
Code        name        nombre
03000000    03000000    Productos de la agricultura, ganadería, pesca,
silvicultura y productos afines.
03100000    03100000    Productos de la agricultura y horticultura.
03110000    03110000    Cultivos, productos comerciales de jardinería
y horticultura.
03111000    03111000    Semillas
03111100    03111100    Soja.
03111200    03111200    Cacahuetes.
```

Aprovechando la naturaleza de la estructura de los códigos CPV; la división representa dos dígitos del código, el grupo tres, la clase cuatro y la categoría cinco, se transforma a partir del código su correspondiente división, grupo, clase y categoría tal y como se muestra en los siguientes ejemplos.

```
code|division|grupo|clase|categoria|name|nombre
03000000|03000000|03000000|03000000|03000000|03000000|Productos de la
agricultura, ganadería, pesca, silvicultura y productos afines
03100000|03000000|03100000|03100000|03100000|03100000|Productos de la
agricultura y horticultura
03110000|03000000|03100000|03110000|03110000|03110000|Cultivos,
productos comerciales de jardinería y horticultura
03111000|03000000|03100000|03110000|03111000|03111000|Semillas
```

```
03111100|03000000|03100000|03110000|03111000|03111100|Soja
03111200|03000000|03100000|03110000|03111000|03111200|Cacahuetes
```

Con la ayuda de una notebook de Colab, se obtienen los términos que corresponden únicamente a las divisiones, grupos, clases y categorías. Finalmente, cada uno de los términos es insertado en la tabla correspondiente.

	id_division	descripcion
1	3000000	Productos de la agricultura, ganadería, pesca, silvicultura y productos afines
2	9000000	Derivados del petróleo, combustibles, electricidad y otras fuentes de energía
3	14000000	Productos de la minería, de metales de base y productos afines
4	15000000	Alimentos, bebidas, tabaco y productos afines
5	16000000	Maquinaria agrícola
6	18000000	Prendas de vestir, calzado, artículos de viaje y accesorios
7	19000000	Piel y textiles, materiales de plástico y caucho
8	22000000	Impresos y productos relacionados
9	24000000	Productos químicos
10	30000000	Máquinas, equipo y artículos de oficina y de informática, excepto mobiliario y paquetes de software

Ilustración 12: Códigos CPV a nivel División

	id_grupo	id_division	descripcion
1	3100000	3000000	Productos de la agricultura y horticultura
2	3200000	3000000	Cereales, patatas, hortalizas, frutas y frutos de cáscara
3	3300000	3000000	Productos de la ganadería, la caza y la pesca
4	3400000	3000000	Productos de la silvicultura y de la explotación forestal
5	9100000	9000000	Combustibles
6	9200000	9000000	Productos del petróleo, del carbón y de aceites minerales
7	9300000	9000000	Electricidad, calefacción, energías solar y nuclear
8	14200000	14000000	Arena y arcilla
9	14300000	14000000	Minerales químicos y abonos minerales
10	14400000	14000000	Sal y cloruro de sodio puro

Ilustración 13: Códigos CPV a nivel Grupo

	id_clase	id_grupo	descripcion
1	3110000	3100000	Cultivos, productos comerciales de jardinería y horticultura
2	3120000	3100000	Productos de horticultura y viveros
3	3130000	3100000	Cultivos de especias y de plantas para bebidas
4	3140000	3100000	Productos de origen animal y productos afines
5	3210000	3200000	Cereales y patatas
6	3220000	3200000	Hortalizas, frutas y frutos de cáscara
7	3310000	3300000	Pescado, crustáceos y productos acuáticos
8	3320000	3300000	Ganado y animales pequeños
9	3330000	3300000	Productos de animales de granja
10	3340000	3300000	Marcas auriculares para animales

Ilustración 14: Códigos CPV a nivel Clase

	id_categoria	id_clase	descripcion
1	3111000	3110000	Semillas
2	3112000	3110000	Tabaco sin elaborar
3	3113000	3110000	Plantas utilizadas para la fabricación de azúcar
4	3114000	3110000	Paja y plantas forrajeras
5	3115000	3110000	Materias vegetales en bruto
6	3116000	3110000	Caucho y látex naturales y productos afines
7	3117000	3110000	Plantas utilizadas para usos específicos
8	3121000	3120000	Productos de horticultura
9	3131000	3130000	Cultivos de plantas para bebidas
10	3132000	3130000	Especias no manufacturadas

Ilustración 15: Códigos CPV a nivel Categoría

En algunos casos, fue necesario agregar un valor jerárquico faltante para poder ingresar registros de un nivel inferior. Por ejemplo, el código 39254000 que corresponde, en el nivel de Categoría, al término relojería no puede ser insertado en la tabla de Categoría debido a que no existe el valor jerárquico superior a nivel de Clase. El valor faltante corresponde al código 39250000. Lo mismo ocurre al insertar algunos valores en un nivel inferior a categoría, por ejemplo, los códigos 35811300; uniformes militares, 35811200; uniformes de policía y 35811100; uniformes para el cuerpo de bomberos, no pueden ser insertados si no existe, a nivel de categoría, el código 35811000. Agregar los códigos no era una tarea complicada, sin embargo, para agregar las descripciones de los códigos faltantes se tuvo que investigar en versiones pasadas del estándar CPV donde sí se encontraban los términos y recuperar la descripción, para algunos otros se ha ideado una descripción acorde a la categoría.

Tabla 3: Códigos CPV faltantes

CPV	Nivel	Descripción
39250000	Clase	Relojería
60110000	Clase	Servicios de transporte por la vía pública
34511000	Categoría	Buques de guerra
35611000	Categoría	Aviones militares
35612000	Categoría	Helicópteros militares
35811000	Categoría	Uniformes
38527000	Categoría	Sistema de dosimetría
42924000	Categoría	Pistolas pulverizadoras, máquinas de chorro de arena o de vapor

Finalmente, en la tabla CPV se agregan todos los términos del vocabulario. Además, se agregan las referencias foráneas a las tablas División, Grupo, Clase y Categoría cuando el código CPV corresponde a dicha categoría.

	id_cpv	id_division	id_grupo	id_clase	id_categoria	descripcion
1	3000000	3000000	(null)	(null)	(null)	Productos de la agricultura, ganadería, pesca, silvicultura y productos afines
2	3100000	3000000	3100000	(null)	(null)	Productos de la agricultura y horticultura
3	3110000	3000000	3100000	3110000	(null)	Cultivos, productos comerciales de jardinería y horticultura
4	3111000	3000000	3100000	3110000	3111000	Semillas
5	3111100	3000000	3100000	3110000	3111000	Soja
6	3111200	3000000	3100000	3110000	3111000	Cacahuetes
7	3111300	3000000	3100000	3110000	3111000	Semillas de girasol
8	3111400	3000000	3100000	3110000	3111000	Semillas de algodón
9	3111500	3000000	3100000	3110000	3111000	Semillas de sésamo
10	3111600	3000000	3100000	3110000	3111000	Semillas de mostaza

Ilustración 16: Códigos CPV

Después de tener en la base de datos todos los códigos CPV, ya es posible agregar registros de las licitaciones públicas en la tabla de ***licitacion_es*** conforme al análisis planteado al comienzo de este apartado. Sin embargo, la creación de un programa desde cero puede ser propenso a errores, por ello se ha hecho uso de la herramienta que adicional a los ficheros comprimidos de las licitaciones, el Ministerio de Hacienda pone a disposición. La herramienta denominada OpenPLACSP permite facilitar la transformación de cada uno de los archivos XML en hojas de cálculo o archivos CSV. Dicha herramienta cuenta con una licencia de software libre que, entre los derechos otorgados por la misma, permite la modificación o realización de obras derivadas. Por lo anterior, se optó por crear un programa basado en OpenPLACSP para navegar por cada uno de los archivos y ficheros comprimidos de la misma manera que la aplicación lo hace para obtener los datos e insertarlos automáticamente en la base de datos. El código fuente del programa, los scripts para la creación de la base, sentencias *inserts* y los archivos utilizados se encuentran en el repositorio principal de este proyecto.

Tabla 4: Ejemplo de un registro en la tabla licitacion_es

Atributo	**Valor**
id_licitacion	726510
id_cpv	90919200
Identificador	https://contrataciondelestado.es/sindicacion/licitacionesPerfilContratante/726510
Expediente	DG-2/2012
Estatus	RES
Presupuesto	37800.07
Moneda	EUR
Objeto	Servicio de limpieza de las Oficinas de Extranjeros de la Avenida Constitución 106-108 y 116 de Valencia, dependientes de la Delegación del Gobierno en la Comunidad Valenciana.
Título	Servicio de limpieza de las Oficinas de Extranjeros de la Avenida Constitución 106-108 y 116 de Valencia, dependientes de la Delegación del Gobierno en la Comunidad Valenciana.
Link	https://contrataciondelestado.es/wps/poc?uri=deeplink:detalle_licitacion&idEvl=1vdceJnWtn4QK2TEfXGy%2BA%3D%3D
Órgano	Delegación del Gobierno en la Comunidad Valenciana
CPVs	null
Fecha actualización	2012-04-24 11:55:06.666
Fecha eliminación	null

3.1.2 Recolección de datos de México

Con la experiencia obtenida en la recolección del corpus de licitaciones públicas de España, para el caso de las licitaciones de México, también se han diseñado tablas de base de datos relacionales para almacenar los registros de los códigos de clasificación de artículos, en este caso CUCoPs, y las licitaciones publicadas en los portales abiertos del Gobierno de México.

Antes de agregar los registros de las licitaciones, se ha de ingresar todos los códigos CUCoPs en su tabla de jerarquía correspondiente. El nivel más alto está compuesto por el Capítulo, continúa con el Concepto, luego la Partida Genérica y finalmente la Partida Específica. La ventaja principal de almacenar las claves en tablas relaciones resulta en tener certeza que un código está asociado con sus niveles jerárquicos superiores inequívocamente, puesto que a diferencia de los códigos CPV, a partir de las claves CUCoP no se puede inferir todos los niveles jerárquicos superiores. Para ilustrar lo anterior, se toma como ejemplo las claves 21500003 y 21500011, ambas corresponden al capítulo 2000, al concepto 2100, a la partida genérica 2150, pero tienen distinta partida específica, estas son 21501 y 21502 respectivamente. Si solo se cuenta con el código CUCoP, no se podría deducir la partida específica correspondiente. Lo anterior ocurre debido a la naturaleza de la estructura de los códigos CUCoP, ya que una clave está compuesta por la partida específica sin el último dígito y un numero consecutivo.

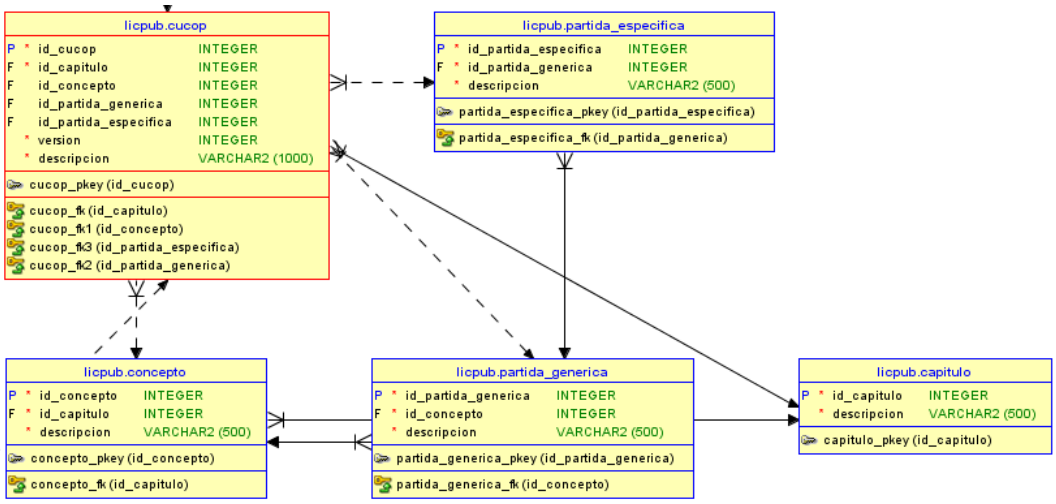

Ilustración 17: Tablas CUCoPs

El vocabulario de términos se ha obtenido del portal de datos abiertos de CompraNet, sin embargo, el listado se presenta en formato de hoja de cálculo, lo que dificulta la extracción de manera automática.

TIPO	CLAVE CUCoP	PARTIDA ESPECÍFICA	CLAVE CUCoP +	DESCRIPCIÓN	NIVEL	CABM	UNIDAD DE MEDIDA (sugerida)	TIPO DE CONTRATACIÓN
1	2000		2000	Materiales y suministros	1			Adquisiciones
1	2100	2000	2100	Materiales de administración, emisión de documentos y artículos oficiales	2			Adquisiciones
1	2110	2100	2110	Materiales, útiles y equipos menores de oficina	3			Adquisiciones
1	21101	2110	21101	Materiales y útiles de oficina	4			Adquisiciones
1	21100001	21101	21101-0001	Abrecartas	5	C210000004	Pieza	Adquisiciones
1	21100002	21101	21101-0002	Achaparrador de letras	5	C450000060	Pieza	Adquisiciones
1	21100003	21101	21101-0003	Acrileta	5	C210000194	Pieza	Adquisiciones
1	21100004	21101	21101-0004	Alfileminas	5	C210000006	Pieza	Adquisiciones
1	21100005	21101	21101-0005	Agenda	5	C210000216	Pieza	Adquisiciones
1	21100006	21101	21101-0006	Aguja para alacrán	5	C450000062	Pieza	Adquisiciones
1	21100007	21101	21101-0007	Alargadera	5	C450000034	Pieza	Adquisiciones
1	21100008	21101	21101-0008	Álbum	5	C210000008	Pieza	Adquisiciones
1	21100009	21101	21101-0009	Alfiler para señalización en mapa	5	C210000202	Pieza	Adquisiciones
1	21100010	21101	21101-0010	Aparato automático para fijar chinches	5	C210000010	Pieza	Adquisiciones
1	21100011	21101	21101-0011	Apoyabrazos	5	C210000012	Pieza	Adquisiciones
1	21100012	21101	21101-0012	Arbol de navidad	5	C180000160	Pieza	Adquisiciones
1	21100013	21101	21101-0013	Arenero	5	C210000252	Pieza	Adquisiciones
1	21100014	21101	21101-0014	Atril mecanográfico	5	C210000014	Pieza	Adquisiciones
1	21100015	21101	21101-0015	Barra listero (porta listas o barra rotafolio)	5	C210000016	Pieza	Adquisiciones
1	21100016	21101	21101-0016	Barril punto rapidógrafo	5	C450000002	Pieza	Adquisiciones
1	21100017	21101	21101-0017	Base agenda	5	C210000018	Pieza	Adquisiciones
1	21100018	21101	21101-0018	Base calendario	5	C210000020	Pieza	Adquisiciones

Ilustración 18: Listado de códigos CUCoPs

De manera manual, se generan las sentencias *inserts* para almacenar todo el vocabulario de términos. Partiendo por niveles jerárquicos superiores y relacionando sus claves con los niveles inferiores asociados.

	id_capitulo	descripcion
1	2000	Materiales y suministros
2	3000	Servicios generales
3	5000	Bienes muebles, e intangibles
4	6000	Inversión Pública

Ilustración 19: Códigos CUCoPs a nivel Capítulo

	id_concepto	id_capitulo	descripcion
1	2100	2000	Materiales de administración, emisión de documentos y artículos oficiales
2	2200	2000	Alimentos y utensilios
3	2300	2000	Materias primas y materiales de producción y comercialización
4	2400	2000	Materiales y artículos de construcción y de reparación
5	2500	2000	Productos químicos, farmacéuticos y de laboratorio
6	2600	2000	Combustibles, lubricantes y aditivos
7	2700	2000	Vestuario, blancos, prendas de protección y artículos deportivos
8	2800	2000	Materiales y suministros para seguridad
9	2900	2000	Herramientas, refacciones y accesorios menores
10	3100	3000	Servicios básicos

Ilustración 20: Códigos CUCoP a nivel Concepto

	id_partida_generica	id_concepto	descripcion
1	2110	2100	Materiales, útiles y equipos menores de oficina
2	2120	2100	Materiales y útiles de impresión y reproducción
3	2130	2100	Material estadístico y geográfico
4	2140	2100	Materiales, útiles y equipos menores de tecnologías de la información y comunicaciones
5	2150	2100	Material impreso e información digital
6	2160	2100	Material de limpieza
7	2170	2100	Materiales y útiles de enseñanza
8	2180	2100	Materiales para el registro e identificación de bienes y personas
9	2210	2200	Productos alimenticios para personas
10	2220	2200	Productos alimenticios para animales

Ilustración 21: Códigos CUCoP a nivel Partida Genérica

	id_partida_especifica	id_partida_generica	descripcion
1	21101	2110	Materiales y útiles de oficina
2	21201	2120	Materiales y útiles de impresión y reproducción
3	21301	2130	Material estadístico y geográfico
4	21401	2140	Materiales y útiles consumibles para el procesamiento en equipos y bienes informáticos
5	21501	2150	Material de apoyo informativo
6	21502	2150	Material para información en actividades de investigación científica y tecnológica
7	21601	2160	Material de limpieza
8	21701	2170	Materiales y suministros para planteles educativos
9	21800	2180	Materiales para el registro e identificación de bienes y personas
10	21801	2180	Materiales para el registro e identificación de bienes y personas

Ilustración 22: Códigos CUCoP a nivel Partida Específica

	id_cucop	id_capitulo	id_concepto	id_partida_generica	id_partida_especifica	version	descripcion
1	2000	2000	(null)	(null)	(null)	2022	Materiales y suministros
2	2100	2000	2100	(null)	(null)	2022	Materiales de administración, emisión de documentos y artículos oficiales
3	2110	2000	2100	2110	(null)	2022	Materiales, útiles y equipos menores de oficina
4	2120	2000	2100	2120	(null)	2022	Materiales y útiles de impresión y reproducción
5	21101	2000	2100	2110	21101	2022	Materiales y útiles de oficina
6	21100001	2000	2100	2110	21101	2022	Abrecartas
7	21100002	2000	2100	2110	21101	2022	Achaparrador de letras
8	21100003	2000	2100	2110	21101	2022	Acrileta
9	21100004	2000	2100	2110	21101	2022	Afilaminas
10	21100005	2000	2100	2110	21101	2022	Agenda

Ilustración 23: Códigos CUCoP

Después de haber almacenado todas las claves CUCoPs, se procede con el análisis para insertar los registros de licitaciones que se publican en el Portal de Contrataciones Abiertas. Las licitaciones se pueden consultar gráficamente en el portal y también es posible extraer la información de cada una de ellas en formato estructurado mediante la API REST que se encuentra en el siguiente E*ndpoint* https://api.datos.gob.mx/v2/contratacionesabiertas. En dicho sitio, los datos de las licitaciones son publicados utilizando el estándar del EDCA en formato JSON, de igual manera que con el formato XML se puede realizar la extracción de datos automáticamente. Para poder almacenar los registros se ha diseñado una tabla relacional con los atributos más importantes para este trabajo.

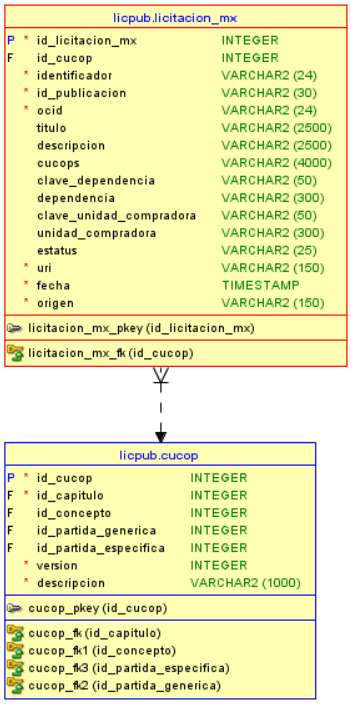

Ilustración 24: Tabla licitacion_mx

La descripción de los atributos de la tabla ***licitacion_mx*** y el mapeo con los elementos del estándar EDCA, de donde son extraídos automáticamente, se muestra a continuación.

- **Id_licitacion_mx**: Es el código del expediente o identificador de la licitación. El dato en el estándar del EDCA es extraído del campo ***tender/id***.

```
"tender": {
      "value": {
            "amount": 0
      },
      "status": "complete",
      "procurementMethodRationale": "Art. 41 fr. I",
      "procurementMethod": "direct",
      "items": [],
      "id": "1451528",
      "hasEnquiries": false
  }
```

- **Id_cucop:** Número que identifica al bien, servicio u obra pública objeto del procedimiento de contratación. El valor de dato es extraído del campo ***contracts/0/items/0/classification/id.***

```
"contracts": [
      {
        "items": [
        {
          "unit": {
            "value": {
               "currency": "MXN",
               "amount": 4183.08
               },
            "name": "Servicio"
            },
          "classification": {
            "id": "33900001",
            "description": "Estudios e investigaciones"
           }
```

- **OCID**: Es un identificador único para un procedimiento de contratación. Se compone de un prefijo del editor y un identificador para el procedimiento de contratación. Este dato se encuentra mapeado en el estándar del EDCA en el campo ***OCID***.

```
"compiledRelease": {
      "parties": [...],
      "contracts": [...],
      "awards": [...],
      "tender": {...},
      "publisher": {...},
      "buyer": {...},
      "ocid": "ocds-07smqs-1451528",
      "language": "es",
      "initiationType": "tender",
      "id": "SFP-1451528-2018-11-12",
      "date": "2017-10-06T10:03:01Z",
```

- **Título:** Es la denominación del procedimiento de contratación que las dependencias y entidades describen. El campo asociado al estándar del EDCA es: ***tender/title.***


```
"compiledRelease": {
      "parties": [...],
      "contracts": [...],
      "awards": [...],
      "tender": {
        "value": {...},
        "tenderPeriod": {...},
        "submissionMethod": [...],
        "procuringEntity": {...},
        "enquiryPeriod": {...},
        "awardPeriod": {...},
        "title": "SERVICIOS PROFESIONALES PARA LA ELABORACIÓN DE AVALÚOS",
        "status": "complete",
        "procurementMethodRationale": "Art. 41 fr. I",
        "procurementMethod": "direct",
        "items": [],
        "id": "1451528",
        "hasEnquiries": false
```


- **Descripción:** Es el objeto del procedimiento de contratación que las dependencias y entidades describen. El campo asociado al estándar del EDCA es: ***tender/description.***


```
"compiledRelease": {
      "parties": [...],
      "contracts": [...],
      "awards": [...],
      "tender": {
        "value": {...},
        "tenderPeriod": {...},
        "submissionMethod": [...],
        "procuringEntity": {...},
        "enquiryPeriod": {...},
        "awardPeriod": {...},
        "title": "SERVICIO DE MANEJO Y OPERACIÓN DE LA ZONA DE MONUMENTOS ARQUEOLÓGICOS DE CHICHEN",
        "status": "complete",
        "procurementMethodRationale": "Art. 41 fr. XIV",
        "procurementMethod": "direct",
        "items": [],
        "id": "1668835",
        "hasEnquiries": false,
        "description": "SERVICIO DE MANEJO Y OPERACIÓN DE LA ZONA DE MONUMENTOS ARQUEOLÓGICOS DE CHICHEN ITZÁ"
```


- **CUCoPs:** Si la licitación cuenta con más de una clave CUCoP asignada, se almacenan en este campo los valores separados por punto y coma.

- **Clave_dependencia:** Es el identificador asociado a la entidad compradora. El dato del estándar del EDCA se encuentra en el elemento ***parties/0/id***, cuando el rol de los integrantes del procedimiento de contratación es ***buyer.***

```
"compiledRelease": {
      "parties": [
        {
            "roles": [
              "buyer"
            ],
            "name": "Instituto Nacional de Antropología e Historia",
            "id": "INAH-195"
```


- **Dependencia:** Es la entidad compradora, la institución pública cuyo presupuesto se usará para adquirir los bienes o servicios. Para identificar a la entidad compradora, el rol que desempeña de acuerdo con el estándar EDCA es: ***buyer***. El valor de este atributo es extraído del campo ***parties/0/name***.


```
"compiledRelease": {
      "parties": [
        {
            "roles": [
              "buyer"
            ],
            "name": "Instituto Nacional de Antropología e Historia",
            "id": "INAH-195"
```


- **Clave_unidad_compradora:** Es el identificador asociado a la entidad contratante, el encargado de administrar la compra. De acuerdo con el estándar del EDCA, el rol que desempeña este integrante en el procedimiento de contratación es ***procuringEntity***. El valor es extraído del campo ***parties/0/id.***


```
"compiledRelease": {
      "parties": [
        {
            "roles": [
              "procuringEntity"
            ],
            "name": "INAH-Dir. de la Coordinación de Recursos Materiales y Servicios #048D00001",
            "id": "INA460815GV1-048D00001"
```


- **Unidad_compradora:** Es la entidad contratante. Esta es la responsable de realizar los procedimientos de contratación a efecto de adquirir los bienes o servicios, puede ser distinta a la entidad compradora. El rol que desempeña en el estándar del EDA es ***procuringEntity.*** El dato se extrae del campo: ***parties/0/id.***


```
"compiledRelease": {
      "parties": [
        {
            "roles": [
              "procuringEntity"
            ],
            "name": "INAH-Dir. de la Coordinación de Recursos Materiales y Servicios #048D00001",
            "id": "INA460815GV1-048D00001"
```

- **Estatus:** Es el estatus del procedimiento de contratación, valor asignado a las claves establecidas por el estándar del EDCA. El dato es extraído del campo ***tender/estatus.***


```
"compiledRelease": {
      "parties": [...],
      "contracts": [...],
      "awards": [...],
      "tender": {
        "value": {...},
        "tenderPeriod": {...},
        "submissionMethod": [...],
        "procuringEntity": {...},
        "enquiryPeriod": {...},
        "awardPeriod": {...},
       "title": "SERVICIOS PROFESIONALES PARA LA ELABORACIÓDE AVALÚOS",
        "status": "complete",
        "procurementMethodRationale": "Art. 41 fr. I",
        "procurementMethod": "direct",
        "items": [],
        "id": "1451528",
        "hasEnquiries": false
```


- **URI:** Es el identificador del paquete de entrega. El valor es mapeado en el campo ***uri.***


```
  {
      "_id": "5c13470b940a520a158dcce6",
      "uri":
"https://api.datos.gob.mx/v2/contratacionesabiertas?records.ocid=ocds-
07smqs-1668835",
```


- **Fecha:** Es la fecha en la cual la información contenida en la entrega fue registrada y publicada por primera vez.


```
"compiledRelease": {
      "parties": [...],
      "contracts": [...],
      "awards": [...],
      "tender": {...},
      "publisher": {...},
      "buyer": {...},
      "ocid": "ocds-07smqs-1451528",
      "language": "es",
      "initiationType": "tender",
      "id": "SFP-1451528-2018-11-12",
      "date": "2017-10-06T10:03:01Z",
```


- **Origen.** Este campo es para identificar el origen de donde se ha extraído la licitación, básicamente es el recurso obtenido al hacer la petición a la API.
  //api//data_file_page_1000.json

Se creo un programa informático que recorre cada una de las páginas del *Endpoint* para extraer los datos de las licitaciones y almacenarlos en la tabla

creada para ese fin. Sin embargo, existían registros de licitaciones que estaban asociados a términos del vocabulario CUCoP, pero los códigos no están presentes en el archivo de claves publicado por CompraNet en su versión más reciente (6 de abril del 2022). Al estar relacionadas las tablas ***CUCoP*** y ***licitacion_mx***, no se podía insertar a la base de datos los registros de dichas licitaciones sin incurrir en un problema de integridad en los datos. Lo que ocurría era que, las claves CUCoPs son actualizadas cada año debido a que este esquema responde a las modificaciones y actualización que sufre el Clasificador por Objeto del Gasto (COG) que es publicado por la Secretaría de Hacienda y Crédito Público. El COG es la clasificación de los bienes y servicios en que se materializará el presupuesto y al que se asignan recursos públicos. Las claves CUCoP no se modifican, solamente se agregan o eliminan en versiones recientes.

Para solventar lo anterior, se procesaron los archivos de versiones pasadas que contienen claves CUCoPs. Retroactivamente se agregaron las claves publicadas en los años 2021, 2020, 2019, 2018, 2017, 2016 y 2013 que complementan las claves del ejercicio 2022. Por ejemplo, la versión del 2021 contiene los mismos códigos que se encuentran en el 2022, pero en el 2020 existía la clave 53100006 que corresponde al bien Andadera ortopédica (equipo médico quirúrgico) y ésta no se encuentra en el listado de claves más reciente. A continuación, se muestra el detalle de las claves agregadas por año.

Tabla 5: Códigos CUCoP insertados por año

Año	No. de Códigos insertados
2022	12635
2021	0
2020	1
2019	159
2018	15
2017	1286
2016	679
2013	38

Después de tener todos los códigos CUCoP desde 2013 hasta la fecha, se procede a insertar automáticamente los datos de las licitaciones públicas.

Tabla 6: Ejemplo de un registro de la tabla licitacion_mx

Atributo	Valor
id_licitacion_mx	1819192
id_cucop	25400294
identificador	5c12e0c6940a520a158ae3f2
id_publicacion	SFP-1819192-2018-11-13
OCID	ocds-07smqs-1819192
Titulo	Adquisicion de Material de Curación (Jeringas para Insulina)
Descripción	Adquisicion de Material de Curación (Jeringas para Insulina)
CUCoPs	Null
Clave Dependencia	IMSS-192
Dependencia	Instituto Mexicano del Seguro Social

Clave Unidad Compradora	IMS421231I45-050GYR020
Unidad Compradora	IMSS-UMAE HOSPITAL DE ESPECIALIDADES, C.M.N.O. #050GYR020
Estatus	complete
URI	https://api.datos.gob.mx/v2/contratacionesabiertas?records.ocid=ocds-07smqs-1819192
Fecha	2018-11-07 10:33:37.0
Origen	C:\Users\alvar\Documents\UPM\TFM\Mexico\Contrataciones Abiertas\api/data_file_page_46.json

Posteriormente al análisis de los procedimientos de contratación almacenados hasta este punto se ha encontrado que los registros de las licitaciones solo corresponden a los años del periodo 2017 y 2018. Con la finalidad de complementar los registros actuales con otros años, se ha explorado el portal de Datos Abiertos de CompraNet donde también se realizan publicaciones de expedientes relativos a las licitaciones que se registran en ese sistema, puesto que este es el portal oficial para las contrataciones públicas. En dicho portal, en el apartado de Datos Abiertos, se encuentran publicados archivos anuales con los expedientes registrados desde 2010 hasta la fecha (abril del 2022) en formato de valores separados por comas. Al revisar detalladamente todos archivos, se ha encontrado que desde 2010 hasta 2017 no se publicaron las claves CUCoPs correspondiente a los expedientes, siendo este un dato importante para fines de este proyecto no se consideran esos archivos para análisis posteriores. En el caso de los archivos del año 2018 hasta 2022, el dato CUCoP es representado por solo 4 dígitos, esto significa que el nivel jerárquico más específico alcanzado por esos expedientes solo corresponde a Partidas Genéricas.

Para guardar los registros de los expedientes del portal de CompraNet, se emplea la misma estrategia que se ha utilizado con los expedientes publicados en el Portal de Contrataciones Abiertas, se ha diseñado una tabla relacional. La tabla relacional denominada ***expediente*** contiene los atributos más importantes relativos a las contrataciones públicas de ese portal.

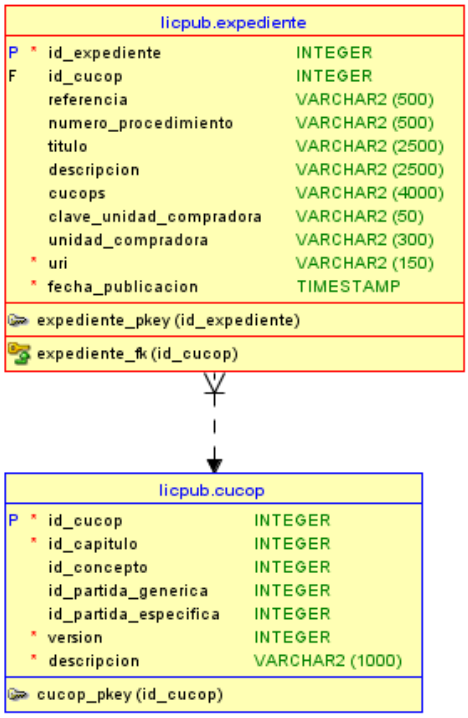

Ilustración 25: Diseño de la tabla expediente

La descripción de los atributos y el mapeo con las columnas de los archivos originales se menciona a continuación. La descripción va acorde con el diccionario de datos puesto a disposición por CompraNet.

- **id_expediente:** Es el número identificador del expediente en CompraNet. Corresponde al valor de la columna **Código del expediente.**

- **id_cucop:** Clave del Clasificador Único de las Contrataciones Públicas (CUCoP) que fueron asignadas por la Unidad Contratante al expediente de contratación; las claves CUCOP de 4 dígitos están relacionadas con el Clasificador por Objeto del Gasto (COG) el cual es administrado por la SHCP. Solamente se registra en este atributo si es una única clave. El valor corresponde a la columna **Clave CUCOP**.

- **Referencia:** Es la referencia del expediente o número de control interno del procedimiento de contratación. La columna asociada a este campo es **Referencia del expediente.**

- **Numero_procedimiento:** Número identificador del procedimiento de contratación en CompraNet. El valor corresponde a la columna **Número del procedimiento.**

- **Título:** Título de identificación del expediente en CompraNet. El valor corresponde a la columna **Título del expediente.**

- **Descripción:** Título de identificación del anuncio público del expediente en CompraNet. La columna asociada al este campo es **Descripción del anuncio.**

- **CUCoPs:** Clave o Claves CUCoPs que fueron asignadas por la Unidad Contratante al expediente de contratación. Corresponde a la columna **Clave CUCoP.**

- **Clave_unidad_compradora:** Clave con la que se identifica a la Unidad Compradora en CompraNet. La columna asociada al campo es **Clave de la UC.**

- **Unidad_compradora:** Nombre de la Unidad Compradora en CompraNet. El valor se extrae de la columna **Nombre de la UC.**

- **URI:** Vínculo correspondiente al anuncio público del procedimiento de contratación en CompraNet. La columna asociada a este atributo es **Dirección del anuncio.**

- **Fecha_publicacion:** Fecha de la primera publicación del anuncio público del expediente en CompraNet. La columna correspondiente a este valor es **Publicación del anuncio.**

Con ayuda de una notebook de Google Colaboratory se extraen los datos mencionados anteriormente de los archivos originales y luego se exportan a la base de datos.

Tabla 7: Ejemplo de un registro de la tabla expediente

Atributo	Valor
id_expediente	2241299
id_cucop	3390
Referencia	CASO 0307/2021
Numero procedimiento	AA-048E00995-E57-2021
Título	SERVICIOS PARA COADYUVAR, DE MANERA PRESENCIAL, VIRTUAL O A DISTANCIA, EN LA COO
Descripción	SERVICIOS PARA COADYUVAR, DE MANERA PRESENCIAL, VIRTUAL O A DISTANCIA, EN LA COORDINACIÓN EJECUTIVA DEL ESTUDIO DE LA ÓPERA DE BELLAS ARTES CUYAS ACTIVIDADES CONSISTEN EN COADYUVAR EN LA ELABORACIÓN DEL PRESUPUESTO ANUAL A FIN DE SER APROBADO POR LA COMPAÑÍA NACIONAL DE ÓPERA, APOYAR EN LA LOGÍSTICA PARA LA REALIZACIÓN DE ENSAYOS Y AUDICIONES DE LOS BENEFICIARIOS DEL EOBA A TRAVÉS DE LA ASIGNACIÓN DE ESPACIOS Y SALONES DE ACUERDO A LAS NECESIDADES DE CADA UNO, O BIEN EN LA DESIGNACIÓN DE HORARIOS PARA LA IMPARTICIÓN DE SESIONES A TRAVÉS DE LA PLATAFORMA ZOOM O PLATAFORMA SIMILAR, APOYAR EN LA VINCULACIÓN CON LOS ESTADOS, DEPENDENCIAS E INSTITUCIONES PARA PROMOVER PRESENTACIONES ARTÍSTICAS DE LOS BENEFICIARIOS DEL EOBA Y CONTRIBUIR EN SU DESARROLLO Y FORMACIÓN COMO CANTANTES, ADEMÁS DE DAR SEGUIMIENTO A LA REALIZACIÓN DE LAS SESIONES ORDINARIAS Y/O EXTRAORDINARIAS DEL CONSEJO DIRECTIVO DEL EOBA
CUCoPs	3390
Clave Unidad Compradora	048E00995
Unidad Compradora	INBAL-Dirección de Recursos Materiales #048E00995

URI	https://compranet.hacienda.gob.mx/esop/guest/go/opportunity/detail?opportunityId=1960205
Fecha publicación	2021-03-17 17:22:00.0

Los programas empleados para la extracción, los archivos con las claves y demás recursos empleados para la adquisición de las licitaciones públicas de México se encuentran publicados en el repositorio oficial de este proyecto.

3.2 Exploración de los datos

Después de recolectar las licitaciones, la siguiente etapa consiste en encontrar información relevante en el conjunto de datos recuperado. Una de las preguntas que se puede contestar con este análisis sería saber cuál es la distribución de las licitaciones por clasificador. Pregunta que permite saber el nivel en el que se encuentran la mayoría de las licitaciones y establecer el nivel jerárquico que el modelo podrá predecir con mayor precisión al tener más observaciones para entrenar.

Antes de explorar los registros de las licitaciones públicas, primero se explora la distribución de los clasificadores de artículos que se emplean en cada conjunto de datos.

3.2.1 Exploración de CPVs

Al explorar el catálogo del Vocabulario Común de Adquisiciones, se han encontrado las siguientes observaciones.

Se cuenta con un total de 45 términos a nivel división, el nivel más alto, y con un promedio de 210 claves por división. Siendo el término 45000000, trabajos de construcción, que presenta más claves con 822. Mientras que la clave 41000000, agua recogida y depurada, solo cuenta con 4 claves CPV asignadas a esa categoría. Al haber mucha variación de CPV por categoría, probablemente la distribución de las licitaciones se comporte de manera similar y se tenga muy pocas muestras por divisiones con pocas claves.

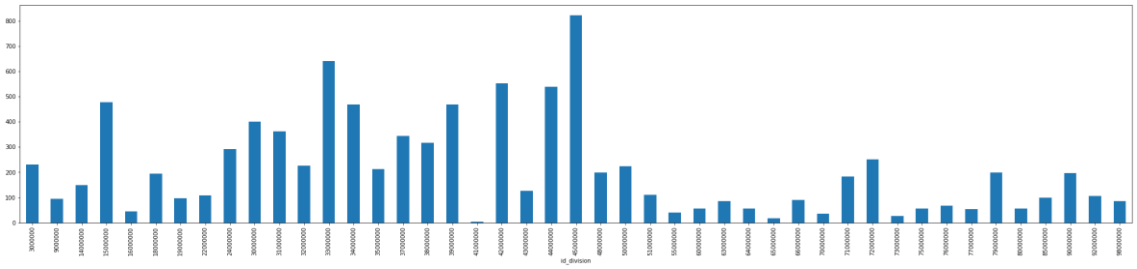

Ilustración 26: Distribución de claves CPV por división

A nivel de grupo, se presentan un total de 272 términos. En promedio, hay alrededor de 6 grupos por división. La clave 41000000, a nivel división, cuenta con un solo grupo. En cuanto a CPVs por grupo, en promedio hay 34 claves por cada grupo, la clave que más asignaciones tiene es la 45200000, trabajos generales de construcción de inmuebles y obras de ingeniería civil, que cuenta con 614. Sin embargo, la mayoría de los grupos tienen entre 1 y 20 claves asociadas. Solo hay 17 grupos que superan más de 100 CPVs.

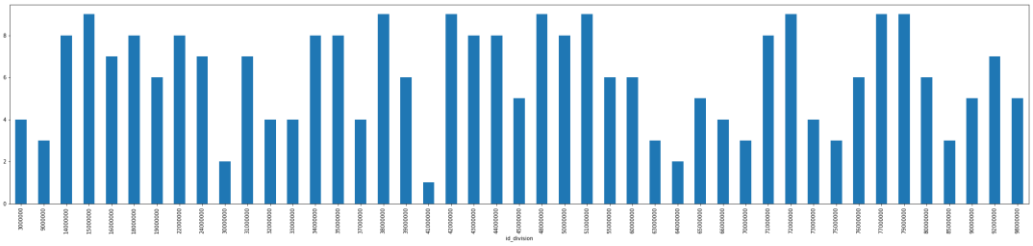

Ilustración 27: Distribución de grupos por división

El siguiente nivel inferior es la clase, en este nivel se presentan 1,004 clases distintas. Cada división cuenta con aproximadamente 22 clases. La división que

menos clases presenta es la 41000000 con 2 y la división que más clases presenta es la 48000000, paquetes de software y sistemas de información, con 47 clases. En este nivel ya es más complicado observar la distribución de CPVs por clase.

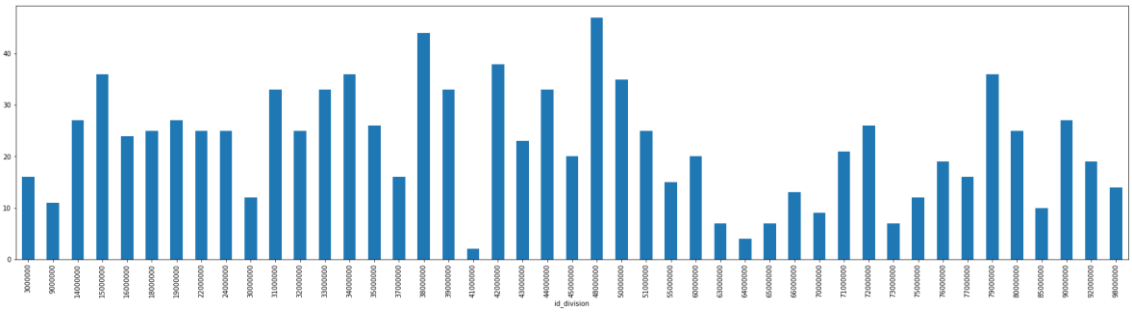

Ilustración 28: Distribución de clases por división

El nivel más bajo corresponde a la categoría, en esta hay 2,385 términos distintos. Cada división presenta alrededor de 54 categorías. La división que menos categorías tiene es la 65000000, servicios públicos, con solamente 4. La clave de división 33000000 (equipamiento y artículos médicos, farmacéuticos y de higiene personal), en cambio, presenta 162 categorías distintas.

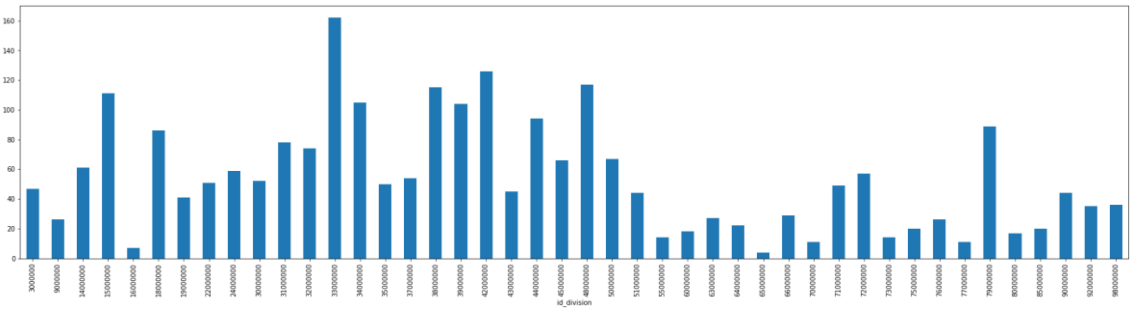

Ilustración 29: Distribución de categorías por división

En resumen, hay un total de 9,462 claves CPVs distintas, distribuidas en un árbol jerárquico que comienza con 45 divisiones, estas se dividen en 272 grupos, luego en 1,004 clases, posteriormente en 2,385 categorías y finalmente en 5,756 claves de niveles inferiores.

3.2.2 Exploración de CUCoPs

En el caso de los clasificadores CUCoPs, hay 14,813 términos distintos y al igual que el vocabulario CPV, se distribuyen en un árbol de jerarquía. El nivel más alto es el capítulo y cuenta con 4 términos, dichos capítulos se dividen en conceptos, en ese nivel hay 30 términos. Luego, los conceptos se dividen en 197 partidas genéricas. Posteriormente, las partidas genéricas se componen por 296 partidas específicas y éstas a su vez se dividen en 14,286 claves en un nivel inferior. A continuación, se describen las claves por cada nivel jerárquico.

El nivel más alto, representado por el capítulo, hay 4 claves, 2000, materiales y suministros; 3000, servicios generales; 5000, bienes muebles, e intangibles y 6000, inversión pública. El capítulo 2000 es aquel que presenta más claves asignados con 8,413 y el capítulo 3000 el que menos tiene con 479.

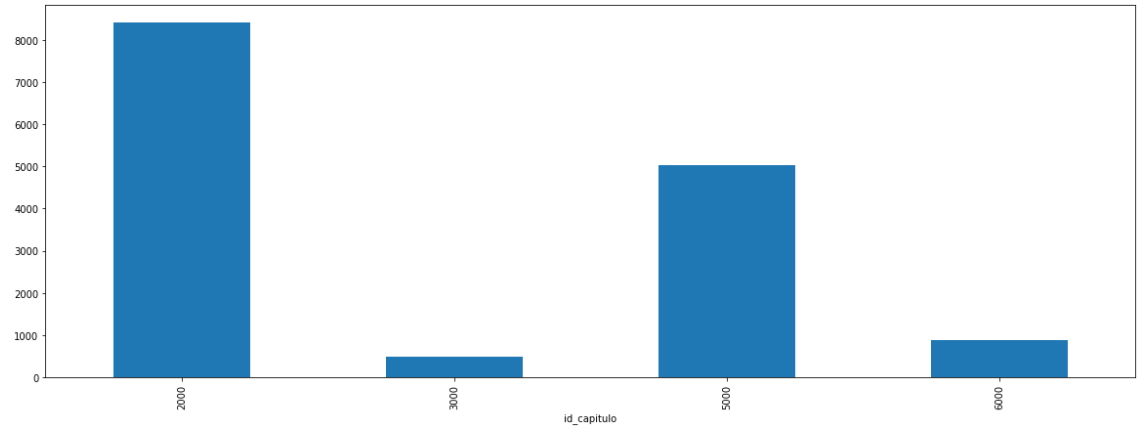

Ilustración 30: Distribución de claves por capítulo

A nivel concepto, un nivel inferior a capítulo, se tiene 30 términos. Cada capítulo cuenta con 9 conceptos, excepto el capítulo 6000 que solamente cuenta con 3. La distribución de claves CUCoPs por concepto es muy diversa, ya que hay conceptos con pocas claves, tal es el caso del concepto 6300, proyectos productivos y acciones de fomento, que solo cuenta con 3 claves asociadas y otros con demasiadas, como el concepto 2500, productos químicos, farmacéuticos y de laboratorio, que cuenta con 4,607 claves asociadas a este concepto.

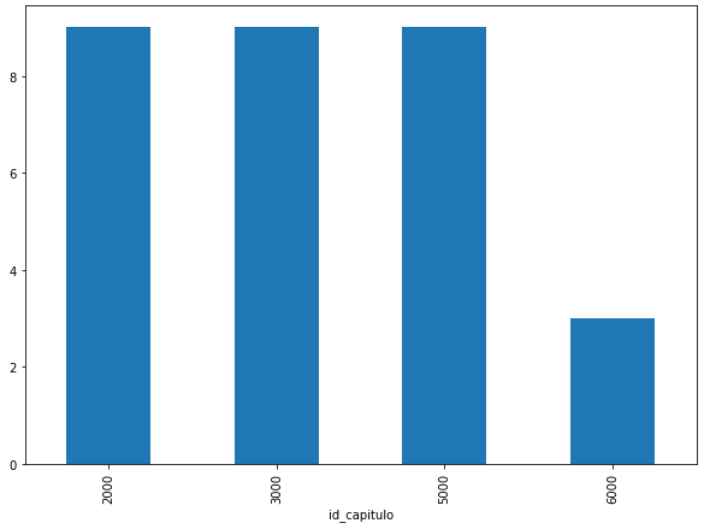

Ilustración 31: Distribución de conceptos por capítulo

En el caso del nivel de partida genérica, hay 197 claves que pertenecen a este nivel. El capítulo que cuenta con menos partidas genéricas asociadas es la 6000 con 18, mientras que aquel capítulo que más presenta es el 3000 con 75 partidas genéricas. La mayoría de las partidas genéricas tienen menos de 100 códigos asociados, pero hay uno en particular que presenta demasiados, alrededor 3,717 y corresponde a la partida genérica 2530, medicinas y productos farmacéuticos, del concepto 2500, productos químicos, farmacéuticos y de laboratorio.

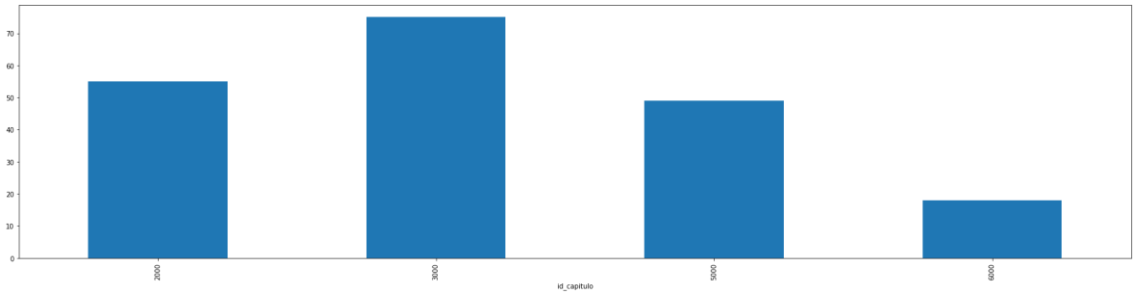

Ilustración 32: Distribución de partidas genéricas por capítulo

A un nivel más bajo que la partida genérica se encuentra la partida específica, en este nivel hay 296 términos. El capítulo que menos partidas específicas tiene es el 6000 con 27 y el que más presenta es el capítulo 3000 con 140. En cuanto a códigos inferiores asignados a este nivel, el término 25301, medicinas y productos farmacéuticos, es el que presenta más claves con alrededor de 3,716.

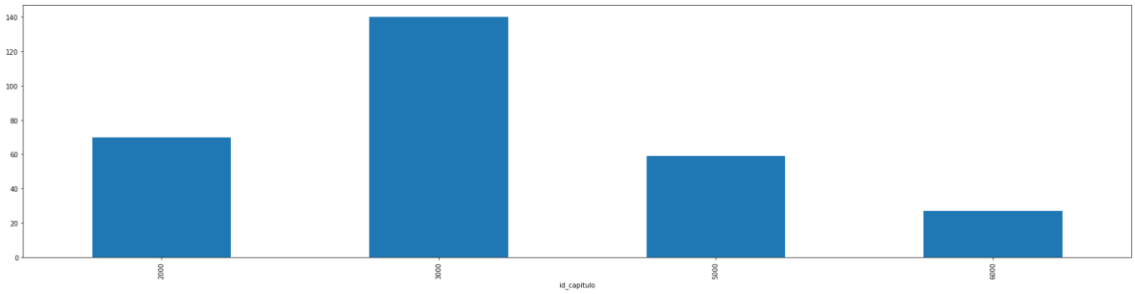

Ilustración 33: Distribución de partidas específicas por capítulo.

3.2.3 Exploración de licitaciones CPV

En este apartado se explorará los registros de las licitaciones que se han recolectado de España, estos registros presentan el vocabulario que se emplea en la Unión Europea, CPV.

En total se han recogido 511,180 licitaciones públicas en el periodo comprendido entre 2012 y abril de 2022 del portal de datos abiertos del Ministerio de Hacienda. De este total, existen 5,800 registros que no presentan asignación de una clave CPV, siendo dicho código un dato esencial para este proyecto, por lo tanto, para siguientes fases estos registros se descartan. Del restante, se tienen 117,985 licitaciones con más de un solo código CPV asignado, para estos registros se podría generar duplicados de las descripciones por cada código CPV y contar con esos registros para entrenar los modelos de aprendizaje automático, sin embargo, para el alcance de este proyecto solo se consideran aquellos que cuentan con un único CPV por licitación. El número total de licitaciones con solo un término CPV son 387,395 y representan más de un 75% del total de los registros recuperados. Por lo anterior, para la etapa de exploración y preprocesamiento de los datos solo se consideran los registros mencionados anteriormente.

Del total de las 387,395 licitaciones, todas presentan un código asociado al nivel de división. En el caso del nivel de grupo, solamente 344,532 registros presentan correspondencia con este nivel. Lo mismo ocurre a nivel de clase, donde ya solamente hay 282,452 registros que tienen códigos que se encuentran a este nivel. Finalmente, en el caso de la categoría, el número de registros cae a 193,798 licitaciones.

Por lo anterior, podemos concluir que los encargados de asignar una clave CPV a las licitaciones, prefieren asignar códigos más genéricos sobre los específicos.

Posteriormente, se realiza un análisis de la distribución de las licitaciones por cada nivel jerárquico de CPV y se ha encontrado que, a nivel división, la clave 45000000, trabajos de construcción, es la división con más licitaciones asignadas, ya que cuenta con 65,291. Mientras que el código 76000000, servicios relacionados con la industria del gas y del petróleo, cuenta con 39 licitaciones, siendo la división con menos licitaciones recolectadas, es probable que esto se deba al hecho de que hay pocas entidades públicas que soliciten este tipo de servicios.

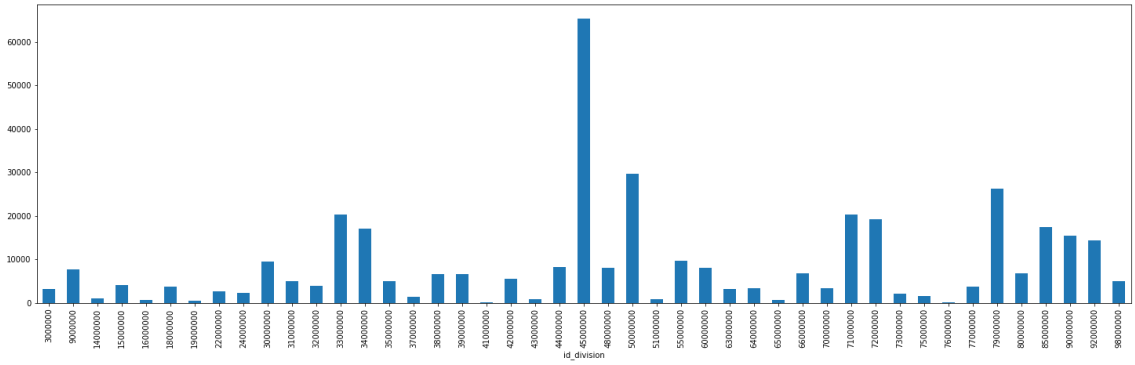

Ilustración 34: Distribución de licitaciones por división

A nivel de grupo, la distribución de las licitaciones se vuelve menos uniforme. Por ejemplo, los grupos de la división 45000000 tienen licitaciones muy dispersas, ya que el grupo 45200000, trabajos generales de construcción de inmuebles y obras de ingeniería civil, es la clave con más licitaciones asignadas, cuenta con alrededor de 44,693 registros y en esa misma división el grupo 45500000, alquiler de maquinaria y equipo de construcción y de ingeniería civil con maquinista, solamente cuenta con 278 licitaciones, una diferencia muy significativa. Se podría decir que, la mayoría de las administraciones públicas solicitan los servicios de construcción a empresas privadas en lugar de rentar la maquinaria necesaria para llevar a cabo dicho trabajo.

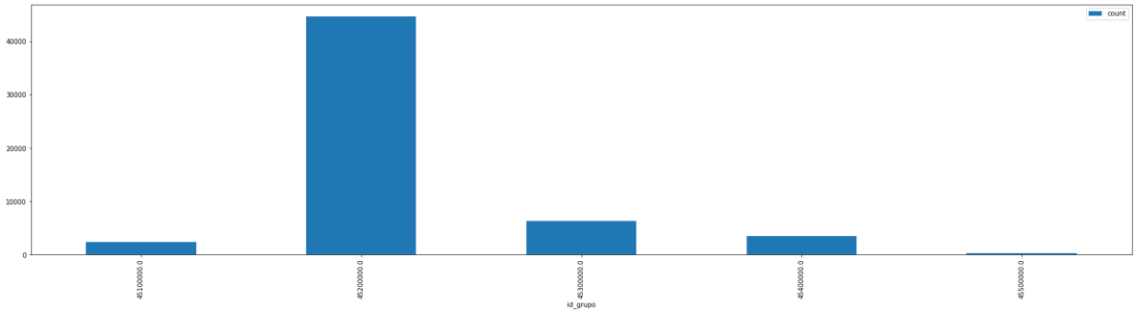

Ilustración 35: Distribución de licitaciones por grupo en la división 45000000

Por otro lado, en el mismo nivel de grupo, solamente 4 de las 45 divisiones se cumple que, cada uno de los grupos asignados a éstas cuenta con al menos 1,000 registros. Estas divisiones son: 30000000, 50000000, 64000000 y 79000000. A nivel de clase, no existe división que en cada una de sus clases tengan al menos 500 registros.

Por todo lo anterior, el nivel más bajo con el que se podría entrenar un modelo corresponde al nivel de grupo, pero solamente para ciertas divisiones, aquellas que cuenten con registros suficientes, unos 1,000 registros por grupo. Sin

embargo, para el alcance de este proyecto solo se contempla la creación de modelos de clasificación a nivel división.

3.2.4 Exploración de licitaciones CUCoP

En esta sección se exploran las licitaciones que se han recuperado de los portales abiertos de México y que emplean el vocabulario mexicano conocido como Clasificador Único de Contrataciones Públicas (CUCoP).

De los dos repositorios recolectados **expediente** y **licitacion_mx.** Se tiene lo siguiente. De la tabla **licitacion_mx** se han recuperado 300,265 licitaciones, sin embargo, la mayoría de los registros (206,093) no presenta una clave CUCoP. Aquellas que sí lo presentan son solamente 94,172, estas corresponden a 80,954 que presentan un único término y 13,218 con más de uno.

En el caso de la tabla **expediente** se recuperó un total de 654,109 licitaciones y todas cuentan con al menos un código CUCoP asignado. Hay 626,168 registros con una sola clave y 27,941 son los que presentan más de una.

Cabe mencionar que ambos repositorios comparten 117,966 registros, y considerando que la tabla **licitacion_mx** solamente cuenta con 94,172 registros que pueden ser útiles para este trabajo, se ha optado por procesar aquellos de la tabla **expediente** que presenta mayor cantidad de registros con potencial para este proyecto.

Del total de 626,168 registros, todos tienen asociado una clave a nivel de capítulo. A nivel de concepto, de igual manera todos los registros presentan una clave para este nivel. Lo mismo ocurre para el nivel de partida genérica. Sin embargo, para niveles inferiores no hay claves asociadas. Por lo que el nivel más bajo alcanzado por estos registros en el árbol de jerarquía es el nivel de partida genérica.

En cuanto a la distribución de las licitaciones a nivel capítulo, se tiene que los capítulos 2000 (materiales y suministros) y 3000 (servicios generales) presentan alrededor de 27,000 registros cada uno y el capítulo que menos registros presenta es el 5000 (bienes muebles, e intangibles) con solo 17,605.

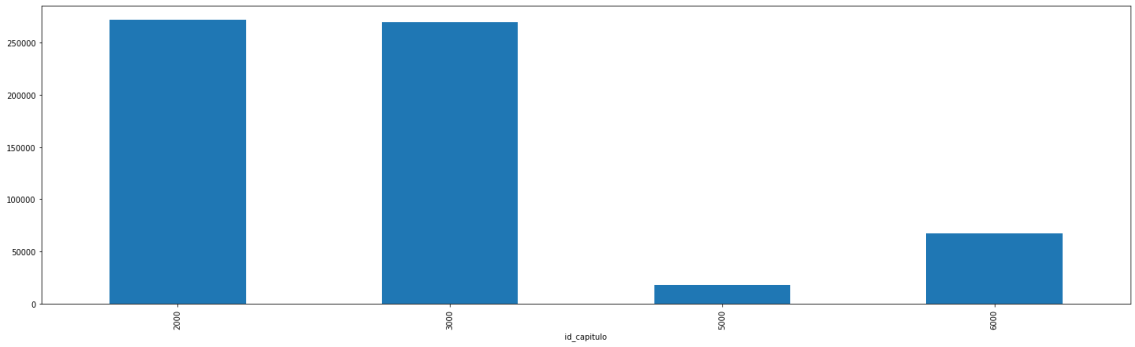

Ilustración 36: Distribución de licitaciones a nivel capítulo

A nivel concepto, se tiene que hay 4 claves que superan los 50,000 registros, estas claves son: 3500, servicios de instalación, reparación, mantenimiento y conservación, con 64,458; el concepto 2500, productos químicos, farmacéuticos y de laboratorio, con 95,718; la clave 2300, materias primas y materiales de producción y comercialización, con 108,849 registros; y el concepto con más registros es la clave 3300, servicios profesionales, científicos, técnicos y otros servicios, con un total de 154,451. Por otro lado, existen 6 claves de nivel concepto que no superan los 1,000 registros y estas son: 5200, mobiliario y

equipo educacional y recreativo, con 932; 2800, materiales y suministros para seguridad, con 611; 5800, bienes inmuebles, con 583; 5900, activos intangibles, que tiene 420; con 137 la clave 5500, equipo de defensa y seguridad y la clave con menos registros es la 5700, activos biológicos, con solamente 62 licitaciones. Es decir, solo los capítulos 3000 y 6000 cuentan con grupos asignados donde cada uno tiene al menos 1,000 registros.

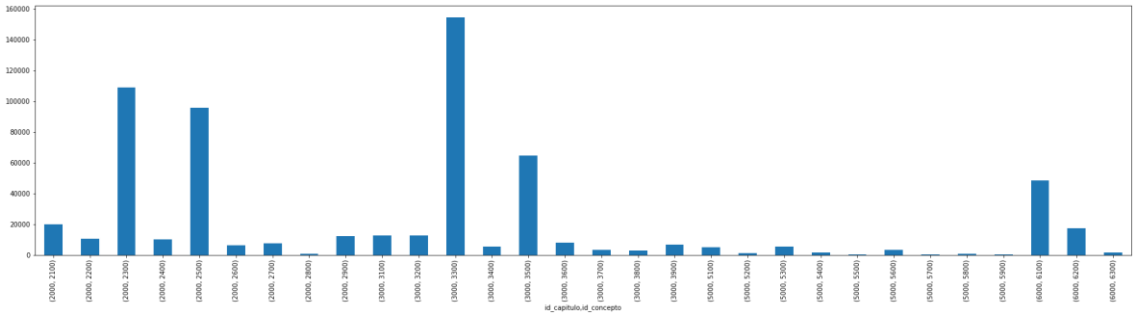

Ilustración 37: Distribución de licitaciones a nivel concepto

A nivel partida genérica hay dos claves que sobresalen entre los demás términos ya que presentan más de 100,00 registros cada una. Estas claves son 2380 (mercancías adquiridas para su comercialización) y 3390 (servicios profesionales, científicos y técnicos integrales).

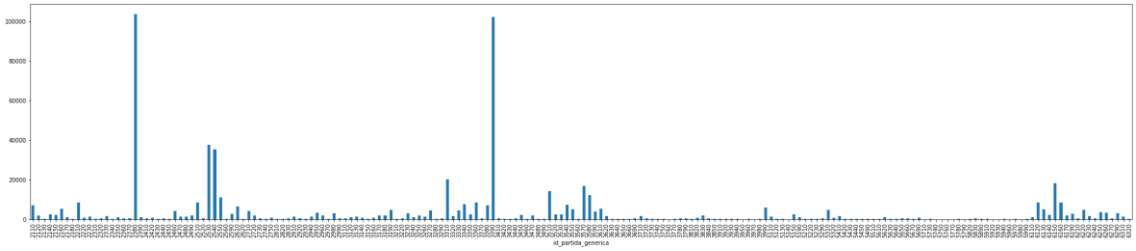

Ilustración 38: Distribución de licitaciones a nivel partida genérica

El *notebook* de Google Colaboratory empleado para realizar el análisis exploratorio se encuentra en el repositorio principal del proyecto.

3.3 Preprocesamiento de los datos

Después de realizar una exploración por los registros de las licitaciones, se procede a la etapa de preprocesamiento y limpieza de los datos, específicamente esto solo se ha de realizar para el atributo objeto del contrato, cuyo contenido es la descripción textual del bien o servicio adquirido por cada licitación. La finalidad principal de aplicar esta limpieza es presentar a los modelos de *machine learning* textos uniformes o normalizados para que, al vectorizar las sentencias, el modelo se enfoque principalmente en encontrar patrones en los datos y no en el formato.

Este preprocesamiento se ha realizado a través de Google Colaboratory y utilizando Python como lenguaje de programación. Además, se ha de efectuar de la misma manera para ambos conjuntos de datos, ya que ambos están en idioma español. Las técnicas empleadas son las que se emplean regularmente en el procesamiento de lenguaje natural. La metodología de limpieza o *pipeline* que se aplica a las sentencias consiste en:

- **Lowercasing**: Convertir las descripciones a texto en minúsculas.
- **Tokenization**: Separar las sentencias en *tokens* o palabras.
- **Noise removal:** Eliminar *tokens* que correspondan a signos de puntuación o que no sean caracteres alfabéticos. Además, se eliminan palabras demasiado cortas, cuyo tamaño sea menor o igual a tres caracteres.
- **Stop-word removal:** Eliminar palabras vacías y sin significado *stopwords.*
- **Lemmatization:** Lematizar las palabras, convertir las palabras a su forma canónica.

A modo de ejemplo, se mostrará los resultados al aplicar cada una de las técnicas de procesamiento al siguiente texto que corresponde al objeto del contrato de una licitación.

```
SERVICIOS PROFESIONALES CONSISTENTES EN LA COORDINACIÓN DEL PROGRAMA CULTURAL
DEL PROYECTO BOSQUE DE CHAPULTEPEC NATURALEZA Y CULTURA, A TRAVÉS DE SUS EJES:
LA CONEXIÓN ENTRE LO BIOLÓGICO Y LO CULTURAL, UN PLAN INTEGRAL DE MOVILIDAD
ENTRE LAS CUATRO SECCIONES Y SU ENTORNO URBANO, LA PROYECCIÓN DE UN ESPACIO DE
POLÍTICA AMBIENTAL Y ESPACIO PÚBLICO CULTURAL, Y UN ESPACIO PÚBLICO CON DIVERSA
OFERTA CULTURAL, HISTÓRICA, AMBIENTAL Y RECREATIVA PARA FAVORECER EL DESARROLLO
DE ACTIVIDADES QUE MEJOREN LA CALIDAD DE VIDA DE LA POBLACIÓN DE TODO EL PAÍS
E IMPULSE LA CONVIVENCIA E INCLUSIÓN SOCIAL Y DIVERSA.
```

La primera técnica consiste en convertir las descripciones a minúsculas, para ello se hace uso del método *lower* que retorna una copia de la cadena con todos los caracteres en minúsculas, este método está definido en la biblioteca estándar de Python para el manejo de cadena de caracteres. Posteriormente, se *tokenizan* cada una de las descripciones, la *tokenization* divide las sentencias en una lista de palabras que conforman las descripciones. Después de realizar estas dos operaciones al texto de ejemplo, el resultado es el siguiente:

```
['servicios', 'profesionales', 'consistentes', 'en', 'la', 'coordinación',
'del', 'programa', 'cultural', 'del', 'proyecto', 'bosque', 'de', 'chapultepec',
'naturaleza', 'y', 'cultura', ',', 'a', 'través', 'de', 'sus', 'ejes', ':',
'la', 'conexión', 'entre', 'lo', 'biológico', 'y', 'lo', 'cultural', ',', 'un',
'plan', 'integral', 'de', 'movilidad', 'entre', 'las', 'cuatro', 'secciones',
'y', 'su', 'entorno', 'urbano', ',', 'la', 'proyección', 'de', 'un', 'espacio',
'de', 'política', 'ambiental', 'y', 'espacio', 'público', 'cultural', ',', 'y',
'un', 'espacio', 'público', 'con', 'diversa', 'oferta', 'cultural', ',',
'histórica', ',', 'ambiental', 'y', 'recreativa', 'para', 'favorecer', 'el',
```

```
'desarrollo', 'de', 'actividades', 'que', 'mejoren', 'la', 'calidad', 'de',
'vida', 'de', 'la', 'población', 'de', 'todo', 'el', 'país', 'e', 'impulse',
'la', 'convivencia', 'e', 'inclusión', 'social', 'y', 'diversa', '.']
```

Una vez se tiene el texto dividido en *tokens*, se suprimen aquellos que su tamaño sea igual o inferior a tres caracteres o que correspondan a signos de puntuación. El resultado de realizar dichas operaciones al texto de ejemplo sería el que se muestra a continuación.

```
['servicios', 'profesionales', 'consistentes', 'coordinación', 'programa',
'cultural', 'proyecto', 'bosque', 'chapultepec', 'naturaleza', 'cultura',
'través', 'ejes', 'conexión', 'entre', 'biológico', 'cultural', 'plan',
'integral', 'movilidad', 'entre', 'cuatro', 'secciones', 'entorno', 'urbano',
'proyección', 'espacio', 'política', 'ambiental', 'espacio', 'público',
'cultural', 'espacio', 'público', 'diversa', 'oferta', 'cultural', 'histórica',
'ambiental', 'recreativa', 'para', 'favorecer', 'desarrollo', 'actividades',
'mejoren', 'calidad', 'vida', 'población', 'todo', 'país', 'impulse',
'convivencia', 'inclusión', 'social', 'diversa']
```

Después de mantener los *tokens* que no son signos de puntuación o que su tamaño es mayor a tres caracteres, se procede a realizar operación que consiste en eliminar *stopwords* o palabras vacías, cuyo uso muy común en el idioma español hace que no aporten significado por sí mismas. Si esta operación se realiza en el texto de ejemplo el resultado es el siguiente:

```
['servicios', 'profesionales', 'consistentes', 'coordinación', 'programa',
'cultural', 'proyecto', 'bosque', 'chapultepec', 'naturaleza', 'cultura',
'través', 'ejes', 'conexión', 'biológico', 'cultural', 'plan', 'integral',
'movilidad', 'cuatro', 'secciones', 'entorno', 'urbano', 'proyección',
'espacio', 'política', 'ambiental', 'espacio', 'público', 'cultural',
'espacio', 'público', 'diversa', 'oferta', 'cultural', 'histórica',
'ambiental', 'recreativa', 'favorecer', 'desarrollo', 'actividades', 'mejoren',
'calidad', 'vida', 'población', 'país', 'impulse', 'convivencia', 'inclusión',
'social', 'diversa']
```

La última técnica de preprocesamiento y limpieza que se ha empleado a las descripciones ha sido la *lemmatization*, método que consiste en encontrar el lema correspondiente para una palabra o *token*, se podría decir que un lema es la forma normal de las palabras y de ellas se derivan las demás. Al buscar el lema de las palabras del texto de ejemplo y generar la nueva descripción se obtiene el siguiente resultado:

```
servicio profesional consistent coordinación programa cultural proyecto bosque
chapultepec naturaleza cultura través eje conexión biológico cultural plan
integral movilidad cuatro sección entorno urbano proyección espacio político
ambiental espacio público cultural espacio público diverso oferta cultural
histórico ambiental recreativo favorecer desarrollo actividad mejorar calidad
vida población país impulse convivencia inclusión social diverso
```

Hasta este punto, se ha implementado una metodología para normalizar las descripciones que funciona perfectamente y otorga los resultados esperados, sin embargo, se tiene que considerar que los métodos implementados sean eficientes en cuanto al tiempo de procesamiento, esto debido a la gran cantidad de registros que se tienen que procesar: un conjunto de datos presenta casi 700 mil registros. Al tener grandes conjuntos de datos, el preprocesamiento se ha de realizar primero y antes de implementar cualquier modelo para almacenarlos en un nuevo conjunto de datos a fin de utilizarlos posteriormente sin tener que esperar nuevamente el tiempo de limpieza. La primera versión del método que se implementó para realizar el preprocesamiento tarda alrededor de 1 segundo en procesar cada descripción, lo que significa que para procesar todo el conjunto

de datos requiere alrededor de 194 horas, demasiado tiempo. El código de esta versión es el siguiente:

```
def preprocessing(text):
  des_removed_punc = remove_punct(text.lower())
  tokens = nltk.word_tokenize(des_removed_punc, "spanish")
  larger_tokens = remove_small_words(tokens)
  clean_tokens = remove_stopwords(larger_tokens)
  lemma_words = lemmatize_es(clean_tokens)
  clean_text = return_sentences(lemma_words)
  return clean_text
```

Debido a que no era viable tener un método que requiera más de una semana en procesar y limpiar todas las descripciones. Se ha desarrollado una nueva versión que realiza y aplica las mismas técnicas de procesamiento natural, solo que ahora reduce el número de operaciones al efectuar un único ciclo en el listado de *tokens*. El tiempo de ejecución por descripción pasó de 1 segundo a 0.02 segundos, resultando en un tiempo total de preprocesamiento de 4 horas en total. Tiempo que se ha considerado aceptable para los fines de este trabajo. El código de la versión empleada es el siguiente:

```
def preprocessing_fastest_v2(text):
  #tokenizamos words
  token_word = nltk.word_tokenize(text.lower(), "spanish")
  sentence = []
  for token in token_word:
    #omit tokens if are less than 4 characters or are punctuation marks
    if len(token) <= 3 or token in st.punctuation:
      continue
    #omit if the token has numbers
    if not (token.isalpha()):
      continue
    #omit tokens if are stopwords in spanish
    if token in nltk.corpus.stopwords.words('spanish'):
      continue
    sentence.append(token)
  #lematization of sentence
  doc = sp(' '.join(sentence))
  return ' '.join([token.lemma_ for token in doc])
```

Finalmente, en esta etapa se ha hecho la limpieza de los registros y el nuevo conjunto de datos parece estar listo para emplearse en el modelado, sin embargo, al realizar la limpieza, resultó en nuevos registros con valores vacíos o nulos en el atributo objeto del contrato. Para evitar ingresar estos datos como ruido en la etapa de modelado, se han de suprimir los registros y solo mantener aquellos que sí proporcionan información útil a los modelos.

Los *notebooks* que se han empleado para el preprocesamiento y limpieza de los registros se pueden encontrar en el repositorio del proyecto.

3.4 Modelado y evaluación

En este apartado se presentan los modelos de aprendizaje automático que se han implementado para cada uno de los conjuntos de datos y su respectivo desempeño en la predicción de la clase para nuevas instancias, esta clase corresponde a los códigos de los clasificadores de artículos tanto del vocabulario empleado en la Unión Europea como el utilizado en México. Además, se describe brevemente, las diferentes técnicas para codificar las descripciones textuales de los objetos del contrato, los algoritmos de inteligencia artificial implementados y las métricas que se han utilizado para evaluar el desempeño de los modelos.

Partiendo del hecho de que, cada clasificador de artículos contiene miles de términos distintos, en lugar de crear un clasificador binario para cada uno de los posibles códigos, se aprovecha la estructura jerárquica de las claves para crear un clasificador multiclase de las categóricas más genéricas para CPV o CUCoP. La principal ventaja de esto es que permite utilizar más instancias como datos de entrenamiento para una categoría, al tener que agrupar los registros solamente en las claves más genéricas y, además, el modelo resultante sería capaz de predecir las distintas categorías establecidas.

Por otro lado, para poder transformar el corpus, contenido textual de las descripciones de cada licitación, a valores numéricos que una computadora pueda entender y procesar, se han empleado estrategias denominadas vectorización o codificación. Estas técnicas nos permiten transformar el texto del objeto del contrato en un vector numérico que servirá como entrada para los algoritmos de clasificación. Dependiendo de la técnica, el resultado o *performance* de un algoritmo puede variar, ya que cada método puede codificar en mayor o menor medida ciertas particularidades de un texto. Las técnicas que se han implementado en este proyecto se describen a continuación, además se presentan las características principales que estas codifican:

- ***CountVectorizer:*** es una de las maneras más sencillas de codificar las palabras, ya que cada descripción de una licitación es representada por un vector con todas las palabras del corpus y el número de veces que esta aparece en la oración, es decir, codifica la frecuencia de ocurrencia de cada término (palabras) en un documento (descripción de una licitación). Con esta técnica, básicamente se codifica el número de veces que una palabra aparece en la descripción.
- **TF-IDF:** Esta otra técnica es más sofisticada que la anterior, puesto que, además de contar con la frecuencia de cada término del corpus, se codifica cuán relevante es una palabra en un documento, siendo más relevante si el término tiene poca frecuencia. Con esta técnica una palabra poco usada o rara será más relevante que otras más comunes.
- **Word2Vect:** Es un método más avanzado que los anteriores, ya que utilizando esta técnica se crea una representación vectorial única para cada palabra. Ahora se toma en cuenta el contexto de las palabras y se puede encontrar si existe relación entre ellas o no. Por ejemplo, gracias a esta técnica podemos encontrar sinónimos y antónimos de las palabras utilizadas en las descripciones. Para vectorizar toda una sentencia, comúnmente se promedia los vectores de cada palabra presente en la oración generando un nuevo vector que representa la descripción completa.
- **Sentence Transformer (BERT):** Es una de las técnicas más avanzadas en el procesamiento de lenguaje natural para vectorizar sentencias de un texto, a diferencia del método anterior, la sentencia es vectorizada como un todo y no palabra por palabra. Con esta forma de vectorizar las sentencias, el orden y el contexto de cada palabra es representado de una

mejor manera, por lo que, con este método podemos encontrar oraciones que comparten el mismo significado, es decir, las sentencias podrían utilizar palabras distintas para transmitir la misma idea.

En cada una de las estrategias empleadas para la representación vectorial de las descripciones se implementan los siguientes algoritmos de aprendizaje supervisado, mismos que permiten modelar problemas de clasificación multiclase o multi-etiqueta, además puedan emplear los vectores numéricos previamente generados como datos de entrada, ya que estos vectores son de dimensiones muy grandes y finalmente que los algoritmos puedan procesar la gran cantidad de registros en tiempos de entrenamiento y predicción considerables:

- **MultinomialNB:** El algoritmo de clasificación multinomial Naïve Bayes es adecuado para modelos de clasificación de texto con datos de entrada discretos como bag-of-words o TF-IDF.
- **SVM:** Utilizando este algoritmo las instancias de las clases son representadas en un espacio de mayor dimensión para ser divididas por hyperplanos donde cada región de la división representa una clase correspondiente. Las nuevas instancias son clasificadas de acuerdo con la región en la que sea representada.
- **SGDClassifier:** Es un algoritmo de clasificación lineal que emplea *stochastic gradient descent* (SGD) como técnica de optimización e implementa métodos de regularización. Por su eficiencia, este algoritmo es adecuado para entrenar modelos de grandes conjuntos de datos.
- **KNN:** Es un algoritmo simple, cuya idea principal es clasificar una nueva instancia evaluando el número de K instancias más cercanas y clasificarla de acuerdo con la clase más común de las K instancias, que ya cuentan con una clasificación.
- **Random Forest.** Es un algoritmo que se usa ampliamente para tareas de clasificación, y está basado en conjuntos de árboles de decisiones.

Algunos de los algoritmos que se han descartado de este proyecto después de realizar algunas pruebas han sido *KNN* y *RandomForest*, pues la desventaja principal de KNN es el tiempo de predicción y en el caso del *RandomForest* el tiempo de entrenamiento. En ambos casos debido a la gran cantidad de registros con las que se cuenta.

Adicionalmente, para cada algoritmo se implementa la técnica *GridSearchCV* para realizar el ajuste de hiper-parámetros y encontrar los valores de los parámetros que permitan maximizar su desempeño. Esta técnica emplea la estrategia de división de validación cruzada (*Cross Validation)* para obtener un valor promedio justo del *performance* del modelo para cada combinación de parámetros.

Al tener que modelar un problema multiclase, los datos se tratan como una colección de problemas binarios y con la finalidad de evaluar el desempeño de cada modelo compuesto se han utilizado las métricas que se describen a continuación. Además, considerando que las clases no se encuentran balanceadas, tal como se vio en el apartado de exploración, estas técnicas permiten promediar justamente el desempeño de los modelos compuestos por problemas binarios.

- **Precision:** Mide la calidad de la predicción, el porcentaje que el modelo ha predicho como una clase y que efectivamente sea esa clase.
- **Recall:** Mide la cantidad de la predicción, el porcentaje de la clase que el modelo ha podido identificar.

- **F1-Score:** Es el resultado de la combinación de *precision* y *recall*, es la media armónica entre ambas métricas.
- **Acuracy:** Mide el porcentaje de casos que el modelo ha predicho correctamente.
- **Balanced Acuracy:** Mide el porcentaje de casos que el modelo ha predicho correctamente tomando en cuenta conjuntos de datos no balanceados. Se define como el promedio de la métrica *recall* por cada clase.
- **Confusion Matrix:** Para visualizar los falsos positivos contra los falsos negativos.

Para las métricas *precisión, recall* y *F1-Score* se utilizan los métodos *weighted-average* y *micro-average* para promediar el desempeño de cada modelo. Estos tipos de promedio son utilizados para tener en cuenta la distribución de las clases en los datos. En el caso de *weighted,* el promedio es calculado considerando el peso que tiene cada clase, siendo más alto si presenta más ocurrencias en los datos. Mientras que *micro* es utilizado para calcular la proporción de observaciones correctamente clasificadas, tomando en cuenta los Verdaderos Positivos, Falsos Negativos y los Falsos Positivos. Cuando se tiene un problema multiclase donde cada registro presenta una única clase, como es el caso en este proyecto, entonces el valor es el mismo para las métricas *accuracy, micro-F1-score, micro-precision* y *micro-recall*.

Finalmente, como el objetivo de este proyecto es generar un modelo que mejor prediga correctamente nuevas instancias, una de las técnicas más comunes para simular nuevos datos es dividir los conjuntos iniciales en subconjuntos a fin de contar con datos de entrenamiento y datos de prueba para la evaluación. En este trabajo, la relación de la división de los conjuntos de datos es 80 a 20, es decir, el 80% de los registros de las licitaciones se empleará como datos de entrenamiento y el restante como datos de prueba. Además, sabiendo que los conjuntos de datos presentan clases no balanceadas, la división en conjuntos de *training* y *testing* se realiza de manera estratificada, es decir, se mantiene la proporción entre las clases tanto en los datos de entrenamiento como en los datos de prueba.

El código empleado en la etapa de modelación se ha realizado en varios *notebooks* de Google Colaboratory y se pueden encontrar en el repositorio principal de este trabajo.

3.4.1 Modelado y evaluación de códigos CPV

En esta parte se muestran los modelos desarrollados para los registros que corresponden a licitaciones CPV. Debido a que no hay la cantidad suficiente de registros para entrenar un modelo de clasificación a nivel de grupo, se ha optado por elegir la división como el nivel jerárquico a predecir. La cantidad de clases que el modelo podrá predecir corresponde con la cantidad de divisiones, es decir, 45 categorías distintas. Además, para simplificar la representación, se eliminan los ceros de las claves CPV. Por ejemplo, la división 45000000 pasa a ser la división 45.

El conjunto total de registros de licitaciones es de 387,194. Estos registros se han dividido en 309,755 para la etapa de *training* y 77,439 para *testing.*

3.4.1.1 Técnica de vectorización CountVectorizer

Para aplicar la técnica de countVectorizer, se ha empleado el código que aparece a continuación. Como argumentos importantes, se ha utilizado el parámetro *strip_accents* para eliminar los acentos de las palabras y así normalizar aquellas que no han sido escritas con acentos. También, se utiliza el parámetro *ngram_range* para incrementar el vocabulario y en lugar de tener una palabra se contempla un conjunto, es decir, se captura un poco de contexto.

```
CountVectorizer(analyzer="word",
                ngram_range = (1,3),
                tokenizer=word_tokenize,
                strip_accents='ascii',
                max_features=None,
                lowercase=True)
```

Utilizando el algoritmo ***MultinomialNB***, los parámetros que se han configurado para la búsqueda del mejor modelo son: *ngram_range*, de la técnica de vectorización y el parámetro *alpha.* El parámetro *alpha* permite suavizar datos categóricos utilizando la técnica *additive smoothing.* El valor 0 significa no realizar dicho proceso. Los valores para el ajuste de hiper-parámetros son los siguientes:

```
parametersMNB = {
                 'vectorizer__ngram_range': [(1,1), (1,2), (1,3)],
                 'MNB__alpha':[0, 0.5, 1.0]
                 }
```

La combinación de parámetros que mejor desempeño obtuvo fue el modelo que emplea *ngram_range= (1,3)* y *alpha=0.* El tiempo que el algoritmo dedicó a la etapa de entrenamiento fue de 90 segundos y el *accuracy* obtenido fue de 75% en promedio. El *accuracy* logrado al evaluar el modelo con los datos para *testing* es de 77%. A continuación, se muestra el desempeño de cada modelo en la etapa de ajuste de hiper-parámetros.

Tabla 8: Ajuste de parámetros del modelo MultinomialNB - CountVectorizer

ngram range	alpha	Training time (s)	Accuracy
(1,1)	0	63.69	0.69
(1,1)	0.5	61.55	0.70
(1,1)	1	61.41	0.66
(1,2)	0	72.27	0.72
(1,2)	0.5	72.76	0.72
(1,2)	1	72.03	0.67
(1,3)	**0**	**88.17**	**0.75**
(1,3)	0.5	87.01	0.73
(1,3)	1	82.83	0.69

Para el algoritmo ***LinearSVM***, se ha empleado los siguientes parámetros para la creación del modelo base.

```
modelSVM = LinearSVC( C=1.0,
                      penalty='l2',
                     class_weight='balanced',
                     random_state=42,
                     max_iter=1000,
                     verbose=3)
```

Antes de realizar el ajuste de hiper-parámetros se optado por verificar el comportamiento de los siguientes parámetros:

- *Class_weight:* Este parámetro permite ajustar la importancia que se le da a una clase y depende del valor C.
- *Ngram_range:* Argumento del método countVectorizer.

```
parametersSVM = {
                 'SVM__class_weight': ['balanced', None],
                 'vectorizer__ngram_range': [(1,1), (1,3)]
                 }
```

Los resultados se muestran en la siguiente tabla.

Tabla 9: Comparación de parámetros para el algoritmo LinearSVM - CountVetorizer

Class weight	ngram range	Training time (s)	Accuracy
balanced	(1,1)	224.765	0.772
balanced	**(1,3)**	**473.685**	**0.806**
None	(1,1)	229.840	0.779
None	(1,3)	433.898	0.805

Los parámetros que mejor se desempeñan son los valores (1,3) para *ngram_range* y *balanced* en el caso de *class_weight*, por lo que estos valores se mantendrán constantes en el proceso de ajuste de hiper-parámetros. Para dicho proceso se ha considerado los siguientes argumentos del modelo base:

- *C:* Es un parámetro de regularización y su fuerza es inversamente proporcional al valor establecido.
- *Max_iter:* Es el máximo número de iteraciones realizadas.

El conjunto de parámetros a probar con *GridSearchCV* es*:*

```
parametersSVM = {
                 'SVM__max_iter': [300, 500, 1000],
                 'SVM__C': [0.01, 0.1, 1]
                 }
```

El *accuracy* obtenido por la mejor la combinación de parámetros (500 y 0.1) en el conjunto de *training* es de 82%, mientras que para el conjunto de *testing* es de 84%. El tiempo de entrenamiento fue de 542 segundos. El desempeño de cada combinación de parámetros se muestra en la siguiente tabla.

Tabla 10: Ajuste de parámetros del modelo LinearSVM - CountVectorizer

Max_iter	C	Training time (s)	Accuracy
300	0.01	238.25	0.80
300	0.1	370.19	0.82
300	1	415.93	0.80
500	0.01	253.15	0.80
500	**0.1**	**416.39**	**0.82**
500	1	542.78	0.80
1000	0.01	242.92	0.80
1000	0.1	455.44	0.82
1000	1	845.67	0.80

En el caso del algoritmo ***SGDClassifier*** se ha partido del siguiente modelo base.

```
modelSGD = SGDClassifier(loss= 'perceptron',
                         penalty= 'l2',
                         max_iter= 1000,
                         n_jobs= -1,
                         class_weight= 'balanced',
                         learning_rate= 'optimal',
                         early_stopping= True,
                         random_state=42, verbose=2)
```

Al igual que los demás algoritmos, se ha probado el parámetro *class_weight* para observar el efecto en el *performance* del modelo, siendo el valor *balanced* el que tiene mejor desempeño. Ese valor se mantiene constante en el proceso de búsqueda de mejores parámetros.

Tabla 11: Comparación de parámetros para el algoritmo SGDClassifier- CountVectorizer

Class weight	Training time (s)	Accuracy
balanced	**144.02**	**0.75**
None	141.44	0.74

Los argumentos del algoritmo que se consideran en el proceso de ajuste de hiper-parámetros son los siguientes:

- *Max_iter:* Es el número máximo de epochs. El número de veces que se presenta los datos completos de *training.*
- *Loss*: Son las funciones de error que se emplean para el entrenamiento. *Modified_huber:* es una función de error que tiene tolerancia a valores atípicos, *squared_hinge* es similar al que se emplea en un *LinearSVM* solamente que en este caso la penalización de un error es cuadrática, *perceptrón* es un error lineal que es empleado por el algoritmo *perceptron.*

La configuración empleada para el ajuste de hiper-parámetros es la siguiente:

```
parametersSGD = {
```

```
'SGD__max_iter':[300, 500, 1000, 1500],
'SGD__loss':['modified_huber', 'squared_hinge', 'perceptron']
    }
```

Los parámetros que mejor desempeño obtienen son el valor 300 para el número máximo de iteraciones o *ephocs* y la función de error *modified_huber*. Estos valores alcanzan un *accuracy* de 76% en los datos de entrenamiento y un 78% para los datos de evaluación. Los resultados del desempeño de cada modelo se presentan a continuación.

Tabla 12: Ajuste de parámetros del modelo SGDClassifier - CountVectorizer

Max_iter	Loss	Training time (s)	Accuracy
300	**mod_huber**	**156.65**	**0.76**
500	mod_huber	147.66	0.76
1000	mod_huber	147.34	0.76
1500	mod_huber	142.43	0.76
300	squared_hinge	142.72	0.74
500	squared_hinge	150.28	0.74
1000	squared_hinge	138.26	0.74
1500	squared_hinge	141.67	0.74
300	perceptron	139.25	0.75
500	perceptron	142.13	0.75
1000	perceptron	141.37	0.75
1500	perceptron	140.24	0.75

Resumiendo, utilizando la técnica de *CountVectorizer*, los modelos que mejor desempeño alcanzan son aquellos que utilizan el algoritmo SVM lineal, en la siguiente tabla se puede observar la comparación con los demás modelos:

Tabla 13: Comparación de los modelos para CountVectorizer

Algoritmo de Clasificación	Precision - Weighted	Recall - Weighted	F1-Score-Weighted	Accuracy/ Micro Recall/ Precision/ F1-Score	Balanced Accuracy
MultinomialNB	0.77	0.77	0.76	0.77	0.64
LinearSVM	**0.84**	**0.84**	**0.84**	**0.84**	**0.73**
SGDClassifier	0.80	0.78	0.78	0.78	0.66
RandomForest	0.79	0.79	0.78	0.79	0.64

En la matriz de confusión del mejor modelo para countVectorizer, el algoritmo *LinearSVM* , se puede observar que la mayoría de las instancias que tienen más de 1,000 registros para la evaluación, clasifican correctamente la clase a la que pertenecen:

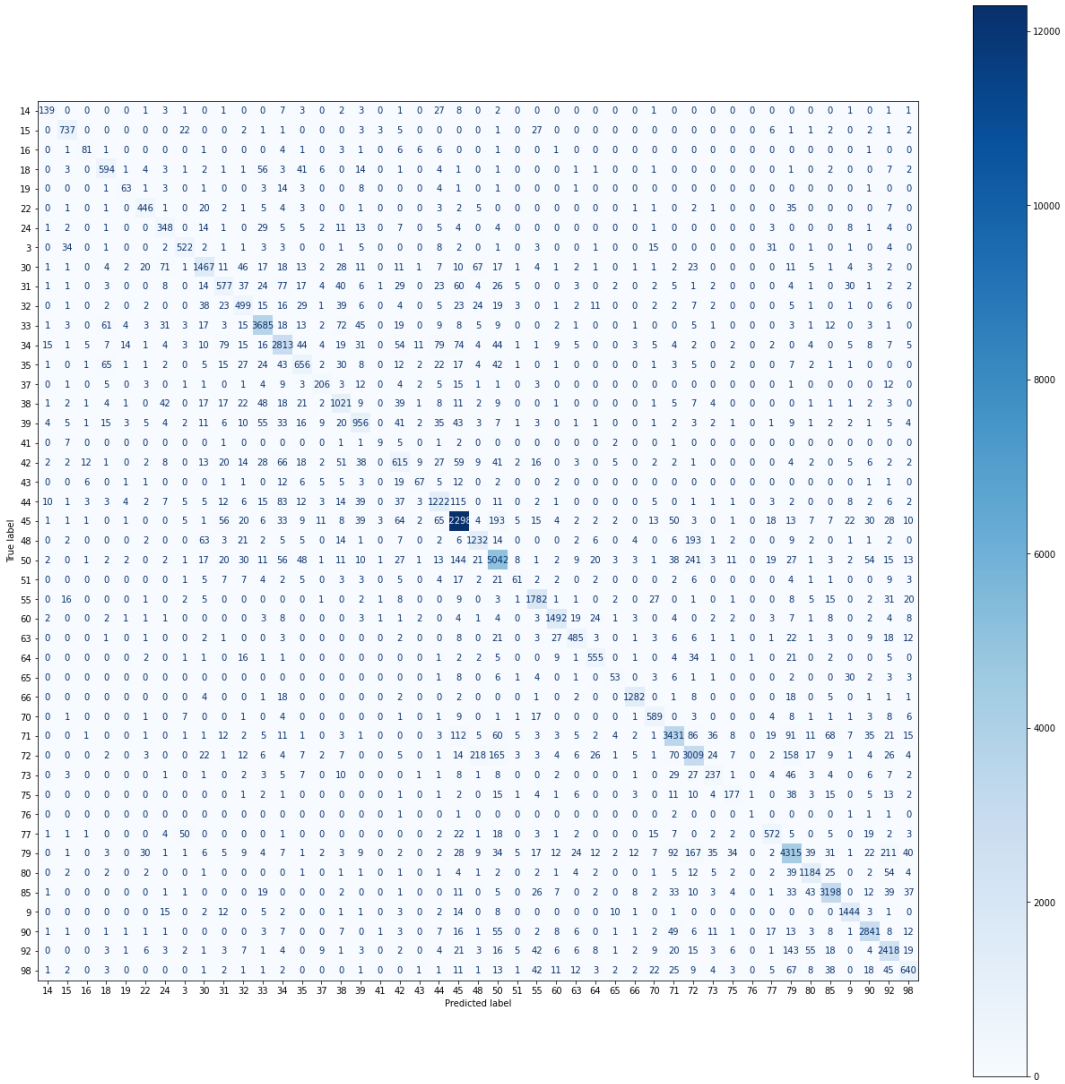

Ilustración 39: Matriz de confusión del algoritmo SVM - CountVectorizer

3.4.1.2 Técnica de vectorización TF-IDF

El código empleado para realizar la vectorización de las descripciones con la estrategia TF-IDF es el siguiente, al igual que en la técnica *CountVectorization,* se eliminan los acentos y se incluyen al vocabulario combinaciones de hasta tres palabras utilizando el argumento *ngram_range(1,3)* .

```
#Tf-Idf feature vectors
tfidf = TfidfVectorizer(ngram_range=(1,3),
                        analyzer="word",
                        tokenizer=word_tokenize,
                        strip_accents='ascii',
                        max_features=None,
                        lowercase=True)
```

El algoritmo ***MultinomialNB*** se crea con los parámetros por defecto y con el método *GridSearchCV* se realiza la búsqueda del mejor valor para el parámetro *alpha.* El mejor desempeño se presenta con el valor 0 y se obtiene un *accuracy* del 74% en promedio para el conjunto de *training* y un 77% para el *testing.*

Tabla 14: Ajuste de parámetros del modelo MultinomialNB – TF-IDF

alpha	Training time (s)	Accuracy
0	**14.75**	**0.74**
0.5	10.96	0.65
1	8.00	0.60

En el caso del algoritmo ***LinearSVM*** se utilizan los siguientes valores para la creación del modelo base.

```
modelSVM = LinearSVC(C=1.0,
                     penalty='l2',
                     class_weight='balanced',
                     random_state=42,
                     max_iter=1000,
                     verbose=3)
```

Antes de realizar la configuración para un ajuste de hiper-parámetros se ha optado por probar si el valor de *C* afecta al *performance* mientras se mantiene un mismo valor de *max_iter,* para esta prueba se ha establecido el valor de 300, la configuración de la prueba es la siguiente:

```
parametersSVM = {
                 'max_iter':[300],
                 'C':[0.01, 0.5, 0.9, 0.1, 1, 10, 100]
                 }
```

Los resultados son los que se muestran en la tabla que se presenta a continuación, el mejor *accuracy* en la etapa de entrenamiento es de 82% y en la evaluación de 84%, este desempeño corresponde al modelo con valor de *C=1.*

Tabla 15: Comparación de parámetros del modelo LinearSVM – TF-IDF

Max_iter	C	Training time (s)	Accuracy
300	0.01	101.10	0.70
300	0.1	96.75	0.78
300	0.5	139.84	0.81
300	0.9	193.10	0.81
300	**1**	**209.52**	**0.82**
300	10	783.240	0.81
300	100	1495.09	0.81

Por lo anterior, se mantiene constante el valor de C en la etapa de ajuste de hiper-parámetros. La configuración empleada para dicho propósito y haciendo uso del método *GridSearchCV* es la siguiente:

```
parametersSVM = {
                'max_iter':[100, 200, 300, 500, 700],
                'C':[1]
                }
```

El desempeño de cada una de las combinaciones de los parámetros se muestra en la siguiente tabla, siendo el valor 200 para el parámetro *max_iter* el cual alcanza el mejor *accuracy*, un 82% para los datos de entrenamiento y un 84% en la evaluación.

Tabla 16: Ajuste de parámetros del modelo LinearSVM-TF-IDF

Max_iter	C	Training time (s)	Accuracy
100	1	173.39	0.82
200	**1**	**182.04**	**0.82**
300	1	237.04	0.82
500	1	243.62	0.82
700	1	194.79	0.82

Para el algoritmo ***SGDClassifier,*** la creación del modelo base y la configuración empleada para el ajuste de hiper-parámetros se presenta a continuación.

```
modelSGD = SGDClassifier(loss='perceptron',
                         penalty='l2',
                         max_iter=1000,
                         n_jobs=-1,
                         class_weight='balanced',
                         early_stopping=True,
                         random_state=42,
                         verbose=2)
parametersSGD = {
                'max_iter':[200, 300, 500],
                'loss':['modified_huber', 'squared_hinge', 'perceptron']
                }
```

Los resultados muestran que la mejor combinación de parámetros para los argumentos *max_iter* y *loss* es 200 y *perceptrón* respectivamente, el modelo obtiene un *accuracy* de 76% en los datos de entrenamiento y un 80% para los datos de evaluación.

Tabla 17: Ajuste de parámetros del modelo SGDClassifier – TF-IDF

Max_iter	Loss	Training time (s)	Accuracy
200	mod_huber	57.49	0.76
300	mod_huber	58.48	0.76
500	mod_huber	56.63	0.76
200	squared_hinge	56.10	0.76
300	squared_hinge	56.60	0.76
500	squared_hinge	59.02	0.76
200	**perceptron**	**57.73**	**0.76**

300	perceptron	56.68	0.76
500	perceptron	54.79	0.76

Comparando el desempeño de cada algoritmo utilizando la técnica de vectorización TF-IDF, se obtiene que el algoritmo que mejor *accuracy* obtiene es el modelo lineal de SVM con un valor del 84% para los datos de evaluación. En la siguiente tabla se muestran los valores obtenidos.

Tabla 18: Desempeño de los algoritmos para tipo de vectorización TF-IDF

Algoritmo de Clasificación	**Precision - Weighted**	**Recall - Weighted**	**F1-Score- Weighted**	**Accuracy/ Micro Recall/ Precision/ F1-Score**	**Balanced Accuracy**
MultinomialNB	0.77	0.77	0.77	0.77	0.68
LinearSVM	**0.84**	**0.84**	**0.84**	**0.84**	**0.73**
SGDClassifier	0.80	0.80	0.80	0.80	0.69
RandomForest	0.81	0.81	0.80	0.81	0.65

3.4.1.3 Técnica de vectorización Word2Vec

Para generar el modelo Word2Vect se han utilizado la librería *gensim* con los siguientes parámetros.

```
w2v_model=gensim.models.Word2Vec(df['clean_text'], size=100, window=8, mi
n_count=1, sg=1, iter=20, seed=1852)
```

Este modelo aprende asociaciones de palabras a partir del corpus de las descripciones, una vez que se ha entrenado el modelo podemos encontrar palabras similares entre ellas, por ejemplo, si se busca las palabras similares a *software* el resultado es el siguiente:

```
[('licenciar', 0.8179511427879333),
 ('sofware', 0.7880822420120239),
 ('soportar', 0.7856779098510742),
 ('hardware', 0.7772219181060791),
 ('alienvault', 0.7438065409660339),
 ('sotfware', 0.7430359125137329),
 ('rational', 0.7394864559173584),
 ('liciencias', 0.7363829612731934),
 ('remedy', 0.7321535348892212),
 ('desktop', 0.7299513816833496)]
```

Ilustración 40: Palabras similares a Software utilizando Word2Vect

Con ese modelo se puede vectorizar cualquier palabra del corpus. Como se ha mencionado previamente, para vectorizar una sentencia entera se promedia los valores de los vectores de cada una de las palabras que conforman la descripción, el código empleado para este propósito es el siguiente.

```
def vectorize_sentence(sentence,model):
```

```
a = []
for i in sentence:
    try:
        a.append(model.get_vector(str(i)))
    except:
        pass
a=np.array(a).mean(axis=0)
a = np.zeros(100) if np.all(a!=a) else a
return a
```

El algoritmo ***MultinimialNB,*** al funcionar correctamente solo con datos discretos de entrada, tal como los son valores generados por las técnicas de *CounVectorizer* y *TF-IDF,* no se puede emplear con esta estrategia de codificación, ya que Word2Vect genera vectores de valores continuos, por lo que el algoritmo es sustituido por ***GaussianNB.*** En la siguiente tabla se muestran los resultados de los mejores modelos generados utilizando la técnica de vectorización de Word2Vect.

Tabla 19: Desempeño de los algoritmos para tipo de vectorización Word2Vec

Algoritmo de Clasificación	**Precision - Weighted**	**Recall - Weighted**	**F1-Score- Weighted**	**Accuracy/ Micro Recall/ Precision/ F1-Score**	**Balanced Accuracy**
GaussianNB	0.59	0.54	0.55	0.54	**0.48**
LinearSVM	**0.66**	**0.65**	**0.65**	**0.65**	0.43
SGDClassifier	0.60	0.49	0.50	0.49	0.39

3.4.1.4 Técnica de vectorización SentenceTransformer

Para la vectorización de cada una de las descripciones utilizando un *SentenceTransformer* para ello se descargó el modelo *PlanTL-GOB-ES/ roberta-base-bne.* Posteriormente se vectorizaron las descripciones y estos se ingresaron a los modelos de entrada para los algoritmos que se han utilizado previamente.

Los resultados de los mejores algoritmos después del ajuste de hiper parámetros se presentan a continuación.

Tabla 20: Desempeño de los algoritmos para tipo de vectorización SentenceTransformer

Algoritmo de Clasificación	**Precision - Weighted**	**Recall - Weighted**	**F1-Score- Weighted**	**Accuracy/ Micro Recall/ Precision/ F1-Score**	**Balanced Accuracy**
GaussianNB	0.56	0.47	0.50	0.47	**0.42**
LinearSVM	**0.72**	0.39	0.43	0.39	0.36
SGDClassifier	0.67	**0.54**	**0.54**	**0.54**	0.40

3.4.2 Modelado y evaluación de códigos CUCoP

En este apartado se muestran los modelos que se implementaron para los registros que corresponden a licitaciones de México. Debido al hecho de que, el nivel Capitulo cuenta solamente con 4 clases distintas y cada una de ellas con suficientes registros para entrenar un buen modelo de clasificación. Por ejemplo, el algoritmo *LinearSVM* obtiene un *accuracy* del 95% utilizando *CountVectorizer* y sin la necesidad de realizar ajustes de hiper-parámetros. Por lo anterior, se ha optado por entrenar modelos que permitan predecir un nivel inferior a capítulo, el nivel de concepto.

	precision	recall	f1-score	support
2000	0.96	0.96	0.96	54321
3000	0.96	0.95	0.95	53952
5000	0.64	0.74	0.69	3521
6000	0.94	0.95	0.94	13440
accuracy			0.95	125234
macro avg	0.87	0.90	0.89	125234
weighted avg	0.95	0.95	0.95	125234

Ilustración 41: Desempeño del modelo LinearSVM a nivel Capítulo

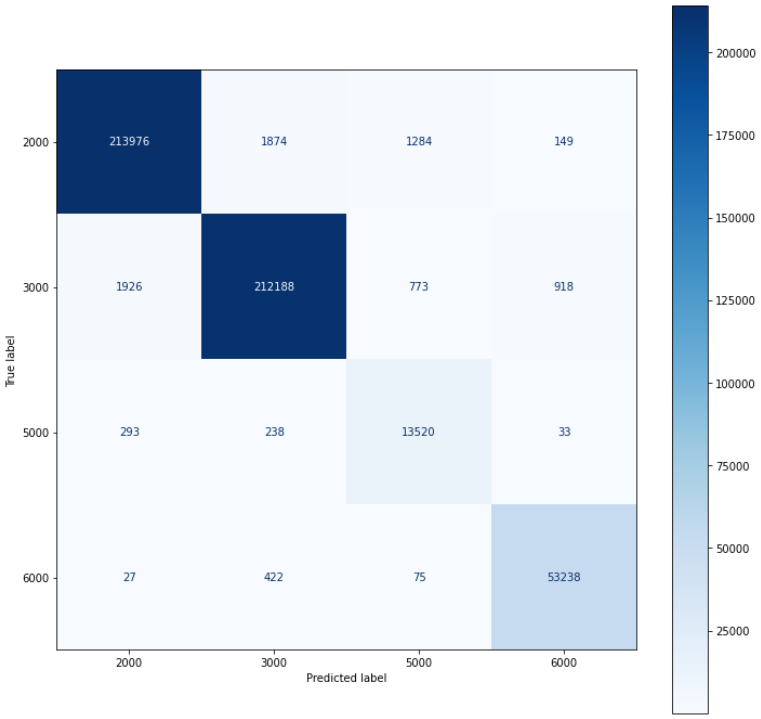

Ilustración 42: Matriz de confusión del algoritmo LinearSVM a nivel Capítulo

La cantidad de clases que el modelo será capaz de predecir corresponde con la cantidad de conceptos, es decir, 30 clases distintas. Al igual que se hizo con los códigos CPV, se eliminan los ceros de las claves para simplificar la representación de los datos. El conjunto total de registros de licitaciones es de 617,067. Estos registros se han dividido en 493,653 para la etapa de *training* y 123,414 para *testing*.

3.4.2.1 Técnica de vectorización CountVectorizer

En el caso del algoritmo ***MultinomialNB***, se ha empleado la siguiente configuración para el ajuste de los mejores parámetros. El mejor modelo se obtiene utilizando los valores *alpha*= 0 y *ngram_range= (1,3),* el *accuracy* alcanzado es del 86% en *training* y 88% en la evaluación.

```
parametersMNB = {
                 'vectorizer__ngram_range': [(1,1), (1,2), (1,3)],
                 'MNB__alpha':[0, 0.5, 1.0]
                 }
modelMNB = MultinomialNB()
```

Tabla 21: Ajuste de parámetros del modelo MultinomialNB - CountVectorizer

alpha	ngram_range	Training time (s)	Accuracy
0	(1,1)	128.92	0.83
0	(1,2)	140.48	0.85
0	**(1,3)**	**169.65**	**0.86**
0.5	(1,1)	117.17	0.82
0.5	(1,2)	140.75	0.84
0.5	(1,3)	170.97	0.85
1	(1,1)	119.15	0.80
1	(1,2)	146.04	0.82
1	(1,3)	158.14	0.83

En el caso del algoritmo ***LinearSVM,*** para la creación del modelo y proceso de ajuste de hiper-parámetros se ha empleado la configuración que se muestra a continuación. El modelo con la combinación de valores 300 y 0.1 para *max_iter* y *C* respectivamente es el que mejor desempeño obtiene con un *accuracy* del 89% en los datos de *testing.*

```
parametersSVM = {
                 'SVM__max_iter':[200, 300, 500],
                 'SVM__C':[0.1, 1]
                 }
modelSVM = LinearSVC(C=1.0, penalty='l2',
                  class_weight='balanced',
                  random_state=42,
                  max_iter=1000,
                  verbose=3)
```

Tabla 22: Ajuste de parámetros del modelo LinearSVM - CountVectorizer

max_iter	C	Training time (s)	Accuracy
200	0.1	532.05	0.89
200	1	640.93	0.88
300	**0.1**	**560.74**	**0.89**

300	1	760.37	0.88
500	0.1	735.39	0.89
500	1	913.53	0.88

Para el algoritmo ***SGDClassifier,*** se emplean las configuraciones siguientes:

```
parametersSGD = {
            'SGD__max_iter':[200, 300, 500],
            'SGD__loss':['modified_huber', 'squared_hinge', 'perceptron']
                  }
modelSGD = SGDClassifier(loss= 'perceptron',
                         penalty= 'l2',
                         max_iter= 1000,
                         n_jobs= -1,
                         class_weight= 'balanced',
                         learning_rate= 'optimal',
                         early_stopping= True,
                         random_state=42, verbose=2)
```

Tabla 23: Ajuste de parámetros del modelo SDGClassifier - CountVectorizer

Max_iter	Loss	Training time (s)	Accuracy
200	**mod_huber**	**201.55**	**0.81**
300	mod_huber	197.89	0.81
500	mod_huber	193.23	0.81
200	squared_hinge	189.98	0.74
300	squared_hinge	187.44	0.74
500	squared_hinge	185.30	0.74
200	perceptron	188.16	0.78
300	perceptron	182.50	0.78
500	perceptron	191.05	0.78

El mejor modelo utilizando la técnica *CountVectorizer* es el algoritmo *LinearSVM* con un *accuracy* de 91% en los datos de evaluación.

Tabla 24: Desempeño de los modelos utilizando CountVectorizer

Algoritmo de Clasificación	Precision - Weighted	Recall - Weighted	F1-Score- Weighted	Accuracy/ Micro Recall/ Precision/ F1-Score	Balanced Accuracy
MultinimialNB	0.88	0.88	0.88	0.88	0.71
LinearSVM	**0.91**	**0.91**	**0.91**	**0.91**	**0.77**
SGDClassifier	0.85	0.84	0.84	0.84	0.68

3.4.2.2 Técnica de vectorización TF-IDF

La configuración base que se ha empleado en la creación del modelo ***MultinomialNB*** se muestra en el siguiente código. Además, se realiza la búsqueda de los mejores parámetros y se obtiene que utilizando un *alpha* = 0 el modelo obtiene su mejor desempeño, ya que alcanza un 86% de *accuracy* en la etapa de entrenamiento y un 87% en la evaluación.

```
#Model
parametersMNB = {
                 'alpha':[0, 0.5, 1.0]
                }
modelMNB = MultinomialNB()
```

Tabla 25: Ajuste de parámetros del modelo MultinomialNB - TF-IDF

alpha	Training time (s)	Accuracy
0	**10.4**	**0.86**
0.5	9.8	0.81
1	7.3	0.78

Para el modelo ***LinearSVM,*** se emplea la configuración siguiente y solamente se realiza la búsqueda del mejor valor para el parámetro *max_iter*. El valor 100 es cuando se alcanza el mejor desempeño en la etapa de entrenamiento, tal como se muestra en la tabla de *performance* de este modelo.

```
parametersSVM = {
                 'max_iter':[100, 200, 300, 500, 700],
                 'C':[1]
                }
modelSVM = LinearSVC(C=1.0,
                     penalty='l2',
                     class_weight='balanced',
                     random_state=42,
                     max_iter=1000,
                     verbose=3)
```

Tabla 26: Ajuste de parámetros del modelo LinearSVM - TF-IDF

Max_iter	C	Training time (s)	Accuracy
100	**1**	**257.15**	**0.89**
200	1	277.56	0.89
300	1	289.41	0.89
500	1	286.01	0.89
700	1	260.37	0.89

En el caso del modelo ***SGDClassifier*** los resultados se presentan a continuación.

Tabla 27: Ajuste de parámetros del modelo SDGClassifier - TF-IDF

Max_iter	Loss	Training time (s)	Accuracy
100	**mod_huber**	**61.82**	**0.85**
200	mod_huber	47.47	0.85
300	mod_huber	51.26	0.85
500	mod_huber	47.56	0.85
100	squared_hinge	52.29	0.85
200	squared_hinge	52.19	0.85
300	squared_hinge	52.37	0.85
500	squared_hinge	53.96	0.85
100	perceptron	56.65	0.84
200	perceptron	55.79	0.84
300	perceptron	56.02	0.84
500	perceptron	42.17	0.84

En la siguiente tabla se muestran los mejores modelos obtenidos utilizando la técnica de vectorización TF-IDF. Tal como se observa, el mejor modelo es aquel que emplea el algoritmo lineal de SVM, cuyo *accuracy* es de 90%.

Tabla 28: Desempeño de los modelos utilizando TF-IDF

Algoritmo de Clasificación	Precision - Weighted	Recall - Weighted	F1-Score- Weighted	Accuracy/ Micro Recall/ Precision/ F1-Score	Balanced Accuracy
MultinimialNB	0.88	0.87	0.88	0.87	0.73
LinearSVM	**0.90**	**0.90**	**0.90**	**0.90**	**0.76**
SGDClassifier	0.87	0.85	0.86	0.85	0.76

3.4.2.3 Técnica de vectorización Word2Vec

Para generar el modelo Word2Vect se ha empleado el código siguiente.

```
w2v_model=gensim.models.Word2Vec(df['clean_text'], size=100, window=5, mi
n_count=2, sg=1, iter=15, seed=1852)
w2v_model.train(df['clean_text'], total_examples=w2v_model.corpus_count,
epochs=w2v_model.iter)
```

En la siguiente tabla se muestran los mejores modelos obtenidos después de realizar el proceso de ajuste de hiper-parámetros. El mejor desempeño pertenece al algoritmo SVM lineal con un *accuracy* del 71%.

Tabla 29: Desempeño de los modelos utilizando Word2Vect

Algoritmo de Clasificación	Precision - Weighted	Recall - Weighted	F1-Score- Weighted	Accuracy/ Micro Recall/ Precision/ F1-Score	Balanced Accuracy
GaussianNB	0.77	0.68	0.71	0.68	0.57
LinearSVM	**0.82**	**0.71**	**0.75**	**0.71**	**0.58**
SGDClassifier	0.80	0.65	0.70	0.65	0.53

3.4.2.4 Técnica de vectorización SentenceTransformer

Utilizando la técnica de vectorización con el modelo *SentenceTransformer,* los resultados muestran que el algoritmo con mejor desempeño es el GaussianNB, ya que obtiene un *accuracy* del 45%. En la siguiente tabla, se muestran los resultados de los mejores modelos después de utilizar el método *GridSearchCV* para cada uno de los algoritmos.

Tabla 30: Desempeño de los modelos utilizando SentenceTransformer

Algoritmo de Clasificación	Precision - Weighted	Recall - Weighted	F1-Score- Weighted	Accuracy/ Micro Recall/ Precision/ F1-Score	Balanced Accuracy
GaussianNB	0.73	**0.45**	**0.52**	**0.45**	**0.36**
LinearSVM	**0.81**	0.29	0.39	0.29	0.28
SGDClassifier	0.79	0.33	0.44	0.33	0.29

4 Resultados y conclusiones

En el presente trabajo se presenta un enfoque para clasificar códigos CPV y CUCoP a partir de las descripciones textuales del objeto del contrato de licitaciones públicas en idioma español. Estos estándares corresponden a los que se emplean en las contrataciones públicas de España y México respectivamente. Para realizar dicha clasificación, se han recopilado, de los principales portales de datos abiertos de contrataciones públicas de cada país, una gran cantidad de registros previamente clasificados. Además, se ha realizado un análisis exploratorio de los datos con la finalidad de identificar el nivel jerárquico ideal para la predicción de las clases. Posteriormente, se han aplicado las principales técnicas de preprocesamiento de lenguaje natural como: *Tokenization*, *Noise removal*, *Stop-word removal* y *lemmatization* para efectuar la limpieza de los registros y presentar a los modelos datos normalizados. Para realizar la representación vectorial de las descripciones textuales, se han empleado cuatro métodos distintos de vectorización, éstos son: *CountVectorizer, TF-IDF, Word2Vect* y *SentenceTransformer*. Finalmente, los vectores numéricos generados por los métodos de vectorización han servido como datos de entrada para entrenar tres modelos de clasificación supervisada: *MultinomialNB*, *Linear SVM* y *SGDClassifier*.

Los resultados muestran que, para el estándar CPV prediciendo clases a nivel división, el mejor modelo corresponde al algoritmo SVM lineal empleando TF-IDF como método de transformación vectorial. El desempeño obtenido, utilizando *accuracy* como métrica de evaluación, es de 84%. Un desempeño que supera aquellos modelos previos a este trabajo. En el caso del clasificador de artículos CUCoP, a nivel capítulo, el modelo SVM lineal utilizando *CountVectorizer* obtiene un *accuracy* del 95% y en cuanto a nivel concepto, el mejor modelo consiste en utilizar el algoritmo SVM lineal con la misma técnica de vectorización, dicho modelo obtiene un excelente *accuracy* de 91%. Cabe mencionar que, para ambos estándares, los modelos que mejor desempeño obtienen son aquellos en los que se emplean los métodos *CountVectorizer* o *TF-IDF*, resultados que no se esperaban al comienzo de este trabajo y pueden deberse al empleo de algoritmos de clasificación adecuados para este tipo de datos de entrada.

Por todo lo anterior, se puede concluir que los objetivos planteados al comienzo de este proyecto se han logrado perfectamente. Puesto que, se ha generado un modelo de clasificación para cada estándar de clasificador de artículos, cuyos desempeños son mejores a los ya existentes en el caso de CPV. Todos los datos, *scripts* para la recolección, exploración y preprocesamiento de los datos, así como los *notebooks* empleados para el entrenamiento y evaluación de los modelos se han puesto disponibles en un repositorio público con la finalidad de que cualquier parte interesada pueda reproducir o mejorar los resultados de este trabajo.

Para futuros trabajos, se recomienda a cualquier interesado utilizar los demás datos que se han extraído de las licitaciones públicas a fin de mejorar los desempeños de los modelos. Por ejemplo, el campo que almacena el órgano o la institución a cargo de la contratación pública, cuyo valor se menciona en varias de las descripciones textuales del objeto del contrato. Este dato al estar presente en las descripciones de los bienes o servicios solicitados podría no ser relevante en cuanto a la predicción del código del clasificador de artículos. De esta forma, las predicciones realizadas por los modelos no se encuentran sesgadas por la entidad pública que demanda un procedimiento de contratación pública.

Con respecto a líneas futuras de trabajo sobre las que se podría continuar este proyecto, podría ser la implementación de una API o aplicación web que permita a las partes involucradas en los procesos de contratación pública, especialmente a la encargada de la contratación, ingresar las descripciones de los objetos del contrato de los bienes o servicios que requieren para que dicho sistema sugiera el código CPV o CUCoP que mejor se adecue a la clasificación solicitada.

Entre las principales dificultades enfrentadas a lo largo del desarrollo de este proyecto, se encuentran los tiempos de espera bastante largos dedicados al procesamiento, entrenamiento y evaluación cruzada de los modelos, esto debido a la gran cantidad de registros recolectados. Asimismo, al emplear *Google Colaboratoy,* las desconexiones a los servidores eran recurrentes y se tenía que volver a empezar las ejecuciones desde cero.

Por otro lado, la experiencia de trabajar en un proyecto cuyo objetivo principal reside en aplicar cada una de las fases de la ciencia de datos es, en lo personal, la mayor satisfacción y logro conseguido de este trabajo. Asimismo, con este trabajo, se logra contribuir en el avance de la transparencia y eficiencia de las contrataciones públicas en España y México.

5 Bibliografía

[1] European Commission, «SMR Needs Analysis in Public Procurement,» 2022. [En línea]. Available: https://ec.europa.eu/docsroom/documents/46111. [Último acceso: 18 05 2022].

[2] J. Grandia, «Public Procurement in Europe,» de *The Palgrave Handbook of Public Administration and Management in Europe*, London, Palgrave Macmillan, 2018, pp. 363-380.

[3] Secretaría de la Función Pública, «Contrataciones Abiertas en el Gobierno de México,» 2022.

[4] OECD, «Public Procurement in the State of Mexico: Enhancing Efficiency and Competition,» OECD Publishing, Paris, 2021.

[5] OECD, «Contratación Pública en el Estado de México: Mejorando la Eficiencia y la Competencia,» OECD Publishing, Paris, 2021.

[6] OECD, «Estudio del Sistema Electrónico de Contratación Pública de México: Rediseñando CompraNet de manera incluyente,» OECD Publishing, Paris, 2018.

[7] Opentender Spain, «Opentender,» 2022. [En línea]. Available: https://opentender.eu/es/about/about-opentender. [Último acceso: 24 Mayo 2022].

[8] Boletín Oficial del Estado, «Agencia Estatal Boletín Oficial del Estado,» 09 2017. [En línea]. Available: https://www.boe.es/eli/es/l/2017/11/08/9/con. [Último acceso: 30 Mayo 2022].

[9] Ministerio de Política Territorial y Función Pública, «IV Plan de Gobierno Abierto España 2020-2024,» Secretaría General Técnica. Centro de Publicaciones, 2020.

[10] OECD, «Gobierno Abierto en América Latina,» OECD Publishing, Paris, 2015.

[11] Open Goverment Partnership, «Open Goverment Partnership,» [En línea]. Available: https://www.opengovpartnership.org/. [Último acceso: 25 Mayo 2022].

[12] OECD, «OCDE Recomendación del Consejo Sobre Contratación Pública,» OECD Publishing, París, 2015.

[13] Gobierno de Mexico, «Clasificador Único de las Contrataciones Públicas,» [En línea]. Available: https://sites.google.com/site/cnetcucop/. [Último acceso: 31 Mayo 2022].

[14] Oficina de Publicaciones de la Unión Europea, «SIMAP,» 2022. [En línea]. Available: https://simap.ted.europa.eu/web/simap/cpv. [Último acceso: 24 Mayo 2022].

[15] C. Shearer, «The CRISP-DM model: the new blueprint for data mining,» *Journal of data warehousing,* vol. 5, nº 4, pp. 13-22, 2000.

[16] OECD, «Contratación pública,» de *Panorama de las Administraciones Públicas América Latina y el Caribe 2020*, Paris, OECD Publishing, 2020, pp. 158-169.

[17] OECD, «Towards Agile ICT Procurement in the Slovak Republic: Good Practices and Recommendations,» OECD Publishing, Paris, 2022.

[18] Secretaría de la Función Pública, «Gobierno de México,» 09 Mayo 2017. [En línea]. Available: https://www.gob.mx/sfp/acciones-y-programas/1-3-1-licitacion-publica. [Último acceso: 2022 Mayo 30].

[19] Open Contracting Partnership, «Open Contracting Partnership,» [En línea]. Available: https://www.open-contracting.org/es/what-is-open-contracting/. [Último acceso: 2022 junio 30].

[20] OECD, «Preventing Corruption in Public Procurement,» OECD Publishing, Paris, 2016.

[21] Open Knowledge Foundation, «Open Data Handbook,» [En línea]. Available: http://opendatahandbook.org/guide/es/what-is-open-data/. [Último acceso: 25 Mayo 2022].

[22] Ministerio de Asuntos Económicos y Transformación Digital, «datos.gob.es,» 2009. [En línea]. Available: https://datos.gob.es/es/acerca-de-la-iniciativa-aporta. [Último acceso: 25 Mayo 2022].

[23] Gobierno de México, «Alianza para el Gobierno Abierto MX,» [En línea]. Available: https://gobabiertomx.org/. [Último acceso: 31 Mayo 2022].

[24] Comisión Intersecretarial para el Desarrollo del Gobierno Electrónico, «Datos Abiertos,» 15 Noviembre 2015. [En línea]. Available: https://www.gob.mx/cidge/acciones-y-programas/datos-abiertos. [Último acceso: 31 Mayo 2022].

[25] Gobierno de México, «datos.gob.mx,» [En línea]. Available: https://datos.gob.mx/. [Último acceso: 31 Mayo 2022].

[26] U.S Government, «Data.gov,» [En línea]. Available: https://data.gov/. [Último acceso: 2022 junio 30].

[27] The Government Digital Service, «data.gov.uk,» [En línea]. Available: https://data.gov.uk/. [Último acceso: 2022 junio 30].

[28] Ministerio de Hacienda y Función Pública, «Plataforma de Contratación del Sector Público,» [En línea]. Available: http://www.contrataciondelestado.es/. [Último acceso: 31 Mayo 2022].

[29] Ministerio de Hacienda y Función Pública, «Portal de datos abiertos del Ministerio de Hacienda,» [En línea]. Available: https://www.hacienda.gob.es/es-ES/GobiernoAbierto/Datos%20Abiertos/Paginas/licitaciones_plataforma_contratacion.aspx. [Último acceso: 31 Mayo 2022].

[30] Subdirección General de Coordinación de la Contratación, «Manual de uso de OpenPLACSP,» 2021.

[31] European Commission, «Licencia Pública de la Unión Europea (EUPL) Versión 1.2,» Diario Oficial de la Unión Europea, Bruselas, 2017.

[32] European Commission, «TED,» [En línea]. Available: https://ted.europa.eu/. [Último acceso: 2022 junio 30].

[33] European Commission, «data.europa.eu,» [En línea]. Available: https://data.europa.eu/. [Último acceso: 2022 junio 30].

[34] Gobierno de México, «CompraNet,» [En línea]. Available: https://compranet.hacienda.gob.mx/web/login.html. [Último acceso: 31 Mayo 2022].

[35] Gobierno de México, «Contrataciones Abiertas,» [En línea]. Available: https://www.gob.mx/contratacionesabiertas. [Último acceso: 30 Mayo 2022].

[36] Ministerio de Economía y Hacienda, «Proyecto CODICE: Componentes y Documentos Interoperables para la Contratación Electrónica,» Madrid, 2006.

[37] Ministerio de Hacienda y Administraciones Públicas, «Guía de implementación de documentos CODICE 2.0,» 14 09 2015. [En línea]. Available: https://contrataciondelestado.es/codice/2.0/doc/CODICE_2_GuiaImplementacion_v1.3.pdf. [Último acceso: 2022 junio 30].

[38] Open Contracting Partnership, «Estándar de Datos para las Contrataciones Abiertas,» [En línea]. Available: https://standard.open-contracting.org/latest/es/. [Último acceso: 2022 julio 01].

[39] CompraNet, «Clasificador Único de las Contrataciones Públicas - CUCoP,» Ciudad de México, 2021.

[40] Secretaría de Hacienda y Crédito Público, «Clasificador por Objeto del Gasto para la Administración Pública Federal,» Diario Oficial de la Federación, Ciudad de México, 2010.

[41] K. Görgün, «MKaan/multilingual-cpv-sector-classifier,» [En línea]. Available: https://huggingface.co/MKaan/multilingual-cpv-sector-classifier. [Último acceso: 2022 julio 01].

[42] J. Devlin, M.-W. Chang, K. Lee y K. Toutanova, «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,» *arXiv preprint,* nº arXiv:1810.04805, 2018.

[43] O. Corcho, D. Garijo y M. Navas-Loro, «Multi-label Text Classification for Public Procurement in Spanish,» *SEPLN,* 2022.

[44] A. Gutiérrez-Fandiño, J. Armengol-Estapé, M. Pàmies, J. Llop-Palao, J. Silveira-Ocampo, C. Pio Carrino, C. Armentano-Oller, C. Rodriguez-Penagos, A. Gonzalez-Agirre y M. Villegas, «MarIA: Spanish Language Models,» *arXiv preprint,* vol. arXiv:2107.07253, 2021.

[45] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer y V. Stoyanov, «RoBERTa: A Robustly Optimized BERT Pretraining Approach,» *arXiv preprint,* vol. arXiv:1907.11692, 2019.

[46] G. Gupta y S. Malhotra, «Text document tokenization for word frequency count using rapid miner (taking resume as an example),» *International Journal of Computer Applications,* vol. 0975, nº 8887, 2015.

[47] C. C. Aggarwal, Machine learning for text, Springer Cham, 2018.

[48] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes y D. Brown, «Text Classification Algorithms: A Survey,» *MDPI,* 2019.

[49] K. S. Jones, «A statistical interpretation of term specificity and its application in retrieval,» *Journal of documentation,* 1972.

[50] T. Mikolov, C. K. y J. Dean , «Efficient estimation of word representations in vector space,» *arXiv preprint arXiv:1301.3781,* 2013.

[51] N. Reimers y I. Gurevych, «Sentence-bert: Sentence embeddings using siamese bert-networks,» *arXiv preprint arXiv:1908.10084,* 2019.

[52] A. Charu C. y Z. ChengXiang, A survey of text classification algorithms, Boston: Springer, 2012.

[53] scikit-learn, «SGDClassifier,» [En línea]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html. [Último acceso: 2022 julio 01].