Universitat de les Illes Balears

# TRABAJO DE FIN DE MÁSTER

# ¿QUÉ CARACTERÍSTICAS TIENEN LAS PATENTES LITIGADAS POR LOS TROLLS?


## Antonio Linde Medina


**Máster Universitario Análisis de Datos Masivos en Economía y Empresa**

**(Especialidad/Itinerario *Herramientas en Gestión y Análisis Inteligente de Datos*)**

**Centro de Estudios de Postgrado**

**Año Académico 2020-21**

# ¿QUÉ CARACTERÍSTICAS TIENEN LAS PATENTES LITIGADAS POR LOS TROLLS?


## Antonio Linde Medina


## Trabajo de Fin de Máster


## Centro de Estudios de Postgrado

## Universidad de las Illes Balears


**Año Académico 2020-21**


Palabras clave del trabajo:

NPE, PAE, patent troll


*Nombre Tutor/Tutora del Trabajo Abel Ernesto Lucena Pimentel*

*Nombre Tutor/Tutora (si procede)*

*Nombre Tutor/Tutora (si procede)*

# ¿Que características tienen las patentes litigadas por los *Trolls*?

Antonio Linde Medina
**Tutor:** Abel Ernesto Lucena Pimentel
Trabajo fin de Máster Universitario. Análisis de Datos Masivos en Economía y Empresa (MADM)
Universitat de les Illes Balears
07122 Palma de Mallorca
antonio.linde1@estudiant.uib.cat

## Resumen

En el presente trabajo queremos averiguar las características de las patentes litigadas por los *Trolls*. Para ello utilizaremos los datos sobre litigios de la web *Unified Patents*[1]. Estos datos incluyen que tipo de organización es el demandante. Por lo tanto, tenemos los casos etiquetados según sea un *Troll* o no.

## Abstract

In this work we want to find the attributes of *Trolls*' patents. Data used is obtained from *Unified Patents* web. In this web cases are labeled by its entity type. This enable to classify cases as filed by *Trolls* or not *Trolls*.

**Palabras clave:** NPE, PAE, patent troll

## 1. Introducción

En el campo de las patentes, *entidad no practicante* o NPE (de sus siglas en inglés *non-practicing entity*) se refiere a una empresa o individuo que posee los derechos de una patente pero no tiene la capacidad para ponerla en práctica. Pueden ser inventores o empresas pequeñas. Existe gran cantidad de literatura sobre este tema. En Martín Sánchez (2020) tenemos una revisión actualizada de la misma.

Dentro del grupo de NPE están los llamados *Trolls* o PAE (*patent assertion entity*). Estos no buscan licenciar ni vender sus patentes sino que basan su modelo de negocio en litigar por ellas. Sus estrategias están orientadas a sacar el mayor provecho posible de los litigios (Pénin, 2012). Una de estas estrategias podría ser, por ejemplo, esperar a que la patente infringida sea difícil de sustituir. En este caso el *Troll* tendrá más fuerza en la negiciación y podrá sacar mayor provecho que si hubiese intentado licenciarla desde el inicio. La empresa infractora podría incluso haber hecho un desarrollo para evitar el uso de esa patente. Esta actividad está creciendo con rapidez y atrayendo a capital de riesgo (Henkel y Reitzig, 2008).

Hay sectores como el farmacéutico o el biotecnológico donde un producto depende de una única patente clave y ésta se protege con todos los medios disponibles. En otros sectores, como el de las comunicaciones móviles, un producto puede tener dependencia con miles de patentes. Muchas de estas patentes son de otras organizaciones (Henkel y Reitzig, 2008). Esto hace posible una práctica habitual de defensa utilizada por las empresas cuando son litigadas: Contraatacar denunciando a la empresa demandante por una infracción sobre una patente propia y así negociar un acuerdo. Pero dicha estrategia no es válida para defenderse de los *Trolls* ya que éstos no producen nada y por lo tanto no pueden infringir ninguna patente (Lemus y Temnyalov, 2017).

Un tema ampliamente discutido es el impacto de los *Trolls* en la innovación. Hay opiniones encontradas. Abrams, Akcigit, Oz, y Pearce (2019) elaboran un modelo teórico y experimental para averiguar si los *Trolls* son algo positivo o negativo para el desarrollo tecnológico. Aunque no llegan a una conclusión definitiva.

Cohen, Gurun, y Kominers (2016) también creen que los *Trolls* tienen un impacto negativo en las empresas afectadas. Concretamente en la innovación. Dichas empresas reducen su inversión en I+D después de llegar a un acuerdo o perder un litigio con un *Troll*. Los *Trolls* litigan independientemente de si se ha infringido o no una patente. Por ello sugieren que se hagan cambios en las normativas para pararlos lo antes posible en el proceso de litigio.

Además de reducir su inversión en I+D, la actividad de los *Trolls* hace que aumenten las pérdidas en las empresas atacadas. Por otra parte los pequeños inventores sacan una parte muy pequeña de lo que pierden las grandes empresas (Bessen, Ford, y Meurer, 2011).

Hay otros autores que consideran que la actividad de los *Trolls* es algo positivo. Según estos autores se trata de intermediarios que aportan valor entre la oferta y la demanda de patentes (Benassi y Di Minin, 2009).

Si nos fijamos en los litigios. La mayor parte de los defensores son empresas pequeñas. Por lo tanto, los *Trolls* no litigan únicamente con las grandes corporaciones (Bessen y Meurer,

[1] https://www.unifiedpatents.com/

2014). Hay que tener en cuenta que aproximadamente un tercio de los litigios contra las empresas pequeñas no llegan a los tribunales. Por lo tanto la información sacada de los litigios no será completa.

Debido al impacto cada vez mayor de los *Trolls* se han ido modificando las leyes para corregir algunas ineficiencias del sistema de patentes (Pénin (2012), (Martín Sánchez, 2020, p. 13)).

En cuanto a las métricas para medir la calidad de las patentes, las más utilizadas son (Lu, 2012; Reitzig, Henkel, y Schneider, 2010):

- *Forward citations*: Citas a la patente en otros documentos. Correlacionada con la calidad de la patente.
- *Backward citations*: Citas a patentes anteriores. Captura el tamaño del espacio tecnológico existente. Cuanto mayor es su valor más difícil es buscar una alternativa.
- *Number of clains*: Número de reivindicaciones incluidas en la patente.

En sectores como el de la electrónica o el software, donde la evaluación de las patentes es compleja, los *Trolls* consiguen registrar patentes de baja calidad. Son patentes que no presentan novedad o que son obvias y que afectan a productos fabricados por otras empresas. Éstas se dan cuenta de la infracción cuando llega la denuncia por parte del *Troll* (Davis, 2008). Por ello, es importante mejorar la calidad de la revisión de las patentes. Recientemente se están utilizando técnicas de Inteligencia Artificial para detectar solicitudes de patentes no válidas. *Unified Patents* es una de las entidades que está avanzando en esto. Incluso ofrecen premios por la elaboración de informes que demuestren la invalidez de una solicitud[2].

Leiponen y Delcamp (2019) no sustentan la opinión general de que los *Trolls* sistemáticamente trabajan con patentes de baja calidad, que mantienen las patentes ocultas durante un largo período de tiempo para que los fabricantes las utilicen en productos de alto coste o que intenten litigar rápidamente antes de que la patente sea invalidada por los tribunales. Además, creen que si las patentes fuesen de alta calidad y no tuviesen un gran número de reivindicaciones definidas vagamente se reducirían los litigios y su duración. Según Lu (2012) la calidad de las patentes litigadas es en general más alta que las no litigadas midiendo la calidad de éstas con las métricas anteriores.

Fischer y Henkel (2012) también están en contra de la opinión de que los *Trolls* trabajan con patentes de baja calidad. Consideran que éstos buscan patentes que tengan mayor probabilidad de ser infringidas, difíciles de sustituir y resistentes a cambios legales. Esto lo consiguen con patentes de amplio ámbito tecnológico y alta calidad. Cuanto mayor es el ámbito de la patente mayor probabilidad de que se infrinja. Una mayor densidad de la patente en el sector tecnológico hace que el coste de sustituirla sea mayor. La calidad de la patente mejora las posibilidades de defenderla en un tribunal.

En este trabajo vamos a analizar las características de las patentes litigadas por los *Trolls*. Para ello utilizaremos minería de datos con información obtenida de la web. Veremos que las patentes litigadas por los *Trolls* son de calidad. También realizaremos el análisis con información financiera obtenida de *Compustat*. En este caso observamos que la capacidad financiera de las empresas queda en segundo plano siendo más importantes las características de las patentes.

[2]https://patroll.unifiedpatents.com/contests

## 2. Motivación

Las patentes de los *Trolls* no son fáciles de identificar usando bases de datos públicas (Reitzig y cols., 2010). *Unified Patents* publica en su web un listado de todos los litigios de patentes en Estados Unidos desde 2010. Lo interesante para este trabajo es que los litigios están etiquetados con el tipo de entidad del demandante (ver **tabla 1**).

El objetivo principal de este trabajo es ver si con los datos de los litigios se pueden encontrar las características de las patentes litigadas por los *Trolls*.

Un segundo objetivo consiste en analizar contra que tipo de empresas litigan los *Trolls*. Para ello cruzaremos la información anterior con datos sobre variables financieras de las empresas demandadas obtenidos de *Compustat*. Esto es para ver si, como evidencian trabajos previos (Cohen, Gurun, y Kominers, 2019), los *Trolls* litigan contra empresas con capacidad financiera y, en principio, más dadas a llegar a un acuerdo y pagar antes que ir a juicio.

A continuación, describiremos el análisis de minería de datos realizado para conseguir estos objetivos.

## 3. Entorno de trabajo

Todo el código y los datos están almacenados en un repositorio de git[3]. Las herramientas utilizadas han sido:

- *Jupyter Notebooks* y *Pandas* para el análisis y procesamiento de los datos.
- Base de datos *MongoDB* para almacenar los datos de los litigios y las patentes.
- *Docker* para ejecutar todas las herramientas anteriores.

Todo lo necesario para reproducir los resultados de este documento está en el repositorio. En el mismo se incluyen las instrucciones de cómo hacerlo (ver **Apéndice A**).

## 4. Fuentes de datos

Las fuentes de datos utilizadas son las siguientes:

- La fuente principal son los datos de litigios recopilados en la *web* de *Unified Patents*[4]. Estos datos son accesibles a través de una API.

[3]https://bitbucket.org/antonio_linde_medina/madm-tfm

[4]https://portal.unifiedpatents.com/litigation/caselist

- Información financiera descargada de *Compustat* en formato CSV.

## 4.1. Descarga - API Rest

Los datos de los litigios se han obtenido mediante consultas a un *API Rest*. El resultado de la consulta es un documento *json* con los detalles de los casos. Estos documentos se han guardado en una base de datos *MongoDB*. De cada caso se han cogido todas las patentes afectadas y se han recuperado los detalles de la misma *web* y almacenado en la misma base de datos. En resumen, tenemos una base de datos *MongoDB* llamada `unified_patents` y en ésta la colección `cases` contiene los datos de los litigios y la colección `patents` los datos de las patentes.

El período de los litigios descargados va del 4 de enero de 2010 al 20 de agosto de 2020. Se han recuperado $42790$ casos y $38357$ patentes.

## 4.2. *Pipeline*

En la **figura 1** se puede ver el flujo de datos completo.

1. Con el contenido de las colecciones $cases$ y $patents$ de *MongoDB* se crean los *dataframes* $cases\_raw$ y $patents\_raw$ respectivamente. Estos *dataframes* tienen exactamente el contenido de las colecciones. No se ha realizado ningún procesado de los datos.
2. Se crea el dataframe *holdout*, siguiendo el procedimiento descrito por Geron (2017), con parte del contenido de *cases_raw*. Estos datos se guardan para validar el modelo al final.
3. Los dataframe $cases\_raw$ y $patents\_raw$ se procesan y se guardan como $cases$ y $patents$. En el **Apéndice B** se incluye una descripción de los campos.
4. Los datos de *Compustat* se leen de un fichero CSV.
5. Utilizando **tf-idf** y la **similitud coseno** se enlazan los datos de patentes con los de *Compustat* (ver **4.4.3**). En el **Apéndice B** hay una descripción de los campos.

## 4.3. Comprobación de los datos

Vamos a hacer algunas comprobaciones para ver que los datos descargados de la *web* sean correctos.

Lo interesante de la información sobre los casos es que la entidad que litiga está clasificada según los tipos de entidad que se muestran en la **tabla 1**. El objetivo de este trabajo es ver si se pueden obtener las características de las patentes que son litigadas por *Trolls*. Para ello enlazaremos los datos de patentes y de empresa con los casos. Así obtendremos una etiqueta $pae$ que indicará si la patente ha sido litigada por un *Troll* o no. En la **figura 2** vemos la distribución de los casos según el tipo de entidad y según la clase $pae$.

La **figura 3** muestra que el crecimiento en los litigios se ha debido principalmente a *Trolls* tal como indican Cohen y cols. (2016).

| **Tipos de entidad** |
|---|
| Operating Company |
| NPE (Patent Assertion Entity) |
| NPE (Individual) |
| NPE (Small Company) |
| University |
| Gov/NGO/Non-Profit |

Cuadro 1: Los demandantes están clasificados en uno de estos tipos de entidad.

Cohen y cols. (2019) muestran que los *Trolls* eligen el tribunal que creen será más favorable a sus intereses. En la fecha en la que se escribió el artículo uno de los tribunales que tenía más casos ($43\,\%$) era *Eastern District of Texas* (Cohen y cols., 2019, p. 28). En la **figura 4** vemos que se refleja este comportamiento.

En la **figura 5** se ve que los litigios de los *Trolls* se centran casi exclusivamente en patentes de tecnología punta. Los que aparecen como $unknown$ son valores nulos. Vemos que este atributo solo se registra desde 2015.

La figura **figura 6** muestra la distribución del atributo $cause\_of\_action$. La gran mayoría de casos relacionados con *Trolls* son de tipo *Infringement*. Lo cual era de esperar ya que la estrategia consiste en litigar cuando se infringe una patente.

## 4.4. Preparar los datos

La información de un litigio incluye una lista de las patentes afectadas. Con esta lista y el tipo de entidad, se etiquetan las patentes. Generamos así el *dataframe* $unified\_patents$. En la **tabla 6** del **Apéndice B** se describen los atributos del *dataframe* $patents$.

### 4.4.1. Fechas

Las patentes incluyen los siguientes atributos que son fechas:

- $priority\_date$
- $application\_date$
- $grant\_date$
- $publication\_date$
- $expiration\_date$

Todas estas fechas están correlacionadas (ver **tabla 2**) ya que corresponden a distintos momentos del proceso del registro de la patente. Además de estar correlacionadas entre sí, también lo están con el año. Algunos autores opinan que los *Trolls* litigan patentes que están a punto de expirar (Cohen y cols., 2019, p. 5470). Para ver si esto se puede corroborar, añadimos un nuevo atributo $days\_to\_expire = expiration\_date - filed\_date$. Este atributo representa los días que faltan para que expire la patente en el momento de registrar el ligitio. El resto de fechas las eliminaremos y utilizaremos solo este nuevo atributo. En la **figura 7** vemos que

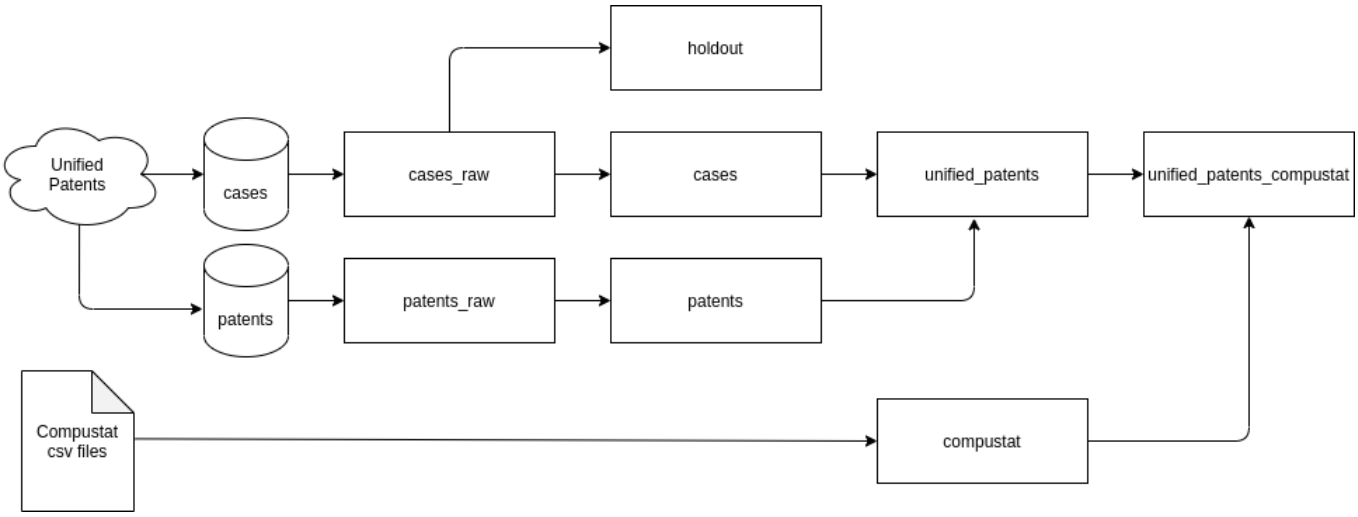



Figura 1: La principal fuente de datos es la web de *Unified Patents*. A través de su API obtenemos los datos de los casos y las patentes almacenándolas en una base de datos MongoDB[a]. Estos datos luego se procesan generando distintos *dataframes* de pandas[b]. La información financiera descargada de *Compustat* está almacenada en un fichero csv.

[a] `https://www.mongodb.com/es`

[b] `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html`

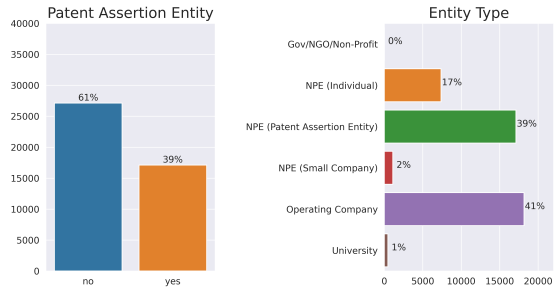



Figura 2: A la derecha muestra la distribución de los casos según el tipo de entidad. En la gráfica de la izquierda se incluye la distribución de los casos según el demandante sea un *Troll* o no.

la mayoría de valores para $days\_to\_expire$ está entre 3 y 12 años aproximadamente.

### 4.4.2. Listas

Algunos campos del *dataframe* $patents$ contienen listas. Por ejemplo el atributo $inventors$ es una lista de los inventores registrados en la patente. Se podría añadir una columna por inventor con datos binarios (1 si el inventor está incluido en la patente, 0 en caso contrario). Pero esto nos daría miles de columnas con casi todos los valores igual a 0. En lugar de hacer ésto, sustituimos dichos atributos por el número de elementos de la lista. Siguiendo con el ejemplo anterior, eliminamos $inventors$ y añadimos $inventors\_no$.

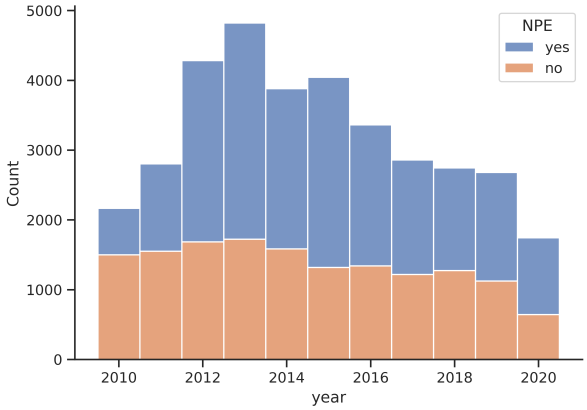



Figura 3: Las barras corresponden al número de casos por año. Cada barra está dividida en dos partes: Los casos en que la entidad demandante es NPE (*non-practicing entity*) y el resto. Comprobamos que el incremento en casos observado hasta 2015 es debido a los primeros.

| | year | days_to_expire | priority_date | publication_date | application_date | expiration_date |
|---|---|---|---|---|---|---|
| year | 1.00 | -0.10 | 0.38 | 0.44 | 0.42 | 0.42 |
| days_to_expire | -0.10 | 1.00 | 0.79 | 0.60 | 0.58 | 0.86 |
| priority_date | 0.38 | 0.79 | 1.00 | 0.72 | 0.76 | 0.91 |
| publication_date | 0.44 | 0.60 | 0.72 | 1.00 | 0.95 | 0.78 |
| application_date | 0.42 | 0.58 | 0.76 | 0.95 | 1.00 | 0.75 |
| expiration_date | 0.42 | 0.86 | 0.91 | 0.78 | 0.75 | 1.00 |

Cuadro 2: Correlación entre las distintas fechas.

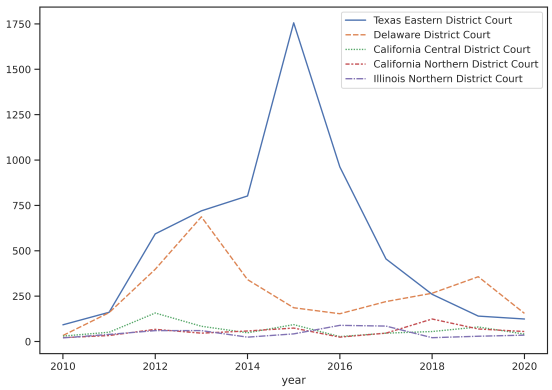



Figura 4: Número de litigios en que la entidad demandante es PAE (*patent assertion entity*) por tribunal y año. Los *Trolls* tienen preferencia por algunos tribunales. Sólo se muestran los 5 más frecuentes.

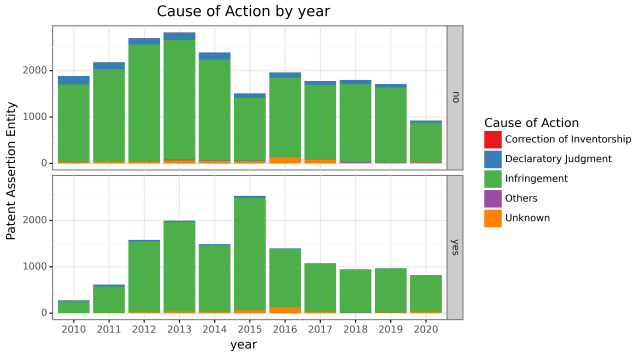



Figura 6: Distribución de la causa del litigio. Los casos relacionados con los *Trolls* son principalmente *Infringement*.


Figura 5: Distribución de los casos por sector industrial. Este campo solo está disponible a partir de 2015. Los *Trolls* tienen una gran actividad en el sector de tecnología punta.

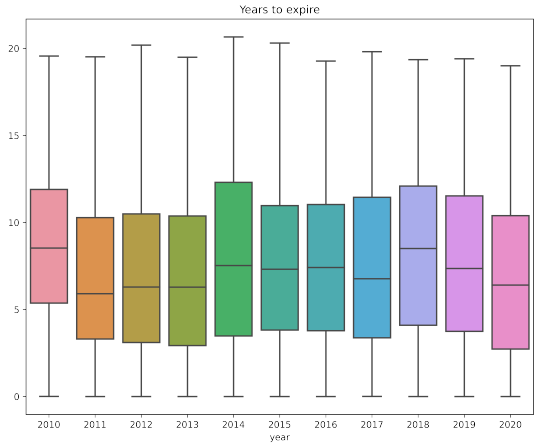



Figura 7: Relación entre $year$ y $days\_to\_expire$. $year$ es el año en que se registró el litigio. Aunque $days\_to\_expire$ toma valores extremos (de 0 a 20 años) vemos que la mayoría de valores se mantiene más o menos en el mismo intervalo a lo largo de los años.

#### 4.4.3. Similitud de textos

La base de datos *Compustat* contiene información financiera de empresas que cotizan en bolsa. El nombre de la empresa está almacenado en el campo $companyname$.

Estamos interesados en añadir la información financiera de las empresas litigadas a los datos de las patentes. En la base de datos de litigios, la empresa está en el campo $defendant$. El problema que nos encontramos al tratar de enlazar estas dos tablas es que los nombres de las empresas no están guardados exactamente igual.

Por lo tanto, para cada empresa litigada hay que buscar cual es la empresa más parecida en *Compustat*. Existen varias librerías en *Python*[5] para buscar textos similares. El problema es que son algo lentas para nuestro caso debido a la cantidad de datos. Hay que comparar cada registro de litigios con todos los de *Compustat*.

En una entrada del blog *van den Blog* se propone una solución a este problema[6]: Calcular la matriz `tf-idf` y utilizar la similitud coseno para encontrar las empresas más parecidas. En el cálculo de la matriz `tf-idf` los documentos son los nombres de las empresas y la mayoría son de una o dos palabras. Ésto hace que la comparación no vaya del todo bien. El procedimiento seguido es utilizar *n-grams* de tres caracteres como vocabulario.

El procedimiento sería:

- Calcular la matriz `tf-idf`
- Calcular la similitud coseno y obtener los valores por encima de un umbral.
- Quedarnos con la primera entrada. Es decir con la que tiene mayor coincidencia.

En una entrada posterior del mismo blog se presenta una clase en *Python* con varios métodos uno de los cuales hace exactamente lo que necesitamos[7]. El método en cuestión es $match\_most\_similar$. Nos devuelve una lista de nombres igual a la original en la que se han sustituido los nombres con una similitud superior a un umbral (por defecto $0{,}8$). En la **tabla 3** vemos el resultado obtenido. Se puede comprobar que no todas las empresas litigadas están en *Compustat*. Esto se debe a que no todas las empresas litigadas cotizan en bolsa.

Si añadimos una columna a la tabla de litigios con estos nombres, podemos utilizarla como clave para unirla con la tabla *Compustat*. En la **tabla 4** se incluye una muestra de los datos obtenidos de esta manera.

#### 4.4.4. Compustat

Además de lo explicado en la **sección 4.4.3**, para añadir los datos económicos es necesario tener en cuenta el año del litigio. Esto se debe a que la información en *Compustat* se registra por año. Por lo tanto, asociaremos los datos de *Compustat* correspondientes al año en que se litiga la patente.

#### 4.4.5. Datos balanceados

La distribución de la variable $pae$ no está balanceada en el *dataframe* $patents$[8].

| $pae$ | |
|---|---|
| Yes | $0{,}76\,\%$ |
| No | $0{,}24\,\%$ |

Los resultados obtenidos mejoran si balanceamos los datos. Como tenemos datos suficientes, cogemos una muestra aleatoria de la clase negativa del mismo tamaño que la positiva. Para ello utilizamos la función `RandomUnderSampler`[9]

#### 4.4.6. Otros cambios

Las variables categóricas con menos de 12 categorías se han sustituido por variables ficticias. Para las demás, se ha utilizado el código numérico de la categoría. Los valores nulos se han completado con la mediana.

## 5. Análisis

### 5.1. Árbol de decisión[10]

Nuestro objetivo es encontrar las características de las patentes litigadas por los *Trolls*. Como primera aproximación, entrenamos un árbol de decisión para clasificar las patentes y ver qué atributos son los que aparecen en las decisiones. De esta manera veremos qué variables son las que tienen mayor efecto. También tendremos un modelo base con el que comparar. Estos son los resultados de entrenar el árbol con validación cruzada y una profundidad de 15 nodos.

- Train Accuracy: $0{,}77$
- Test Accuracy: $0{,}77$
- F1 score: $0{,}63$

| | | Predicción | |
|---|---|---|---|
| | | No | Yes |
| Real | No | $75\,\%$ | $25\,\%$ |
| | Yes | $18\,\%$ | $82\,\%$ |

En la **figura 8** se muestran los primeros niveles del árbol. La primera decisión es sobre la variable $assignee\_changed$ que indica si la patente ha cambiado de propietario. Menos del $20\,\%$ de las patentes gestionadas por *Trolls* las han registrado

[5] spacy, nltk, fuzzymatcher

[6] *String Grouper*. https://bergvca.github.io/2020/01/02/string-grouper.html

[7] *Super Fast String Matching in Python*. https://bergvca.github.io/2017/10/14/super-fast-string-matching.html

[8] 20210523_random_forest.ipynb

[9] https://imbalanced-learn.org/stable/under_sampling.html

[10] 20210523_random_forest.ipynb

| input | output |
|---|---|
| SANDOZ | SANDOZ |
| Trulift Corporation | Trulift Corporation |
| Web.com Group | WEB.COM GROUP INC |
| Iso Beauty, Inc. | Iso Beauty, Inc. |
| Sally Beauty Holdings | SALLY BEAUTY HOLDINGS INC |
| Home Shopping Network, Inc. | Home Shopping Network, Inc. |
| Hair Tech, Inc. | Hair Tech, Inc. |
| Demand Industries, Inc. | Demand Industries, Inc. |
| Stitch N' Genius, Inc. | Stitch N' Genius, Inc. |
| Charles A. McClure | Charles A. McClure |

Cuadro 3: Resultado de utilizar el método $match\_most\_similar$ (ver texto) con los nombres de las empresas litigadas ($defendant$). Se busca una empresa similar en *Compustat* y si no se encuentra ninguna se deja el nombre como está. En caso contrario se sustituye.

| defendant | companyname | year | sic |
|---|---|---|---|
| SANDOZ | SANDOZ | 2010 | NaN |
| Trulift Corporation | Trulift Corporation | 2010 | NaN |
| Web.com Group | WEB.COM GROUP INC | 2010 | 7370.0 |
| Iso Beauty, Inc. | Iso Beauty, Inc. | 2010 | NaN |
| Sally Beauty Holdings | SALLY BEAUTY HOLDINGS INC | 2010 | 5990.0 |
| Home Shopping Network, Inc. | Home Shopping Network, Inc. | 2010 | NaN |
| Hair Tech, Inc. | Hair Tech, Inc. | 2010 | NaN |
| Demand Industries, Inc. | Demand Industries, Inc. | 2010 | NaN |
| Stitch N' Genius, Inc. | Stitch N' Genius, Inc. | 2010 | NaN |

Cuadro 4: Muestra de algunos datos del *dataframe* que contiene la información de litigios y de *Compustat*. $defendant$ es el nombre de la empresa tal como sale en el *dataframe* $unified\_patents$. $companyname$ es el campo que hemos añadido y utilizado como clave para unir los *dataframes* $unified\_patents$ y $compustat$ y generar $unified\_patents\_compustat$. Se corresponde a la columna $output$ de la **tabla 3**. El campo $sic$ es un dato de *Compustat*. Sólo tiene valor en los casos en que hay coincidencia entre la base de datos de litigios y *Compustat*.

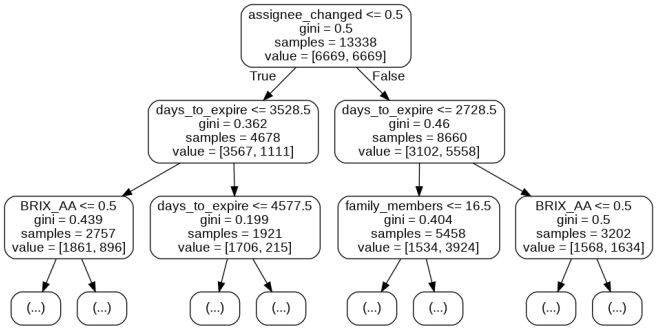



Figura 8: Árbol de decisión.

ellos. Por lo tanto los *Trolls*, en general, no desarrollan sus propias patentes sino que las adquieren de otras empresas o inventores. La siguiente decisión es sobre $days\_to\_expire$. Hay más casos positivos (*Trolls*) para valores menores de esta variable. Este punto lo comentaremos en el siguiente apartado.

## 5.2. Bosque aleatorio[11]

Vamos ahora a entrenar un bosque aleatorio. Con este modelo podremos ver que variables son las más importantes a la hora de clasificar las patentes. Utilizamos $GridSearchCV$ para hacer una búsqueda de hiperparámetros utilizando validación cruzada. El rango de parámetros utilizado es el siguiente:

- **n_estimators**: $[300, 350, 400, 600]$
- **min_samples_leaf**: $[3, 5]$
- **max_depth**: $[30, 40, 50]$

Con los rangos anteriores, la búsqueda da como mejores hiperparámetros los siguientes:

| parámetro | valor |
|---|---|
| max_depth | 40 |
| min_samples_leaf | 3 |
| n_estimators | 400 |

Utilizando estos hiperparámetros entrenamos un bosque aleatorio obteniendo el siguiente resultado:

- Train Accuracy: 0,82
- Test Accuracy: 0,81
- F1 score: 0,69

| | | Predicción | |
|---|---|---|---|
| | | No | Yes |
| Real | No | 80 % | 20 % |
| | Yes | 14 % | 86 % |

[11] 20210523_random_forest.ipynb

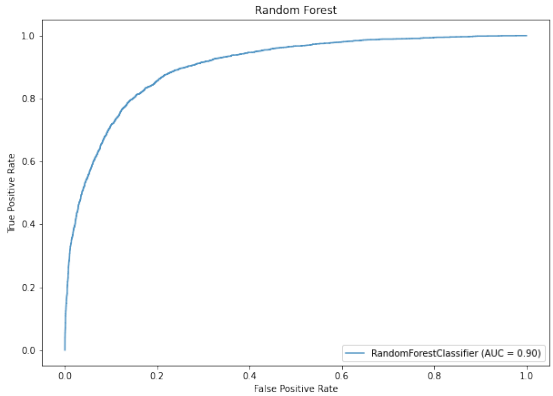



Figura 9: Curva ROC del bosque aleatorio.

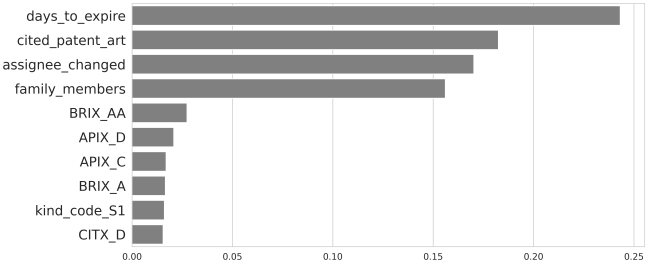



Figura 10: Importancia de las variables en el ajuste del bosque aleatorio. La variable más importante es $days\_to\_expire$. Esta variable indica los días que faltan para expirar la patente en el momento de registrar el litigio. Solo se muestran las 10 primeras.

En la **figura 9** se muestra la curva ROC del modelo. El resultado es algo mejor que en el apartado anterior usando un árbol de decisión. Sin embargo nuestro objetivo no es la clasificación en sí. Queremos ver qué características de las patentes influyen más en el resultado. En la **figura 10** vemos la importancia de los atributos. La variable más importante es $days\_to\_expire$. Las otras son:

- $cited\_patent\_art$: Número de referencias a la patente. Está relacionada con la calidad de ésta.
- $assignee\_changed$: $True$ cuando la cesión de la patente ha cambiado respecto del original.
- $family\_members$: Una familia de patentes es una colección de patentes que abarcan un contenido técnico similar[12]. Un valor más alto indica que el impacto en ese campo tecnológico es mayor.

En **4.4** comentamos que los *Trolls* litigan patentes a las que les falta poco para expirar como último intento de sacarles partido. El atributo $days\_to\_expire$ es uno de los más importantes en la clasificación. La **figura 12** muestra el resultado de utilizar *K-Means* con cuatro agrupaciones sobre el atributo $days\_to\_expire$. En la **figura 12a** vemos los grupos obtenidos y en la **figura 12b** la distribución de la clase dentro de cada grupo. No parece que los *Trolls* se dediquen especialmente a litigar patentes a punto de expirar.

La **figura 11** muestra la distribución del atributo $cited\_patent\_art$. Este atributo son las citas a la patente y está relacionado con la calidad de la misma. Vemos que las patentes litigadas por los *Trolls*, en general, tienen la misma calidad que las del resto de empresas. Los casos descritos en la literatura sobre la baja calidad de las patentes litigadas por los *Trolls* dependen de haber podido registrar dicha patente. Y ésto, con los cambios legislativos y una mejor revisión de las peticiones, cada vez es más difícil. Por lo tanto no es una estrategia sostenible.

## 5.3. Añadir información de *Compustat*[13]

Añadimos los datos económicos (ver **4.4.3**) al *dataframe* de las patentes y entrenamos un bosque aleatorio. Las empresas de la base de datos *Compustat* son empresas que cotizan en bolsa. En promedio, son más grandes y tienen mayor capacidad económica que las no cotizadas. Aquí nos interesa ver si hay alguna diferencia con respecto al resultado de **5.2**. En este caso el conjunto de las instancias incluye solo las que el defensor en el litigio está en *Compustat*. Por lo tanto, gran parte de los datos usados en **5.2** no están aquí. De las 30939 empresas que tenemos en la base de datos de litigios, solo 1840 están en Compustat.

Utilizamos $GridSearchCV$ para hacer una búsqueda de hiperparámetros utilizando validación cruzada. El rango de parámetros utilizado es el siguiente:

[12] https://www.epo.org/searching-for-patents/helpful-resources/first-time-here/patent-families.html

[13] 20201125_random_forest

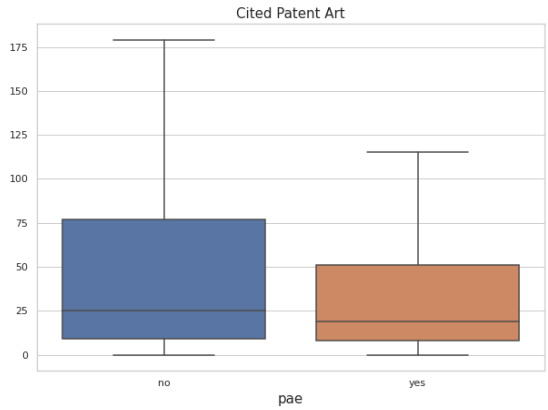



Figura 11: Distribución del atributo $cited\_patent\_art$ según el tipo de entidad. Este atributo está relacionado con la calidad de la patente.

- **n_estimators**: $[250, 300, 350, 400]$
- **min_samples_leaf**: $[3, 5]$
- **max_depth**: $[20, 30, 40, 50]$

Con los rangos anteriores, la búsqueda da como mejores hiperparámetros los siguientes:

| parámetro | valor |
|---|---|
| max_depth | 40 |
| min_samples_leaf | 3 |
| n_estimators | 300 |

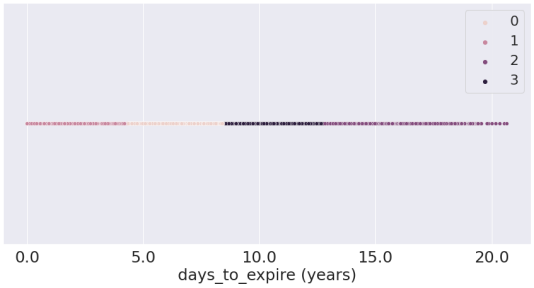
Con los hiperparámetros anteriores entrenamos un bosque aleatorio obteniendo el siguiente resultado:

- Train Accuracy: 0,83
- Test Accuracy: 0,84
- F1 score: 0,82

| | | Predicción | |
|---|---|---|---|
| | | No | Yes |
| Real | No | 83 % | 17 % |
| | Yes | 14 % | 86 % |

En la **figura 13** se muestran los atributos más importantes. El primero es $sic$[14]. Se trata de un código para clasificar las empresas según la indrustria. En la **figura 14** vemos que en algunas categorías sólo hay una clase ($pae$ o no $pae$). En otras categorías hay una gran diferencia en la proporción de cada clase. A este atributo le siguen en importancia los mismos de la sección anterior. Y a algo más de distancia las siguientes variables económicas

[14] *Standard Industrial Classification* (https://www.sec.gov/corpfin/division-of-corporation-finance-standard-industrial-classification-sic-code-list)




(a) 1a


(b) 1b

Figura 12: Resultado de utilizar K-Means. En la figura a) vemos que el conjunto de patentes queda segmentado según el tiempo que les queda para expirar en el momento del litigio. La figura b) muestra la distribución de la variable $pae$ en cada uno de estos segmentos.

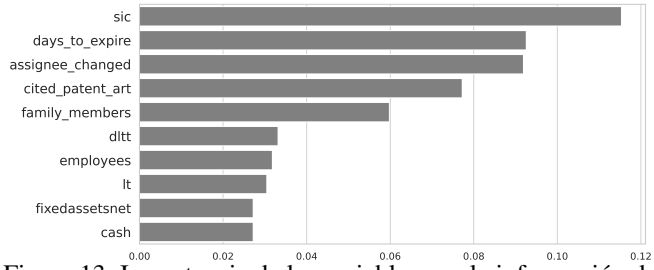



Figura 13: Importancia de las variables con la información de *Compustat*.

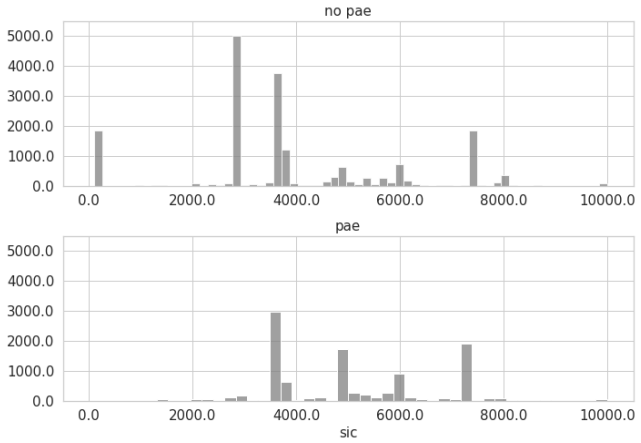



Figura 14: Distribución de las categorías empresariales. En la figura inferior se muestran los *Trolls* y en la superior el resto.

- $employees$: Número de empleados.
- $dltt$: Deuda a largo plazo (*Long-term Debt*).
- $ni$: Ingresos netos (*Net Income*).
- $fixedassetsnet$: Activos fijos (*Property, plan and equipement total (net)*)
- $wcap$: Capital circulante (*Working capital*).

Por lo tanto, las variables económicas quedan en segundo plano y las más importantes a la hora de clasificar las patentes litigadas por los *Trolls* son las mismas que en el apartado anterior. Esto nos dice que los *Trolls* buscan patentes con posibilidades de litigar independientemente de las empresas infractoras.

# 6. Conclusiones

A continuación presentamos las conclusiones de lo desarrollado en este trabajo:

1. En cuanto a la calidad de las patentes. Las patentes litigadas por los *Trolls* parecen ser de calidad alta. Esto está en línea con lo expuesto por Leiponen y Delcamp (2019) y en contra de la idea de que los *Trolls* trabajan sistemáticamente con patentes de baja calidad (Davis, 2008). Si la patente es de calidad tiene más opciones de ganar un litigio.
2. En cuanto a la capacidad financiera de las empresas. Las variables económicas quedan en segundo plano a la hora de clasificar las patentes litigadas por los *Trolls*. Los *Trolls* buscan patentes con posibilidades de litigar y robustas frente a cambios legislativos para que su modelo de negocio sea sostenible.
3. No parece que los *Trolls* se dediquen especialmente a litigar patentes a punto de expirar.

En este trabajo hemos usado datos sobre patentes litigadas. No todos los casos terminan en un juicio sino que en una parte de éstos se llega a acuerdos económicos. Y estos acuerdos no siempre se hacen públicos. Por lo tanto la información de la que disponemos solo cubre una parte de la actividad de los *Trolls*.

Infringir una patente sin saberlo es un riesgo para las empresas. Por ello es necesario que monitoricen el mercado de patentes. Una base de datos como la utilizada en este trabajo puede ser de ayuda ya que los *Trolls* están etiquetados y es posible obtener todas las patentes de los mismos, no solo las litigadas.

Los *Trolls* no suelen crear las patentes sino que las adquie-

ren. Normalmente de empresas en quiebra o de inventores que no tienen capacidad para ponerlas en práctica. Por ello, si se pudiese predecir qué patentes tienen más probabilidad de ser compradas por un *Troll* las empresas podrían considerar adelantarse y obtener las patentes que les afectan antes de que lo haga un *Troll*. Una opción que se podría probar es partir de la base de datos de patentes[15], etiquetar las que están en nuestra base de datos de litigios y utilizar aprendizaje semi-supervisado para etiquetar el resto.

# Apéndice A. Repositorio

Todo el código y los datos utilizados en este trabajo están un repositorio de Bitbucket[16].

Para ejecutar el código se necesita tener instalado `git` y `docker`. Estos son los pasos a seguir:

- Clonar el repositorio
  $ git clone <*repository url*>
- Arrancar los contenedores
  $ cd madm-tfm
  $ docker-compose up
- Para trabajar con *Jupyter* abrir la url que aparece al arrancar los contenedores[17].

Hay una documentación más detallada en el fichero README del repositorio.

[15] https://patentsview.org/apis/api-endpoints
[16] https://bitbucket.org/antonio_linde_medina/madm-tfm/src/master/
[17] Por ejemplo http://127.0.0.1:8888/?token=e8a761a226156ea0...

# Apéndice B. Descripción de los datos

| Campo | Descripción |
|---|---|
| case_id | Unique id |
| case_no | Case identification. Can be duplicated in different courts |
| cause_of_action | Cause of Action |
| closed_date | Closed Date |
| court | Court |
| defendant | List of defendants |
| entity_type | Entity Type. Operating Company, NPE, University, Gov |
| filed_date | The date the Patent Office acknowledges as the date you applied for a patent on your invention |
| industry | Industry. High-Tech, Medical, Other |
| judge | Judge's name |
| markman_hearing_date | Markman Hearing Date |
| patents | List of patents numbers |
| plaintiff | List of plaintiffs |
| product | Infringed product |
| status | Open, Closed |
| related_cases | List of related cases |

Cuadro 5: Campos de la tabla $cases$.

| Campo | Descripción |
|---|---|
| grant_number | Grant number (unique id) |
| application_number | Application number (unique id) |
| priority_date | Priority date |
| application_date | Application date |
| grant_date | Grant date |
| publication_date | Publication date |
| inventors | List of inventors |
| assignees | List of assignees |
| cited_patent_art | Cited Patent Art (number) |
| family_members | Family members (number) |
| CITX | Citation Rating |
| APIX | Validity Rating |
| BRIC | Broadness Rating |
| referenced_by | Number of references |
| expiration_date | Expiration date |
| assignee_changed | True if assignee is not the original |

Cuadro 6: Campos de la tabla $patents$.

| Campo | Descripción |
|---|---|
| gvkey | Compustat identifier |
| datadate | Date |
| fyear | Fiscal year |
| indfmt | Industry format |
| consol | Level of Consolidation |
| popsrc | Population source |
| datafmt | Data format |
| costat | Company Status (Active - Inactive) |
| companyname | Company legal name reported on its EDGAR SEC filings |
| at | Total assets leabilities of a company |
| cash | Cash |
| dltt | Long-term debt |
| dt | Total debt including current |
| ebit | Earnings Before Interest and Taxes |
| employees | Number of employees |
| revenue | Revenue total |
| xrd | Research and Development Expense |
| market_value | Market value |
| fixedassetsnet | Property, Plant and Equipment - Total (Net) |
| ni | Net Income |
| sic | Standard Industry Classification Code |
| wcap | Working Capital |

Cuadro 7: Campos de la tabla $compustat$.

# Referencias


Abrams, D. S., Akcigit, U., Oz, G., y Pearce, J. (2019). The Patent Troll: Benign Middleman or Stick-Up Artist? *University of Chicago, Becker Friedman Institute for Economics Working Paper No. 2019-51*. Descargado de `https://www.ssrn.com/abstract=3361215` doi: 10.2139/ssrn.3361215

Benassi, M., y Di Minin, A. (2009). Playing in between: Patent brokers in markets for technology. *R and D Management*, *39*(1), 68–86. doi: 10.1111/j.1467-9310.2008.00537.x

Bessen, J., Ford, J., y Meurer, M. J. (2011). The Private and Social Costs of Patent Trolls. *Regulation*, *34*(4), 26–35.

Bessen, J., y Meurer, M. J. (2014). The direct costs from NPE disputes. *Cornell Law Review*, *99*(2), 387–424. doi: 10.2139/ssrn.2091210

Cohen, L., Gurun, U. G., y Kominers, S. D. (2016). The growing problem of patent trolling. *Science*, *352*(6285), 521–522. doi: 10.1126/science.aad2686

Cohen, L., Gurun, U. G., y Kominers, S. D. (2019). Patent Trolls: Evidence from Targeted Firms. *Management Science*, *65*(12), 5461–5486. doi: 10.1287/mnsc.2018.3147

Davis, L. (2008). Licensing Strategies of the New 'Intellectual Property Vendors'. *California Management Review*, *50*(2), 6–30.

Fischer, T., y Henkel, J. (2012). Patent trolls on markets for technology – An empirical analysis of NPEs' patent acquisitions. *Research Policy*, *41*(9), 1519–1533. doi: 10.1016/J.RESPOL.2012.05.002

Geron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Henkel, J., y Reitzig, M. (2008). Patent sharks. *Harvard Business Review*, *86*(6).

Leiponen, A., y Delcamp, H. (2019). The anatomy of a troll? Patent licensing business models in the light of patent reassignment data. *Research Policy*, *48*(1), 298–311. doi: 10.1016/j.respol.2018.08.019

Lemus, J., y Temnyalov, E. (2017). Patent privateering, litigation, and R&D incentives. *RAND Journal of Economics*, *48*(4), 1004–1026. doi: 10.1111/1756-2171.12211

Lu, J. (2012). The myths and facts of patent troll and excessive payment: Have nonpracticing entities (NPEs) been overcompensated. *Business Economics*, *47*(4), 234–249. doi: 10.1057/be.2012.26

Martín Sánchez, M. (2020). *Three Essays on Patent Infringment and Litigation* (Tesis Doctoral no publicada). Universitat de les Illes Balears.

Pénin, J. (2012). Strategic uses of patents in markets for technology: A story of fabless firms, brokers and trolls. *Journal of Economic Behavior & Organization*, *84*(2), 633–641. doi: 10.1016/j.jebo.2012.09.007

Reitzig, M., Henkel, J., y Schneider, F. (2010). Collateral damage for R&D manufacturers: how patent sharks operate in markets for technology. *Industrial and Corporate Change*, *19*(3), 947–967. doi: 10.1093/icc/dtq037