

Empleatech: Recomendador Optimizado de Ofertas de Empleo

TRABAJO DE FIN DE MÁSTER DATA SCIENCE

ThePower - Hack(io)

Autor: Elena Cano Castillejo

Curso 2024-2025

Índice general

1	Resumen ejecutivo	1
1.1	Contexto y Problema Identificado	1
1.2	Solución Propuesta: Empleatech	2
2	Pipeline ETL	3
2.1	Extracción de las ofertas	3
2.1.1	Scrapeo de LinkedIn	3
2.1.2	Scrapeo de Infojobs	4
2.2	Transformación de los datos	4
2.2.1	Filtrado con OpenAI	5
2.2.2	Limpieza de los datos	6
2.3	Carga de los datos en MongoAtlas	6
3	Modelo	7
3.1	Análisis Exploratorio de Datos (EDA)	7
3.2	Recomendador	11
3.3	Web en Streamlit	12
4	Impacto de negocio	13
4.1	Beneficios y Oportunidad de Mercado	13
4.1.1	Modelos de Generación Ingresos	13
5	Conclusiones y próximos pasos	15
5.1	Conclusiones	15
5.2	Próximos pasos	16

CAPÍTULO 1

Resumen ejecutivo

Empleatech es una innovadora plataforma de búsqueda de empleo diseñada para optimizar y personalizar la experiencia de los candidatos en la búsqueda de oportunidades laborales. Actualmente, plataformas tradicionales como LinkedIn e InfoJobs presentan problemas significativos en la precisión de sus filtros de búsqueda, generando frustración y pérdida de tiempo en los usuarios.

Nuestro enfoque se basa en una solución disruptiva que, a través de inteligencia artificial y procesamiento de datos, analiza y recomienda ofertas de empleo que realmente se ajustan al perfil del usuario. En lugar de una búsqueda tradicional basada en palabras clave y filtros ineficientes, Empleatech recopila información sobre el nivel de inglés, años de experiencia y habilidades técnicas (hard skills) de cada usuario para ofrecerle únicamente aquellas oportunidades laborales que mejor se alinean con su perfil profesional.

Para ello, se ha desarrollado un proceso de recopilación de datos a través de técnicas de web scraping para extraer información de ofertas publicadas en LinkedIn e InfoJobs. Posteriormente, mediante inteligencia artificial generativa, se identifican y extraen las principales habilidades técnicas requeridas en cada oferta, permitiendo un filtrado preciso y eficiente. Finalmente, los datos son almacenados en una base de datos en la nube y presentados al usuario a través de una plataforma web intuitiva.

Este proyecto, en su fase inicial, se ha centrado en las búsquedas de empleo en tres áreas clave del sector tecnológico: Data Science, Data Engineering y Data Analytics acotadas a la Comunidad de Madrid. No obstante, existe un amplio potencial de expansión hacia otras disciplinas y geografías. Dado que actualmente no existe una solución en el mercado con este nivel de personalización y precisión, Empleatech representa una oportunidad única para transformar el sector de la búsqueda de empleo y mejorar significativamente la experiencia de los candidatos.

1.1. Contexto y Problema Identificado

La búsqueda de empleo es un proceso crítico tanto para los profesionales en búsqueda de nuevas oportunidades como para las empresas que requieren talento cualificado. Sin embargo, las plataformas tradicionales de empleo presentan importantes deficien-

cias en la precisión de sus filtros de búsqueda, lo que genera una experiencia ineficiente y frustrante para los usuarios. Entre los principales problemas se encuentran:

- **Filtros de búsqueda poco precisos:** A pesar de ajustar los criterios de búsqueda (como nivel de experiencia o idioma), las plataformas tradicionales continúan mostrando ofertas que no cumplen con los requisitos establecidos.
- **Resultados irrelevantes:** La búsqueda por palabras clave a menudo devuelve ofertas de empleo no relacionadas con la profesión deseada, obligando a los usuarios a revisar múltiples anuncios innecesarios.
- **Falta de visibilidad sobre habilidades requeridas:** Los usuarios deben ingresar manualmente a cada oferta para verificar si cumplen con las habilidades técnicas solicitadas, lo que prolonga significativamente el proceso de búsqueda.

1.2. Solución Propuesta: Empleatech

Empleatech surge como una alternativa disruptiva a los portales de empleo convencionales, ofreciendo una experiencia personalizada basada en el perfil del usuario. En lugar de depender de filtros de búsqueda genéricos, nuestra plataforma se centra en entender las capacidades y experiencia de cada candidato para mostrarle exclusivamente aquellas ofertas que realmente se ajustan a sus habilidades y expectativas.

El funcionamiento de Empleatech se basa en los siguientes pilares:

1. **Recopilación de Datos:** Se han extraído ofertas de empleo de plataformas como LinkedIn e InfoJobs utilizando tecnologías de web scraping (Scrapy, Selenium y BeautifulSoup). En el futuro, se planea ampliar la recopilación a otras plataformas como Indeed y Glassdoor.
2. **Procesamiento de Datos:** Una vez obtenidas las ofertas, se realiza un proceso de limpieza y estructuración de la información. Mediante inteligencia artificial generativa, se identifican las principales habilidades técnicas requeridas en cada oferta, así como nivel de inglés y experiencia laboral deseada.
3. **Almacenamiento y Consulta:** Los datos procesados son almacenados en una base de datos en la nube, garantizando su consistencia y accesibilidad.
4. **Interfaz de Usuario:** Se ha diseñado para que sea una web simple e intuitiva. Los candidatos ingresan a la plataforma y configuran su perfil con información clave como nivel de inglés, experiencia y habilidades técnicas. En base a estos parámetros, Empleatech recomienda automáticamente las ofertas que mejor se ajustan al perfil del usuario, proporcionando enlaces directos para aplicar a las vacantes. En un futuro se plantea la posibilidad de aplicar a dicho puesto directamente desde nuestra plataforma.

CAPÍTULO 2

Pipeline ETL

Para garantizar un procesamiento eficiente y estructurado de los datos en Emplea-tech, se ha desarrollado un pipeline ETL (Extract, Transform, Load) que permite la recolección, limpieza, transformación y almacenamiento de la información relevante para el sistema de recomendación de empleo. Este proceso es fundamental para asegurar la calidad y disponibilidad de los datos, optimizando su uso en los modelos de recomendación y facilitando una experiencia fluida para los usuarios.

A continuación, se detallan las etapas clave del pipeline. Dado que esta sección es muy técnica, ya que se va a explicar paso a paso como se ha desarrollado el proyecto, se recomienda acceder al repositorio de Github del proyecto e ir echándole un vistazo conforme se lee este capítulo. Acceda pinchando [aquí](#).

2.1. Extracción de las ofertas

En esta sección se detalla el proceso de extracción de las ofertas de trabajo de Infojobs y de LinkedIn para las posiciones de Data Science, Analyst y Engineer en la Comunidad de Madrid.

2.1.1. Scrapeo de LinkedIn

La obtención de las ofertas de LinkedIn se divide en dos pasos y todo el código se puede encontrar dentro de la carpeta *scraping_linkedin*.

Para llevar a cabo este paso se ha hecho uso de una nueva tecnología, scrapy, es un framework de scrapeo y crawling de código abierto, escrito en Python. Mediante scrapy se ataca un endpoint de una API de LinkedIn el cual nos devuelve una lista de ofertas y algunos detalles de estas como título, fecha de publicación, ubicación y el link a la oferta. Este link lo usaremos para poder acceder a la descripción completa de las ofertas que es nuestro objetivo, para poder extraer de estas descripciones detalles como los años de experiencia, nivel de inglés y skills requeridas.

La API no devuelve más de 250 ofertas de empleo, por lo que se ha realizado un script en el cual se lanzan spiders (archivos en los que se define cómo se rastreará

un determinado sitio) una tras otra hasta obtener alrededor de unas 1000 ofertas de empleo para cada uno de los tres empleos anteriormente mencionados.

Una vez hemos recopilado los datos básicos de todas las ofertas realizamos una primera limpieza superficial eliminando filas inválidas, cabeceras repetidas, ordenamos las ofertas de la más a la menos reciente y eliminamos aquellas de hace más de un año.

Continuamos accediendo al link de cada oferta mediante Beautiful Soup para extraer las descripciones. Hay veces que al hacer la petición falla, pero tras varios intentos eventualmente termina por devolver la respuesta esperada. Se ha establecido un bucle para llevar a cabo esta acción con un número máximo de 10 intentos, en la última prueba casi todas las repuestas se recibieron correctamente al primer intento. Los datos devueltos se van almacenando en la carpeta *data* dentro de *datos_descripcion_ofertas* en distintos archivos csv.

2.1.2. Scrapeo de Infojobs

Al igual que para LinkedIn, la obtención de estos datos presentó numerosas complicaciones puesto que tras haberlo intentado con diversas tecnologías como playwright, scrapy, selenium y uso de rotadores de proxy, generadores de cabeceras y solucionadores de captchas, por lo general se me acababa detectando como un bot y denegando la información.

Finalmente, se llevó a cabo la implementación con Selenium, logueándose con un usuario y contraseña e intentando replicar el comportamiento de un humano, moviéndose por las páginas con los botones en vez de ir cambiando las URL. Si solo se ejecuta una vez el archivo de extracción de las ofertas, es posible que no salga ningún captcha, sin embargo, hay veces que puede saltar y entonces debe ser resuelto a mano. Para los tres puestos planteados en este proyecto de Data Science, Analyst e Engineer en Infojobs a hay muchísimas menos ofertas que en LinkedIn, sin embargo, realizar este proceso a mano sería demasiado tedioso por lo que es mejor vigilar como lo hace Selenium durante unos minutos y estar atento por si salta un captcha, en vez de tener que hacer la recogida de la información manualmente. A pesar de ser un pequeño inconveniente, ahorra mucho tiempo. Una vez obtenidos los datos se hace una pequeña limpieza y son almacenados en *scraping_infojobs/datos_limpios*.

2.2. Transformación de los datos

Esta parte juega un papel crucial en el proyecto pues consiste en analizar las descripciones de las ofertas de trabajo y sacar los parámetros necesarios para posteriormente llevar a cabo las recomendaciones.

2.2.1. Filtrado con OpenAI

Una vez tenemos todas las ofertas tanto de LinkedIn como de Infojobs con su título, empresa, descripción y otros datos, el próximo paso consiste en utilizar un modelo de IA generativa, en este caso gpt-4o-mini, para extraer de las descripciones de las ofertas de trabajo los datos que vamos a necesitar para posteriormente hacer la recomendación, es decir, nivel de inglés deseado, años de experiencia requerida, skills necesarias o valorables para desempeñar dicho puesto.

La función principal que va a llevar la a cabo la extracción de los datos de las descripciones es *extraccion_datos_descripciones* y se puede encontrar dentro de la carpeta *src* en *funciones_transformacion.py*. Esta función itera por los csv obtenidos en la primera parte de extracción, para cada uno se recorren todas las filas, las cuales corresponden a una oferta y se accede al campo de la descripción de la oferta, el cual pasará como parámetro a la función *transformacion_openai*. Esta función se conecta a la API de OpenAI mediante una API key y basándose en la descripción extraerá los siguientes datos:

- Nivel de inglés requerido para el puesto si lo hubiera.
- Años de experiencia requeridos para el puesto si los hubiera.
- Porcentaje de cuanto se ajusta la descripción del empleo en cuanto al puesto al que se realizó la búsqueda, por ejemplo, Data Science.
- Skills necesarias que piden en la oferta para postularse al puesto de trabajo.
- Skills deseables que la empresa valora pero no son necesarias.

Cada vez que se recorre un csv los datos extraídos de cada oferta se van guardan en archivos json en la carpeta *transformacion_y_carga/data/response_openai*. Cabe mencionar que por cuestiones de gastos de la API y debido a que lo que se está realizando es una prueba de concepto, se han restringido los datos a las primeras 200 ofertas de cada empleo tanto para LinkedIn como para Infojobs.

Como cada csv corresponde a un tipo de empleo distinto: Data Science, Data Analyst o Data Engineer, para cada uno de ellos se ha generado un prompt distinto intentando así que las respuestas devueltas se ajusten en mayor medida a cada puesto. Se han implementado diversas versiones de los prompt, evaluando si al darle ejemplos más extensos y detallados mejoraba la respuesta, siempre teniendo en cuenta si era rentable, puesto que la plataforma cobra al hacer las llamadas tanto por tokens de entrada como por tokens de salida.

Por lo general, las skills extraídas de las ofertas están bien a pesar de que a veces se cuelan skills no deseadas como 'Grado en Ingeniería' o soft skills como 'Trabajo en equipo', las cuales habrá que filtrar en el siguiente paso.

Finalmente, recalcar que si en un futuro la aplicación se expandiera a otras geografías o puestos de trabajo habría que plantearse alternativas de modelos de IA más económicos como los de DeepSeek, el cual a pesar de ser muy reciente parece tener muy buenos resultados y ser mucho más económico.

2.2.2. Limpieza de los datos

Como se comentó en el aparatado anterior a la hora de extraer las skills de cada oferta obtenemos resultados no deseados que se deben filtrar. Por lo tanto, el primer paso consiste en una limpieza de las skills, no solo eliminar aquellas que no deseamos, sino también agrupar skills que se refieren a una misma habilidad pero se han recogido de distintas formas como PowerBi, Power BI, experto en PowerBi, todo ello lo pasaremos a un mismo término. Las ofertas con las skills ya filtradas se almacenan en dentro de la carpeta *limpieza_y_carga/data/skills_filtradas*.

Una vez finalizado el paso anterior si recorremos todas las skills de cada oferta y las vamos guardando en un lista veremos que tenemos alrededor de 2500 skills diferentes. Para implementar el recomendador la idea planteada es mostrar todas las skills de todas las ofertas, es decir, esta lista de 2500 skills y que el usuario haga check en aquellas que él posee y se le recomienden las ofertas que más se ajusten a las skills chekeadas. Sin embargo, es inviable hacerlo con tantísimas skills, por lo tanto, la mejor idea para reducir la lista y que el recomendador incluso funcione mejor para no ser tan extremadamente específico es la siguiente: Vamos a juntar tanto las ofertas de LinkedIn como de Infojobs y vamos a ver cuales son las skills que más se repiten para cada empleo, es decir, vamos a sacar el top de las skills para los puestos de Data Science, Data Analyst y Data Engineer. En este caso la función que implementa este proceso lo que hará será quedarse solamente con aquellas skills que tengan un mínimo de apariciones dentro de cada empleo, este mínimo se le pasa como parámetro a la función y por defecto es 10. Es decir, si dentro de todas las ofertas de Data Science la skill de Python aparece 23 veces, esa skill se mantiene, sin embargo, si HTML aparece 4 veces esa skill será eliminada. La función nos devuelve una lista únicamente con las skills relevantes que deben quedarse y con esta lista recorreremos todas las ofertas y eliminaremos de cada ofertas todas aquellas skills que no estén en la lista de skills relevantes. Estas ofertas filtradas se almacenan en *data/skills_filtradas_relevantes*.

De esta forma hemos conseguir homogeneizar las skills, manteniendo únicamente aquellas con mayores apariciones y consiguiendo reducir la lista de skills de todas las ofertas a unas 100. Este número es mucho más manejable para el usuario, pues revisar y chequear una lista de 100 skills solo lleva un par de minutos.

2.3. Carga de los datos en MongoAtlas

Finalmente, con todas las ofertas ya filtradas el último paso de la ETL es subir toda esta información a una base de datos. En este caso hemos elegido MongoAtlas, una base de datos no relacional la cual ofrece un servicio en la nube y es muy fácil de manejar a través de llamadas mediante la URI. Se ha creado una nueva base de datos llamada *empleatech_v2* con dos colecciones: linkedin e infojobs y dentro de cada una se han cargado los archivos obtenidos en el paso anterior. El código de carga se puede encontrar al final del notebook *transformacion_y_carga.ipynb*.

CAPÍTULO 3

Modelo

Antes de llevar a cabo la creación del modelo del recomendador primero vamos a realizar un estudio para conocer y analizar todos los datos que han sido recabados, una vez hayamos entendido los datos podremos generar el modelo y finalmente servirlo al usuario a través de una interfaz web.

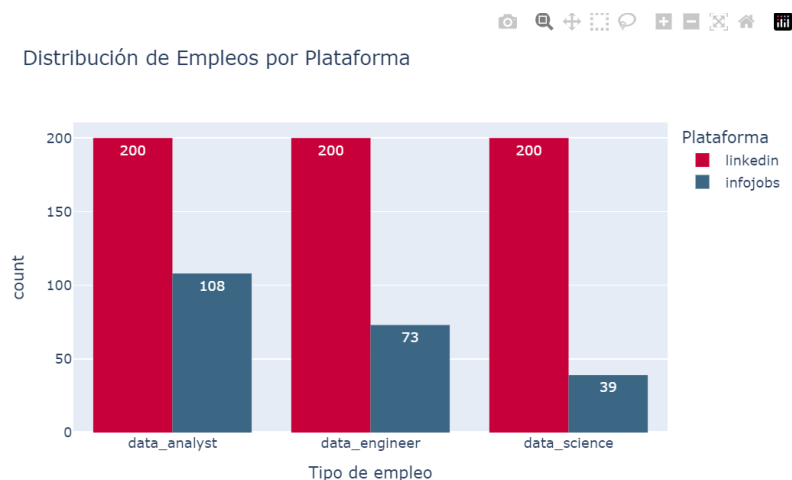
3.1. Análisis Exploratorio de Datos (EDA)

Tras llevar a cabo el proceso de ETL realizamos un EDA cuyo principal propósito es conocer y entender los datos de las ofertas que estamos manejando.

Comenzamos haciendo un recuento de las ofertas que hemos guardado en la base de datos obteniendo un total de **820**.

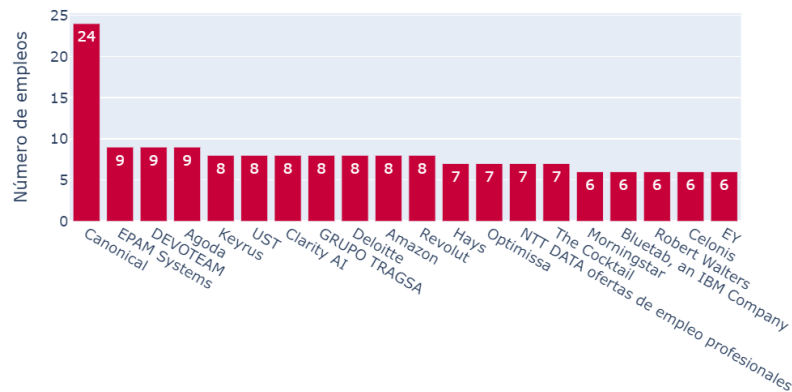
■ Tipos de empleo y plataformas

Las ofertas de trabajo vienen de dos plataformas distintas a día de hoy, LinkedIn e Infojobs. Los empleos ofertados son tres: Data Science, Data Analyst y Data Engineer. Observemos qué proporción hay de cada empleo por plataforma.



■ Empresas

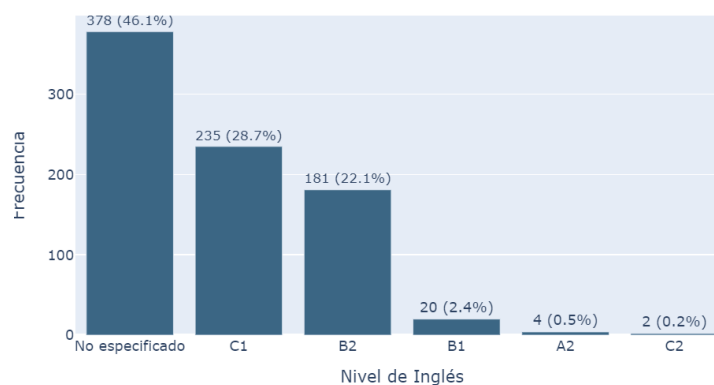
Top 20 empresas con más ofertas publicadas



Otro dato que puede ser de interés es, qué empresas están contratando más perfiles. Podemos ver el top 20 en el cual destaca en primera posición Canonical, una empresa de programación de ordenadores con base en Reino Unido. También podemos encontrar en este top empresas muy conocidas como Deloitte, Amazon, NTT, Bluetab, Celonis o EY, lo cual cobra bastante sentido pues son empresas que siempre están en busca de talento tecnológico con grandes departamentos en este sector.

■ Nivel de inglés

Distribución de los niveles de inglés requeridos



En cuanto al nivel de inglés un 46% de las ofertas no especifican qué nivel necesitan, sin embargo, en el 54% restante sí que exigen un nivel de inglés. En el 29% de los casos piden un C1 equivalente a un nivel avanzado, seguido de cerca por el B2 con un 22% lo cual equivale a un nivel medio.

■ Años de experiencia

Observando la imagen 3.1 vemos que en el 30% de las ofertas no se especifica la experiencia del usuario, en el 38% de los casos se piden entre 2 y 3 años de

Cantidad y porcentaje de ofertas por años de experiencia



Figura 3.1: Ofertas por años de experiencia

experiencia. Un 10 % piden como mínimo un año de experiencia y tan solo un 0.2 % no piden experiencia, siendo esta una cifra alarmante para los jóvenes de nuestro país, los cuales al acabar la carrera no encuentran oportunidades en el mercado laboral.

Distribución de salarios por años de experiencia

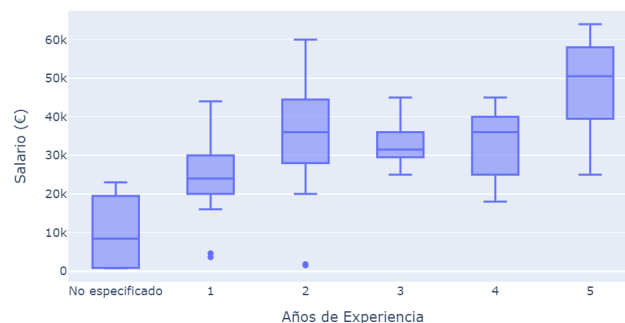


Figura 3.2: Salario según años de experiencia

Viendo la imagen 3.2 del salario por años de experiencia, vemos que los datos para 3 y 4 años parece no encajar demasiado, pues de forma lógica no tiene sentido que el sueldo máximo de una persona con 2 años de experiencia sea superior al de una persona con 3 o 4 años. Sin embargo, esto se puede deber a que no hay muchos empleos que pidan este rango de años de experiencia, por lo que habría que aumentar considerablemente la base de datos para sacar unos insights más realistas. Por otro lado, si que parece tener sentido los salarios para 1, 2 y 5 años de experiencia, puesto que hay más ofertas para estos años. Como next steps se plantea la extracción de un número mayor de ofertas de trabajo y ver cómo se modifica esta gráfica.

■ Skills requeridas en las ofertas

Skills más requeridas en ofertas de Data

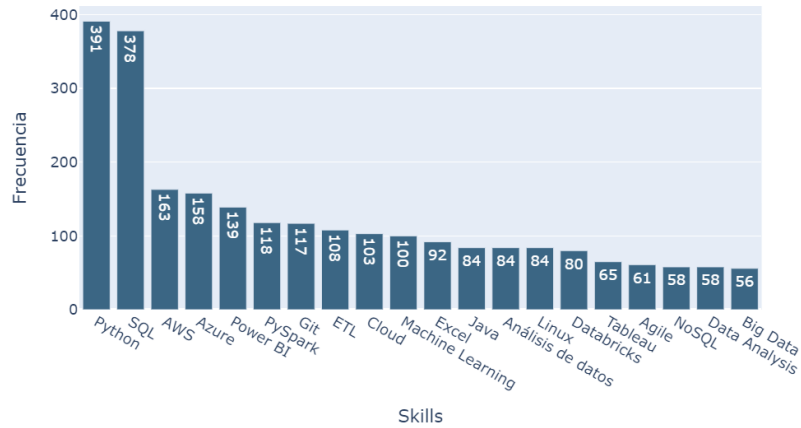


Figura 3.3: Skills en ofertas de Data

Para todos los puestos de data en la imagen 3.3 podemos ver cuales son las skills más solicitadas, liderando Python y SQL, seguidas de tecnologías cloud y otras tecnologías propias de los distintos puestos de data science, analyst y engineer. Por lo que en la siguiente gráfica 3.4 podremos ver de forma más limpia cuales son las skills que más se demandan en cada una de las posiciones.

Top 10 Skills más demandadas en cada empleo

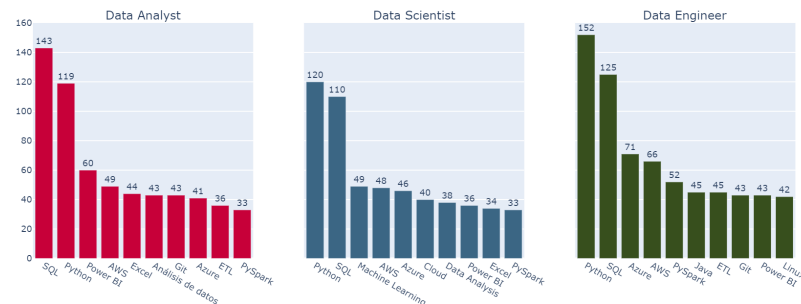


Figura 3.4: Skills por tipo de empleo

En todas ellas tenemos de nuevo en cabeza SQL y Python, ya que son muy características de la rama Data y se utilizan para los tres tipos de empleo. Para data analyst como era de esperar en tercera posición tenemos Power BI, también está Excel y aptitudes de analista de datos. Para data science en tercera posición tenemos Machine Learning y seguido de tecnologías Cloud. En cuanto a data engineer tenemos cloud seguido de pyspark, java y ETL. Todas ellas encajan con tecnologías propias de cada puesto y pueden ayudarnos a ver qué es lo que más se demanda en las ofertas de trabajo y así formarnos para alcanzar un perfil más empleable.

3.2. Recomendador

Tras llevar a cabo todo el proceso de ETL y con los datos ya tratados y almacenados en la base de datos podemos proceder a ver el recomendador, el cual es el eje de todo el proyecto. Su funcionamiento como hemos explicado antes es muy sencillo. El usuario podrá ajustar una serie de filtros: el nivel de inglés que tiene, los años de experiencia que quiere que pidan en las ofertas, el tipo de oferta que quiere en cuanto a Data Science, Data Analyst, Data Engineer o todos los puestos y también le aparecerán la lista de todas las hard skills que hay en las ofertas y podrá chequear cuales cumple o quiere que tengan las ofertas recomendadas. Una vez configurados estos filtros se obtendrá una lista de ofertas de la que más encaja con el perfil del usuario a la que menos.

El recomendador se ha desarrollado en dos fases. La primera fase consiste en recomendar ofertas solamente en base a la lista de skills marcadas por el usuario. Para ello nos traemos de la base de datos todas las ofertas, las cargamos en un dataframe y creamos una nueva columna donde combinamos en una cadena de texto las skills necesarias y valoradas para cada oferta. A esta nueva columna le añadimos un nuevo registro el cual son las skills que han sido chequeadas por el usuario, por lo que nos queda una columna donde cada fila tiene las skills que aparecen en cada oferta y en la última fila las skills que el usuario ha marcado para recomendarle ofertas en base a ellas. Esta nueva columna es vectorizada usando la clase *CountVectorizer* de la librería de *sklearn*, convirtiendo el texto en una matriz de recuentos de términos (bag of words). Es decir, transforma cada cadena de texto en una representación numérica basada en la frecuencia de las palabras, devolviendo una matriz de tokens. Con esta matriz podemos calcular la similitud del coseno entre las filas. La similitud del coseno mide el grado de similitud entre dos vectores, es una medida muy utilizada en procesamiento de lenguaje natural (NLP), machine learning y sistemas de recomendación. En este caso nos interesa la distancia de la fila correspondiente a las skills marcadas por el usuario hacia el resto de filas, que son las skills de cada oferta, aquellas que tengan una similitud de coseno más alta serán las ofertas que más se ajusten a nivel de habilidades técnicas al perfil del usuario. Por lo que creamos una lista donde vamos añadiendo las ofertas y su puntuación desde la que mayor puntuación tiene hasta la que menos, obteniendo así la primera parte del recomendador basado en las skills.

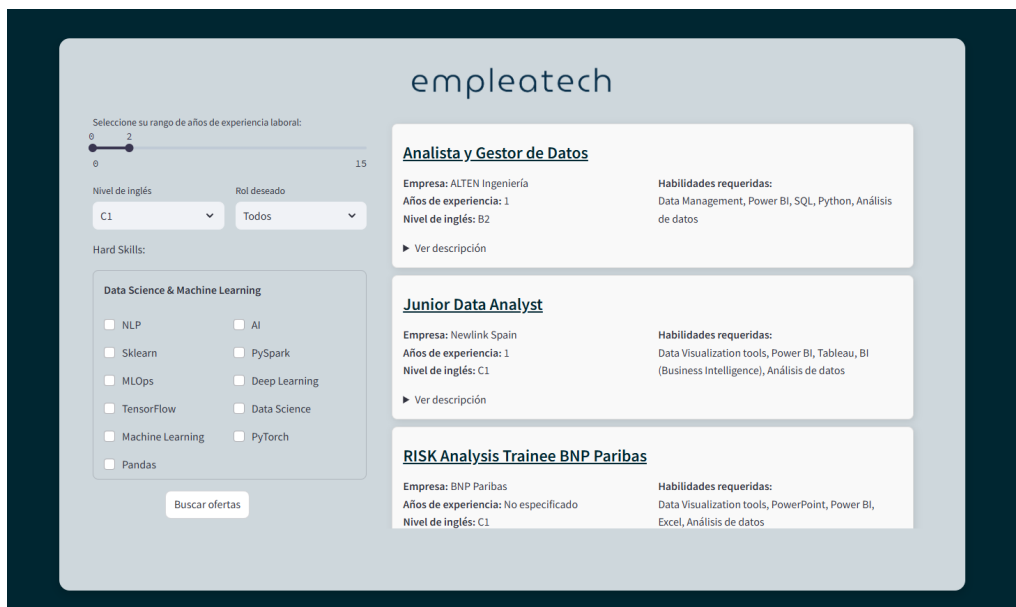
La segunda parte consiste en ajustar el resto de filtros. Una vez hemos obtenido la lista de ofertas de la parte anterior ahora vamos a comprobar si las ofertas de esta lista cumplen con el nivel de inglés introducido por el usuario, los años de experiencia seleccionados y si la oferta corresponde a un puesto de Data Science, Analyst, Engineer o cualquiera de ellos. Una vez hemos comprobado qué ofertas cumplen con estos parámetros revisamos si hubiera alguna oferta repetida, pues una misma oferta puede haber sido publicada tanto en LinkedIn como en Infojobs y haberla recogido dos veces. Finalmente, tras llevar a cabo todas estas comprobaciones tenemos la lista final de ofertas para recomendar al usuario en base a los parámetros de los filtros.

La métrica utilizada para evaluar el rendimiento del recomendador consiste en ver para cada oferta recomendada cuantas de las skills de la oferta cumple el usuario. Por ejemplo, el usuario marca las siguientes skills: Python, SQL, PowerBi, Flask y Java,

en la primera oferta que que le da el recomendador piden Python, SQL, NoSQL y Flask, por lo que le daríamos una puntuación de 3/4 porque el usuario cumple tres de las cuatro skills que pide la oferta. De esta forma hemos comprobado que efectivamente el recomendador muestra en posiciones más altas aquellas cuyo porcentaje en la evaluación es más alto, comprobando así el correcto rendimiento del modelo.

3.3. Web en Streamlit

Una vez implementado el recomendador necesitamos una interfaz donde el usuario pueda interaccionar con los filtros y visualizar las ofertas, para ellos hemos desarrollado una página web sencilla e intuitiva con streamlit. Esta elección se debe a que streamlit es un framework de código abierto en Python diseñado para crear aplicaciones web de manera sencilla y rápida. Su principal ventaja es su facilidad de uso, ya que permite transformar scripts de Python en aplicaciones web sin necesidad de conocimientos avanzados en desarrollo web. La página web está desplegada y operativa en el siguiente enlace: <https://empleatech.streamlit.app/>.



En la parte de la izquierda podemos ver los filtros que son ajustados por el usuario y una vez configurados se da al botón de *Buscar ofertas*. Es entonces cuando en la parte de la derecha aparecerán las ofertas recomendadas, siendo la primera la que más se ajusta al perfil del usuario y conforma vamos bajando el grado de similitud es menor. Para cada oferta podemos ver unos rasgos generales como el nombre del puesto y de la empresa, los años de experiencia que piden, el nivel de inglés y si pinchamos en el nombre de la oferta nos abre una nueva pestaña en LinkedIn o Infojobs para ver los datos completos y poder solicitar el puesto vacante.

CAPÍTULO 4

Impacto de negocio

4.1. Beneficios y Oportunidad de Mercado

Empleatech se posiciona como una solución innovadora en un mercado altamente competitivo, aportando las siguientes ventajas clave:

- **Ahorro de tiempo:** Reducción significativa del tiempo dedicado a la búsqueda de empleo al eliminar ofertas irrelevantes y mostrar solo aquellas alineadas con el perfil del usuario.
- **Experiencia personalizada:** Adaptación dinámica de las recomendaciones en función de las habilidades y preferencias del usuario.
- **Escalabilidad:** Potencial de expansión a múltiples sectores y geografías, integrando nuevas fuentes de datos y funciones avanzadas de personalización.

Actualmente, no existe en el mercado una plataforma de empleo con este nivel de personalización y eficiencia. Esto representa una oportunidad única para Empleatech de posicionarse como un referente en el sector y revolucionar la manera en que los profesionales encuentran oportunidades laborales.

4.1.1. Modelos de Generación Ingresos

Para garantizar la sostenibilidad y escalabilidad de Empleatech, se plantean varios modelos de monetización que permitirían generar ingresos tanto de empresas como de candidatos. Estos modelos están diseñados para ofrecer valor agregado a cada actor dentro del ecosistema de búsqueda de empleo, optimizando la conexión entre talento y oportunidades laborales. A continuación, se presentan algunas ideas para generar fuentes de ingresos que impulsarán el crecimiento de la plataforma.

1. **Planes de Suscripción para Empresas:** Pago de una cuota por parte de las empresas por acceder a un panel de candidatos altamente cualificados y filtrados de forma eficiente.

2. **Publicación Destacada de Ofertas:** Pagos por parte de las empresas para que sus ofertas aparezcan en posiciones prioritarias dentro de la plataforma.
3. **Acceso Premium para Candidatos:** Funcionalidades avanzadas como recomendaciones personalizadas de formación, notificaciones en tiempo real o análisis de mercado laboral basado en su perfil.
4. **Integración con Plataformas de Formación:** Promoción de cursos y certificaciones en habilidades demandadas con comisiones por cada inscripción generada.
5. **Publicidad Selectiva:** Empresas de software, herramientas de productividad o formación pueden anunciarse en la plataforma, con segmentación basada en el perfil de los usuarios.

CAPÍTULO 5

Conclusiones y próximos pasos

Empleatech ha demostrado ser una solución innovadora y efectiva para abordar los desafíos actuales en la búsqueda de empleo, especialmente en el sector tecnológico. A través de los conocimientos adquiridos de Data Science y un enfoque basado en inteligencia artificial, hemos logrado crear una plataforma que no solo optimiza la experiencia del usuario, sino que también mejora la precisión de las recomendaciones de ofertas laborales. A continuación, se presentan las conclusiones clave y los próximos pasos para el desarrollo.

5.1. Conclusiones

Tras probar la herramienta podemos ver como cumple correctamente con su cometido, filtra correctamente los años de experiencia, el nivel de inglés y las skills. Las ofertas devueltas se ajustan perfectamente a los filtros introducidos cumpliendo con el objetivo propuesto en este proyecto.

- **Eficiencia en la Búsqueda de Empleo:** Empleatech reduce significativamente el tiempo que los candidatos dedican a buscar ofertas de empleo relevantes. Al filtrar y personalizar las recomendaciones basadas en habilidades y experiencia, la plataforma permite a los usuarios concentrarse en las oportunidades que realmente se alinean con su perfil.
- **Personalización a través de IA:** La implementación de modelos de inteligencia artificial, como el uso de OpenAI para extraer habilidades y requisitos de las descripciones de trabajo, ha demostrado ser efectiva. Esto no solo mejora la calidad de las recomendaciones, sino que también permite una adaptación continua a las necesidades cambiantes del mercado laboral.
- **Base de Datos Sólida:** La creación de un pipeline ETL eficiente ha permitido la recopilación, limpieza y almacenamiento de datos de manera estructurada. Esto garantiza que la información disponible en la plataforma sea precisa y esté actualizada. Además es un proceso replicable y escalable a la hora de una futura expansión hacia nuevos empleos.

- **Modelo de Negocio Viable:** La identificación de múltiples fuentes de ingresos, como suscripciones para empresas y acceso premium para candidatos, proporciona una base sólida para la sostenibilidad financiera de la plataforma. Esto permitirá a Empleatech crecer y expandirse en el futuro.

5.2. Próximos pasos

- **Expansión de la Base de Datos:** Se planea ampliar la recopilación de datos a otras plataformas de empleo, como Indeed y Glassdoor, para enriquecer la base de datos y ofrecer una mayor variedad de ofertas laborales. Esto también incluirá la exploración de oportunidades en otros sectores más allá del tecnológico.
- **Mejora Continua:** Se plantea la mejora de algunos puntos críticos:
 - El scrapeo de las ofertas es un proceso complicado y tedioso por lo que una alternativa sería obtener los datos a través de APIs de pago.
 - Automatización del pipeline de la ETL para que todos los días a una determinada hora se ejecute y así tener la base de datos siempre actualizada.
 - También hay que tener en cuenta que el modelo de IA utilizado para la extracción de las skills a partir de las descripciones de las ofertas es de pago y habría que valorar otras opciones mas rentables como modelos de Claude o DeepSeek.
 - Mejora de la página web, dado que el principal propósito del proyecto era la implementación de la ETL y el Modelo, el diseño web está en un segundo plano, sin embargo, se plantea un desarrollo de backend y frontend más profesional y consumible para los usuarios.
- **Desarrollo de Funcionalidades Adicionales:** Se explorará la implementación de nuevas características:
 - Posibilidad de aplicar directamente a las ofertas desde la plataforma.
 - Implementar un inicio de sesión para que los usuarios se pudieran registrar, guarden las características de su perfil, puedan guardar ofertas que les interesen e incluso se pueda mandarles notificaciones si aparecen ofertas nuevas ajustadas a su perfil.
 - También sería muy interesante ver para cada oferta qué aptitudes le faltan al perfil del usuario y recomendarle cursos online de plataformas como Coursera para poder adquirir esas skills y ser más empleable.
- **Pruebas de Usuario y Feedback:** Se llevarán a cabo pruebas de usuario para obtener retroalimentación sobre la experiencia de la plataforma. Esto ayudará a identificar áreas de mejora y a garantizar que la interfaz sea intuitiva y fácil de usar.

Empleatech representa un avance significativo en la forma en que los profesionales buscan empleo y cómo las empresas encuentran talento. A medida que continuamos desarrollando y refinando la plataforma, estamos comprometidos a mantener un enfoque centrado en el usuario, garantizando que nuestras soluciones no solo sean innovadoras, sino también efectivas y accesibles. Con un mercado laboral en constante evolución, Empleatech está bien posicionada para liderar el cambio y mejorar la experiencia de búsqueda de empleo.