

INSTITUTO TECNOLÓGICO DE BUENOS AIRES – ITBA
ESCUELA DE GESTIÓN Y TECNOLOGÍA

Automatización de geolocalización de direcciones para OCASA

AUTORES:

Collado, Camila (Leg. N° 61487)

Lara Acosta, Agustin (Leg. N° 60788)

Pettinato, Camila (Leg. N° 61050)

DOCENTE/S TITULAR/ES O TUTOR/ES O DIRECTOR:

Rodríguez Varela, Juan Pablo

Pascale, Ignacio José

PROYECTO FINAL PRESENTADO PARA LA OBTENCIÓN DEL TÍTULO DE
LICENCIADO/A Analítica Empresarial y Social

BUENOS AIRES
SECUNDO CUATRIMESTRE, 2023

Índice

Objetivo del proyecto	4
Medición de valor y KPIs a Impactar	5
Key Performance Indicators (KPIs) del proyecto	5
Key Performance Indicators (KPIs) de la empresa	5
Entregables y Outputs del proyecto	6
Investigación de metodologías	7
Natural Language Processing (NLP)	7
Named Entity Recognition (NER)	7
Spancat	7
Fuzzy Matching	8
Metodologías simples	8
Abordaje del problema	9
Plan de trabajo	10
Desarrollo del Proyecto	10
Riesgos	12
Datos a utilizar	13
Herramientas a utilizar	13
Limitaciones del proyecto	13
Análisis Exploratorio	14
Planteamiento de preguntas iniciales	14
Estructura del dataset	14
Análisis de valores faltantes y duplicados	15
Análisis de variables	16
Hipótesis	22
Conclusiones del análisis exploratorio de datos	23
Análisis de la muestra etiquetada	24
Caso de negocio	26
Enfoque de Solución	27
Búsqueda de coincidencias entre direcciones (Modelos NLP)	27
Geolocalización de direcciones	30
Metodologías a Implementar	30
Named Entity Recognition (NER)	31
Spancat	31
Fuzzy Matching	31
Seteo de Experimentación	31
Modelos NLP	31
Clasificación de coincidencias	32
Desarrollo de la solución	34
Modelos de NLP	34
Entrenamiento Modelo Spancat	35

Entrenamiento Modelo NER	36
Geolocalización	38
Interfaz de Usuario	42
Resultados	43
Proceso	43
Validación	44
Conclusiones	45
Potenciales Próximos Pasos	47
Anexo	48
Bibliografía	49

Objetivo del proyecto

OCASA es una empresa que ofrece servicios logísticos y de distribución a empresas. Su actividad central es la entrega de paquetería y correo a domicilios de todo el país.

Para realizar la logística y distribución de los paquetes, OCASA obtiene la información de las direcciones a través de sus clientes.

Actualmente, para encontrar la dirección proporcionada por el cliente, los operadores de OCASA las comparan con direcciones existentes en su base de datos, es decir, direcciones donde se entregaron con éxito paquetes anteriormente. Si se encuentra coincidencia exacta, la dirección del cliente se geolocaliza con la dirección coincidente en la base de OCASA. En caso de no encontrar coincidencia, se realiza la geolocalización manualmente.

Las direcciones que se encuentran en la base de datos de OCASA son lugares donde se ha hecho la entrega de un paquete en alguna ocasión. Sin embargo, esto no implica que las direcciones estén correctamente escritas. El repartidor puede haber encontrado una manera de llegar a la dirección específica mediante la consulta a personas de la zona, su propio conocimiento u otras razones.

Es importante tener en cuenta que OCASA no es propietaria ni responsable del diseño ni del contenido de las páginas web de sus clientes, no puede influir en la mejora de la escritura de la dirección. La mayoría de las direcciones que OCASA recibe están compuestas de un campo libre donde el receptor puede escribir su dirección como lo considera correcto. Por lo tanto, no están estandarizadas y suelen ser inexactas. Si bien existen algunos clientes que proporcionan la geolocalización exacta donde se debe entregar un paquete, para la mayoría de los casos, la geolocalización está a cargo de OCASA.

La geolocalización se puede obtener mediante comparación o por ruteo manual. Las direcciones se comparan por coincidencia exacta. Esto significa que un mínimo error puede resultar en que no se encuentre una coincidencia. Por ejemplo, si una dirección en la base de datos de OCASA contiene una tilde, como “Avenida Córdoba 3010”, y el cliente la escribe sin tilde, “Avenida Cordoba 3010”, no se encontrará coincidencia, a pesar de que ambas se refieran a la misma dirección. En este caso, la dirección del cliente se envía a revisión manual. Debido a las diferencias en la escritura de direcciones se lleva a una sobrecarga de revisión manual para los operadores de OCASA.

El proyecto tiene como objetivo probar la hipótesis de que al automatizar el proceso de coincidencias, incluyendo un modelo y técnicas de Natural Language Processing (NLP), se mejora la calidad de las direcciones para luego ser efectivamente geolocalizadas. De esta

manera, se reducirá la sobrecarga de revisión manual y se generará un impacto positivo en los costos asociados.

Medición de valor y KPIs a Impactar

Key Performance Indicators (KPIs) del proyecto

Para medir la efectividad del proyecto se analizan métricas que describen el nivel de automatización del proceso y la calidad del mismo. Estas son:

- Porcentaje de revisión manual.
- Porcentaje de geolocalizaciones automáticas.

El porcentaje de revisión manual mide la cantidad de direcciones que escapan al proceso de automatización y deben ser revisadas por un operario de OCASA. El porcentaje es el cociente entre las direcciones pasadas a revisión manual y las direcciones totales.

$$\% \text{ de revisión manual} = \left(\frac{\text{direcciones pasadas a revisión manual}}{\text{direcciones totales}} \right) * 100$$

El porcentaje de geolocalizaciones automáticas representa las direcciones que terminan el proceso con latitud y longitud asignada, calculado como:

$$\% \text{ de geolocalizaciones automáticas} = \left(\frac{\text{direcciones geolocalizadas automáticamente}}{\text{direcciones totales}} \right) * 100$$

Key Performance Indicators (KPIs) de la empresa

Dentro del esquema de costos de OCASA se encuentran dos que son impactados por este proyecto:

- Costo de revisión manual.
- Costo por envíos extra.

El principal objetivo de esta herramienta es la reducción de la revisión manual de direcciones, esto se consigue mediante la automatización de la búsqueda de coincidencias.

Los analistas de OCASA están a cargo, entre otras funciones, de la revisión manual de direcciones. Según información de OCASA, se estima que cada analista tarda 30 segundos en realizar la revisión manual de una dirección.. Por lo tanto, el costo de revisión está ligado al salario de cada analista. Es importante aclarar que la empresa puede, tanto prescindir de analistas como reubicar sus horas de trabajo en otras tareas.

$$\text{Costo de revisión manual} = \% \text{ de revisión manual} * \text{direcciones por día} \\ * \text{tiempo de revisión} * \text{salario}$$

Los valores estimados de estas variable son:

- Porcentaje de revisión manual (actual) = 35%
- Direcciones por día que ingresan a OCASA en promedio = 55.000.
- Tiempo de previsión, tiempo que tarda un analista en geolocalizar una dirección por revisión manual = 30 segundos. (0.008 horas)
- Salario por hora de analista = 2,19 usd.

En segundo lugar, OCASA espera que con la utilización de esta herramienta reduzca la cantidad de paquetes que no se entregan debido a una mala geolocalización. Los paquetes no entregados por la mala geolocalización requieren una segunda visita al domicilio, esto impacta directamente el costo por envío. Se debe repetir el proceso de ruteo y entrega, lo cual duplica este costo.

$$\text{Costo por envíos extra} = N * 0,7$$

N = cantidad de paquetes que necesitan más de una envío

Al mejorar la exactitud de la geolocalización, se reducen las segundas visitas a los domicilios, disminuyendo el costo por envío extra.

Entregables y Outputs del proyecto

Este proyecto tiene como fin desarrollar una herramienta, GeoCenter, que permita buscar coincidencias entre direcciones en la base de datos de OCASA y direcciones de clientes.

El entregable de este proyecto consistirá de una aplicación. La aplicación recibe direcciones sin geolocalizar y entrega las direcciones geolocalizadas e indica cuales deben ser revisadas manualmente.

El propósito principal de esta herramienta es reducir la cantidad de paquetes no entregados por mala geolocalización. La entrega fallida de un paquete representa una pérdida significativa para OCASA, por lo que esta herramienta contribuye a minimizar este problema.

Investigación de metodologías

Las direcciones que se tienen para trabajar son campos de texto abierto. El usuario puede ingresar lo que desee sin ninguna restricción más que la cantidad de caracteres. Por lo que, se usará algún método de Natural Language Processing (NLP), en específico Named Entity Recognition (NER) y Fuzzy Matching. A continuación se describen, en líneas generales, estos conceptos:

Natural Language Processing (NLP)

NLP es el campo de la Inteligencia Artificial que se centra en el lenguaje humano. Se utilizan algoritmos y modelos que buscan comprender, analizar y generar texto similar a como lo haría un humano. Algunas de las capacidades incluyen: analizar sintácticamente el lenguaje humano, extraer metadata importante de textos, resumir textos, etc.

El NLP se utiliza en aplicaciones como la traducción automática, el análisis de sentimientos, la extracción de información y los *chatbots*, entre otros.

Named Entity Recognition (NER)

NER se encarga de extraer entidades de un texto. Las entidades son palabras o grupos de palabras que corresponden a un tipo específico de datos. Este puede ser numérico, temporal, nominal (como nombres de personas o lugares). NER toma un texto y trata de encontrar entidades determinadas previamente. Originalmente se usó para encontrar sujeto y predicado en oraciones, algunas de sus aplicaciones hoy en día son: búsqueda, recomendación, análisis de noticias periodísticas.

Existe la posibilidad de crear un *custom NER* (NER personalizado) que identifique entidades establecidas por el usuario. Para este proyecto es posible aplicar un *custom NER* con *labels* (etiquetas) para cada parte específica de una dirección, por ejemplo: nombre de la calle, número, entre calles, etc.

Spancat

Spancat, también conocido como SpanCategorizer, es una variante de NER que se utiliza para categorizar segmentos arbitrarios y superpuestos en textos. En algunos casos, puede comprender oraciones un poco más complejas que NER.

Una tarea común en NLP es extraer segmentos de texto, incluyendo frases largas o expresiones anidadas. NER puede no ser la herramienta más adecuada para este problema, ya que predice etiquetas basadas en tokens individuales que son muy sensibles a los límites. Esto

es eficaz para nombres propios y expresiones autocontenidas, pero menos útil para otros tipos de frases o segmentos superpuestos. Spancat permite etiquetar segmentos arbitrarios y potencialmente superpuestos de texto. Dado la variabilidad de las direcciones spancat puede ser una alternativa a NER.

Fuzzy Matching

Fuzzy Matching busca y compara cadenas de texto de manera aproximada en vez de exacta. Permite encontrar coincidencias que son similares pero no necesariamente idénticas. Se basa en algoritmos que calculan una puntuación de similitud o distancia entre dos cadenas de texto o valores. Estos algoritmos evalúan cuánto se parecen las cadenas en función de diversos criterios.

En este proyecto se usará para encontrar coincidencias incluso cuando las direcciones no son perfectas, lo que es especialmente útil en el procesamiento de datos.

Metodologías simples

Aunque existen métodos más simples para separar texto en partes, como:

- Funciones de Excel.
- Expresiones Regulares.

Estos enfoques no son adecuados para abordar la complejidad de la separación de direcciones.

Las funciones de excel como EXTRAER() o ENCONTRAR() no funcionarán debido a que no todas las personas escriben las direcciones en el mismo orden y puede haber más de una palabra para cada componente de la dirección. Además, se debe tener en cuenta la variabilidad de las direcciones en Argentina, que incluye calles con nombres o números, calles bis, la elección del destinatario de incluir o no palabras como "calle", "avenida", "ruta" o sus abreviaturas, la presencia de pasajes y la numeración de los domicilios, entre otras variaciones.

Se ha analizado la posibilidad de utilizar expresiones regulares (*regex*). Sin embargo, dado que las direcciones varían significativamente y los destinatarios tienen diferentes preferencias al escribir sus direcciones, las expresiones regulares no serían una solución adecuada en este caso.

Abordaje del problema

En primer lugar, se investiga el proceso actual mediante el cual se determina si una dirección se dirigirá a revisión manual o no.

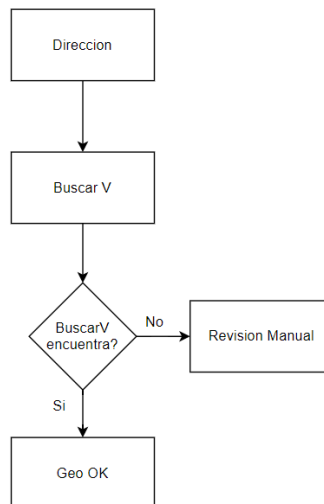


Figura 1 - Proceso actual.

Luego, se analiza en qué partes de este proceso se pueden incorporar automatizaciones que geolocalicen direcciones de manera confiable. Así, se puede disminuir la cantidad de direcciones que van a revisión manual.

Posteriormente, se adquiere una base de datos de direcciones, la cual es sometida a un análisis cualitativo. El propósito del mismo es generar *insights* sobre el estado inicial de las direcciones.

Una de las opciones de automatización que se considerarán para el nuevo proceso es un modelo de NLP, el cual utilizará etiquetas personalizadas que sean relevantes para identificar las partes de una dirección.

Otra opción de automatización de geolocalización es el Fuzzy Matching, que compara la similitud de textos de direcciones existentes con las nuevas direcciones. Este proceso es fundamental y se utilizará para mejorar la performance del modelo NLP y lograr una geolocalización confiable.

Como recurso extra se considera la utilización de la API de Google Maps que puede proporcionar direcciones completas.

Ambas se pueden utilizar en los casos donde se necesite mejorar la performance del modelo de NLP para conseguir una geolocalización confiable.

Adicionalmente, se debe llevar a cabo un etiquetado manual utilizando las etiquetas previamente definidas. Una vez que se tiene los datos etiquetados, se comienzan a entrenar y comparar modelos de NLP. Se evaluarán modelos como NER y *spancat*, teniendo en cuenta métricas específicas para determinar el modelo óptimo.

Una vez que el modelo está listo y se analiza el uso de las opciones anteriores, se define el proceso final por el cual se decide si se revisa manualmente una dirección o no.

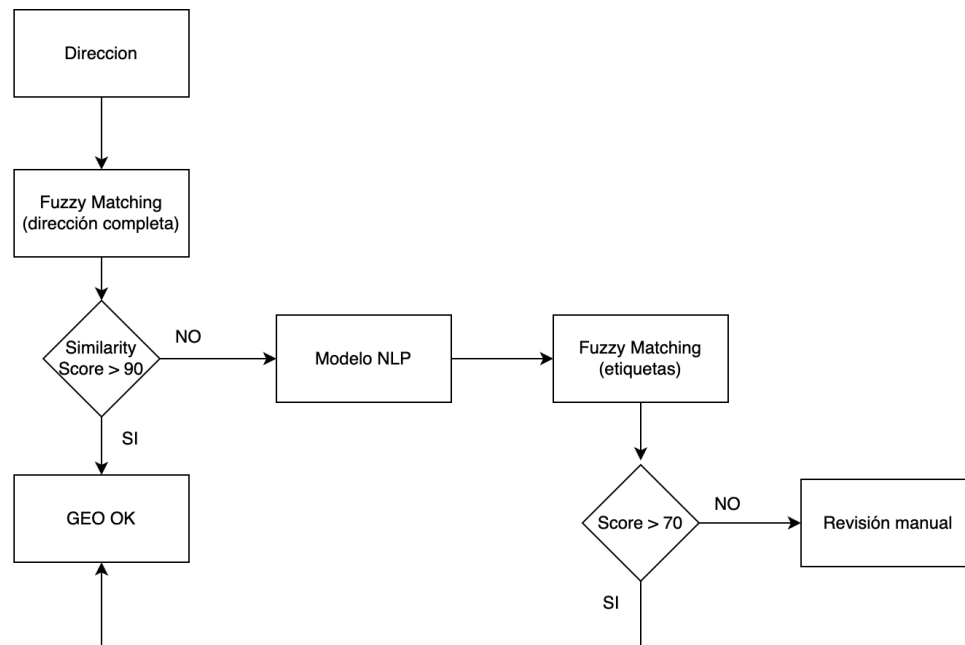


Figura 2 - Proceso con la solución propuesta.

Posteriormente, se inicia la construcción de la aplicación que constituirá el entregable final del proyecto. Esto se logra mediante el uso de código Python para el procesamiento y el desarrollo de una interfaz de usuario.

Por último, se analizan los KPIs propuestos para medir el impacto del proyecto y se harán los ajustes necesarios para conseguir la mejor performance posible.

Plan de trabajo

Desarrollo del Proyecto

A continuación se van a explicar las fases a realizar para el desarrollo del proyecto.

1. **Definir requisitos y objetivos del proyecto:** determinar qué procesos se deben automatizar, qué datos se deben utilizar y geolocalizar, qué resultados se espera obtener.
2. **Obtener datos de direcciones de la empresa:** recopilar la información acerca de las direcciones en donde la entrega fue exitosa.
3. **Análisis Exploratorio de Datos:** se va analizar cómo es la situación actual de la empresa, es decir, cómo reciben y procesan los datos en la empresa.
4. **Definir etiquetas:** una vez hecho el análisis exploratorio de datos, se definirán las etiquetas a utilizar para el modelo.
5. **Etiquetar manualmente un conjunto de direcciones**
6. **Entrenamiento del modelo:** el paso a seguir es entrenar los dos modelos, NER y Spancat, utilizando el conjunto de direcciones etiquetadas.
7. **Puesta a prueba (Testing):** En esta fase, se ponen a prueba los modelos utilizando datos no vistos por el mismo. Se calculan distintas métricas para entender la calidad del rendimiento del modelo.
8. **Elección del modelo:** con los resultados obtenidos de los distintos modelos, se elegirá el más conveniente para resolver el problema de la empresa.
9. **FuzzyMatching:** se prueba la técnica fuzzy matching para las direcciones nuevas y las pertenecientes a la base de datos de la empresa.
10. **Definición del proceso de Geolocalización**
11. **Evaluar y validar resultados**
12. **Interfaz de usuario**
13. **Presentación al cliente:** por último, se va a preparar una presentación para mostrarle a OCASA cómo impactaría la implementación de la herramienta en su empresa.

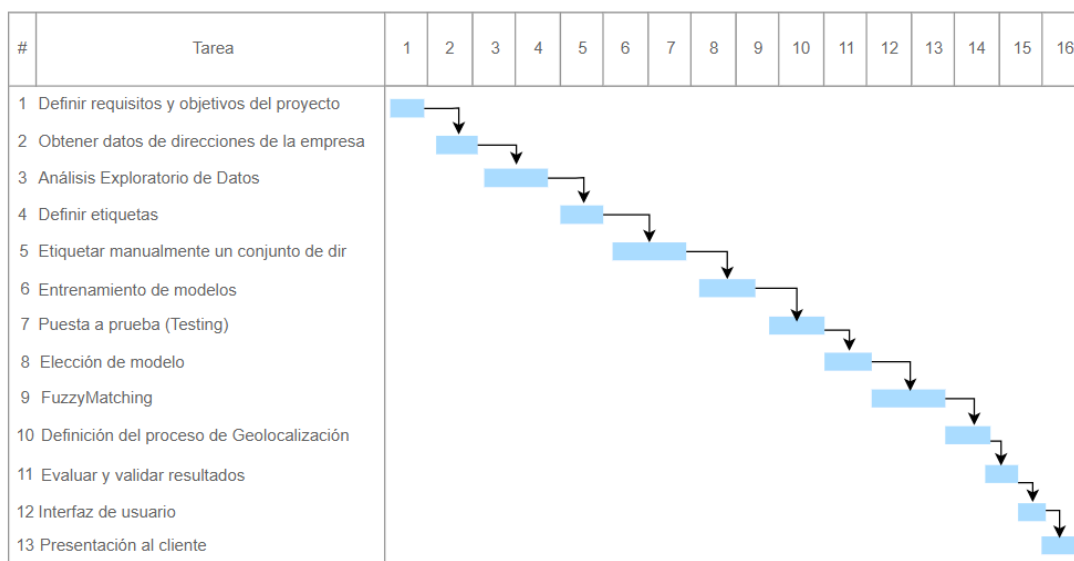


Figura 3 - Diagrama de Gantt del Proyecto.

Como se puede ver en el diagrama de Gantt, cada fase depende de que se complete la etapa anterior. Es por este motivo que el proyecto sigue la metodología Waterfall, en donde cada etapa debe finalizar antes de realizar la siguiente.

Riesgos

Se deben tener en cuenta los riesgos que puede tener el desarrollo del proyecto. La siguiente tabla muestra los posibles riesgos y la estrategia que se va a utilizar para mitigar dichos riesgos.

Riesgo	Ocurrencia	Impacto	Mitigar
<i>Direcciones incompletas o incorrectas</i>	Alta	Medio	Validar y verificar los datos antes de normalizarlos.
<i>Variaciones en las direcciones</i>	Alta	Alta	Desarrollar un sistema que sea capaz de distinguir las direcciones de las distintas regiones o provincias.
<i>Privacidad de los datos</i>	Bajo	Alta	Utilizar medidas de seguridad para la protección de datos

Tabla 1 - Análisis de riesgos.

Datos a utilizar

Como se mencionó anteriormente, los datos a utilizar van a ser las direcciones de paquetes anteriores que van a ser proporcionadas por OCASA. Las direcciones son de tipo texto y contienen información como el nombre de la calle, el número, entre que calles está, entre otros datos.

Estos datos son importantes para el proyecto ya que son la base para el proceso de geolocalización de direcciones. Analizando las direcciones se podrá identificar patrones que permitan el desarrollo de un modelo que geolocalice las direcciones de manera más efectiva por lo que ayudará a mejorar el proceso de entrega de pedidos.

Herramientas a utilizar

Durante el transcurso del proyecto, se utilizarán diferentes herramientas tanto para el desarrollo del mismo como para el seguimiento y la presentación.

Para el desarrollo del proyecto se va a utilizar el lenguaje de programación Python. Para el etiquetado de los datos y la realización del modelo se va usar Spacy y Label Studio.

Por otro lado, para el seguimiento del proyecto se va a emplear el uso de Miró. Por último, para la presentación del proyecto se va a usar la herramienta Canva para el desarrollo colaborativo de la visualización del proyecto.

Limitaciones del proyecto

Este proyecto presenta dos limitaciones:

1. **Performance en producción:** A la hora de comparar métricas para medir el éxito del proyecto en producción, se encuentra la limitante de tiempo. Para que las comparaciones sean válidas se necesita más tiempo del que se tiene.
Por ejemplo, a la hora de visualizar si hubo un impacto en la cantidad de reclamos por mala geolocalización se deberán medir datos de al menos tres meses para poder evaluar si el impacto se produjo por la aplicación del modelo o porque los reclamos bajaron por otros factores.
2. **Eficacia de los modelos:** Los modelos NLP empleados en el nuevo proceso nunca alcanzarán una eficacia del 100%, lo cual es inherente a la naturaleza de los mismos. La finalidad de su uso es reducir la revisión manual, no eliminarla.

Análisis Exploratorio

Planteamiento de preguntas iniciales

El objetivo de este proyecto es desarrollar una herramienta personalizada para OCASA que optimice la geolocalización de las direcciones de sus clientes. La geolocalización de direcciones es esencial para mejorar la eficiencia en la entrega de pedidos.

Sin embargo, para poder construir un proceso efectivo, se debe comprender la situación actual con respecto a cómo se reciben y procesan las direcciones en la empresa. Entender la situación actual implica saber cómo se gestionan las direcciones actualmente en la empresa, desde su recepción hasta su procesamiento y uso en el proceso de entrega. También identificar cualquier desafío o problema existente en el flujo de trabajo actual que pueda resolverse con la implementación de la herramienta de geolocalización. Por ejemplo,

- ¿Cómo se componen las direcciones?
- ¿Qué direcciones se recopilan actualmente en OCASA?
- ¿Hay variabilidad en la forma en que los clientes proporcionan sus direcciones?
- ¿Cómo se utilizan actualmente las direcciones en el proceso de entrega de pedidos?

El análisis exploratorio de datos se realizó en Python utilizando los datos otorgados por la empresa.

Estructura del dataset

Esta base se actualiza mensualmente, se trabajó con la correspondiente al mes de agosto. El dataset de OCASA tiene registros 1.544.403 y 7 variables. La *tabla 2* y la *figura 4* proporcionan una descripción detallada del conjunto de datos en cuestión.

Nombre de la variable	Descripción	Tipo de dato
Población	El nombre de la población o ciudad asociada a la dirección.	String
Codigo_Postal	El código postal de la dirección.	Float64
Calle	El nombre y el número de la calle.	String
Dirección_Latitud	La latitud geográfica de la	Float64

	dirección.	
Dirección_Longitud	La longitud geográfica de la dirección.	Float64
Check	Si dice "No cargado" es porque se necesita revisar la dirección debido a posibles problemas o errores.	String

Tabla 2 - Descripción de las variables del dataset.

	Poblacion	Codigo_Postal	Calle	Direccion_Latitud	Direccion_Longitud	Check
0	Capital Federal	1000.0	CORRIENTES 707	-34.603337	-58.376755	NaN
1	Capital Federal	1000.0	RECONQUISTA 660	-34.600162	-58.372725	NaN
2	Capital Federal	1000.0	CERRITO 1130	-34.595132	-58.382643	NaN
3	Capital Federal	1000.0	CERRITO 1130	-34.595132	-58.382643	NaN
4	Capital Federal	1000.0	LAVALLE 556	-34.602255	-58.374648	NaN
5	Capital Federal	1000.0	RECONQUISTA 484	-34.602223	-58.372602	NaN
6	Capital Federal	1000.0	AZOPARDO 455	-34.612910	-58.367974	NaN
7	Capital Federal	1000.0	MONTEVIDEO 825	-34.599202	-58.389588	NaN
8	Capital Federal	1000.0	RECONQUISTA 660	-34.600162	-58.372725	NaN
9	Capital Federal	1000.0	SUIPACHA 268	-34.605225	-58.379453	NaN

Figura 4 - Dataset de OCASA.

Análisis de valores faltantes y duplicados

Se procede a analizar los valores faltantes que se encuentran en la base de datos. Para esto, se realizó una tabla en donde por un lado se presentan las variables y por el otro la cantidad de valores faltantes que tienen y el porcentaje que representan esos valores para el dataset. A continuación, se puede observar dicha tabla:

	Cantidad_NA	Porcentaje_NA
Poblacion	4819	0.490591
Codigo_Postal	20953	2.133088
Calle	0	0.000000
Direccion_Latitud	0	0.000000
Direccion_Longitud	0	0.000000
Check	975621	99.321582

Figura 5 - Valores faltantes en el Dataset.

Con los resultados obtenidos, se puede destacar que el porcentaje de valores faltantes en las columnas de "Población" y "Código Postal" es notablemente bajo. Como consecuencia,

se eliminaron los registros que presenten valores faltantes (NA) correspondientes a esas variables. Esto permitió conservar la mayor cantidad de información relevante en relación con la variable "Calle", que representa la dirección de entrega.

Por otro lado, el atributo "Check" es un campo que se usa durante el proceso de verificación de las direcciones. Si el campo está vacío, indica que la dirección fue encontrada en el sistema de registro. De lo contrario, significa que la dirección no se puede encontrar en la base de datos interna y es posible requerir una búsqueda más detallada y precisa utilizando la API de Google Maps para determinar la ubicación del destino del envío. La interpretación y el seguimiento de este campo proporcionan valiosos insights sobre el estado y la veracidad de las direcciones, lo que resulta fundamental en la toma de decisiones relacionadas con el proceso de entrega de OCASA.

A su vez, se realizó un análisis de duplicados. Se han identificado 562.118 registros duplicados. En la *figura 6*, se observa cómo una misma dirección tiene las mismas coordenadas. Se eliminaron los valores duplicados de la base de datos, quedando un total de 982.285 registros.

2	Capital Federal	1000.0	CERRITO 1130	-34.595132	-58.382643	NaN
3	Capital Federal	1000.0	CERRITO 1130	-34.595132	-58.382643	NaN

Figura 6 - Registros duplicados.

Análisis de variables

Una vez hecha la limpieza de la base de datos, se continuó analizando las variables categóricas.

En primer lugar se realizó un gráfico de barras de la variable población. Se destacaron las poblaciones con mayor porcentaje de direcciones. En la *figura 7*, se puede observar que las poblaciones con mayor porcentaje de direcciones son: Capital Federal, Córdoba, Rosario, La Plata, Santa Fe, Bahía Blanca, Rafaela y Tandil. Con un 12,79% Capital Federal es la población con mayor porcentaje de direcciones, es decir, con la mayor cantidad de envíos a esa zona.

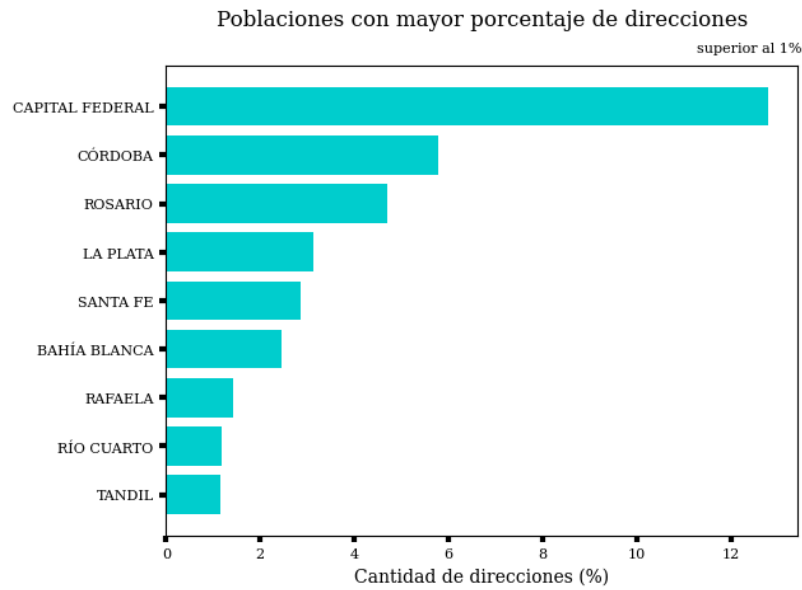


Figura 7 - Gráfico de Barras de las Poblaciones con mayor porcentaje de direcciones.

Con el objetivo de obtener una visión general de los datos con los que se está trabajando, se visualizó un mapa de Argentina. Se verificó que todas las coordenadas estén dentro de Argentina. Por lo tanto, se seleccionó un conjunto de 100 registros de manera aleatoria con el propósito de llevar a cabo la geolocalización. En la *figura 8*, se pueden apreciar pedidos con diversas ubicaciones en toda la región de Argentina, desde Salta hasta Chubut.

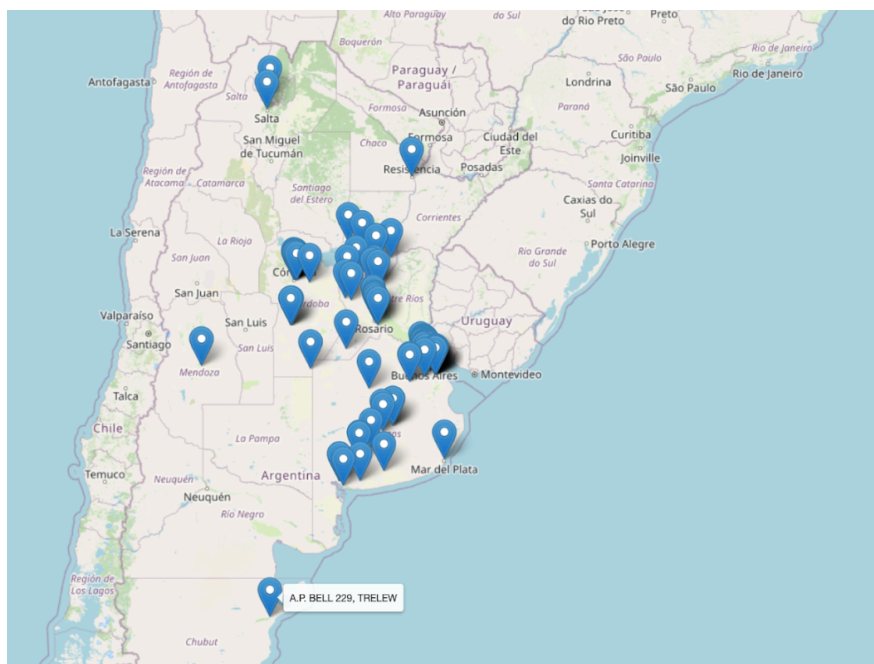


Figura 8 - Mapa de direcciones.

Por otro lado, se analizó la variable Calle. Esta variable contiene las direcciones precisas a las cuales OCASA debe realizar sus entregas. Por este motivo, se investigaron:

- Las distintas "secciones" que componen una dirección.
- Las palabras utilizadas para describir estas "secciones" específicas.
- Identificar posibles etiquetas que puedan ser empleadas para el modelo NER.

En la *figura 9*, se observa un histograma que ilustra la cantidad de palabras por dirección. La mayoría de las direcciones consisten en dos o tres palabras, mientras que las direcciones compuestas por más de cuatro palabras son una minoría. Esta distribución se asemeja a una distribución logarítmica. Además, la mediana de palabras por dirección es de aproximadamente 3 palabras, con una longitud máxima de 16 palabras y una longitud mínima de una palabra.

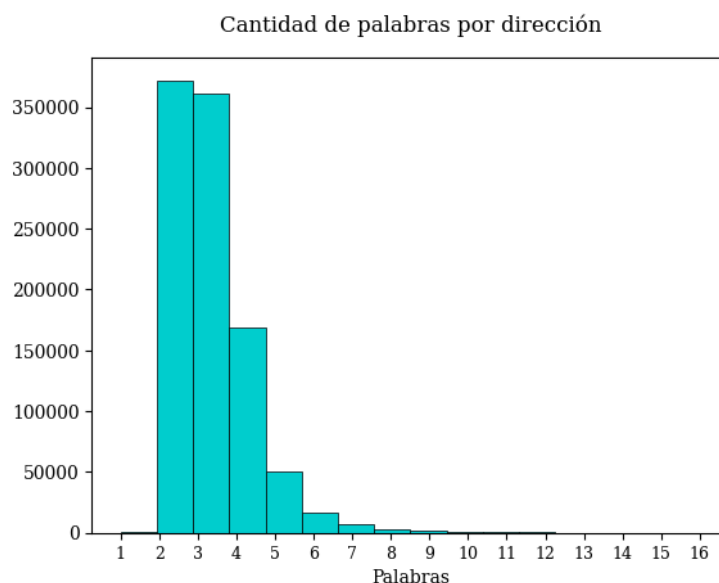


Figura 9 - Histograma de la cantidad de palabras por dirección.

Como se puede ver en la *figura 10*, la mayoría de las direcciones están compuestas por 3 palabras. La direcciones tienen el nombre de la calle, “Eduardo Madero”, y el número “900”.

	Poblacion	Codigo_Postal	Calle	Direccion_Latitud	Direccion_Longitud	Check
13	CAPITAL FEDERAL	1000.0	EDUARDO MADERO 900	-34.597293	-58.370017	NaN
19	CAPITAL FEDERAL	1431.0	CRISOLOGO LARRALDE 5670	-34.565199	-58.499864	NaN
22	CAPITAL FEDERAL	1431.0	LA PAMPA 4540	-34.576950	-58.475012	NaN
24	CAPITAL FEDERAL	1107.0	JUANA MANSO 1181	-34.610988	-58.362705	NaN
27	CAPITAL FEDERAL	1037.0	BARTOLOME MITRE 1648	-34.607934	-58.389918	NaN

Figura 10 - Direcciones con 3 palabras.

A su vez se realizó un análisis para identificar posibles valores atípicos en la variable Calle. Para este análisis, se empleó un diagrama de caja. En la *figura 11*, se puede apreciar la presencia de valores atípicos para la cantidad de palabras que componen una dirección. Se puede ver que hay direcciones que contienen más de 8 palabras, mientras que la mayoría tiene entre 2 a 4 palabras.

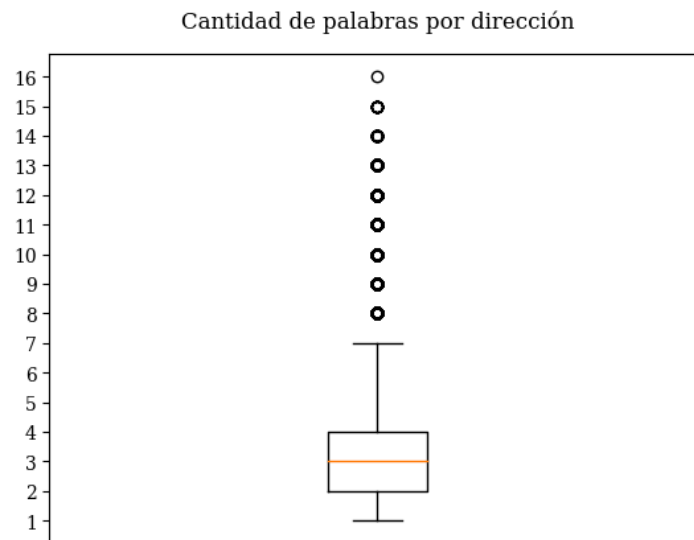


Figura 11 - Diagrama de caja de la cantidad de palabras por dirección.

En la siguiente figura, a modo ilustrativo, se muestra una de las direcciones más largas del conjunto de datos.

RUTA 20 KM 14 Y MEDIO - MZ 64 LTE 04 - SIETE SOLES - VERANDA

Figura 12 - Dirección compuesta por 16 palabras.

Adicionalmente, se realizó una nube de palabras con las palabras que más se repiten en las direcciones que brindan los clientes. Las más mencionadas fueron: "Calle", "San Martín", "De Mayo", "De Julio", "Rivadavia", "Sarmiento" y "Santa Fe".

solo a modo de explorar el contenido de las direcciones, no se harán cambios permanentes a la base. Se realizó en tres pasos:

1. Eliminar números.
2. Eliminar *stopwords* o palabras vacías, conocidas como palabras de relleno, las cuales no aportan información al contenido de oraciones, en este caso, direcciones. Se consideraron como stopwords para este caso especial a las siguientes: "el", "la", "del", "de", "lo", "las" y "los".
3. Eliminar algunos nombres de calles comunes vistos en la nube de palabras. Estos incluyen, entre otros: "San", "Martin", "Mayo", "Julio", "Santa", "Fe", "Juan", "Belgrano", "General", etc.

Luego de esto se volvieron a calcular los estadísticos como la mediana de palabras por dirección, el cual se redujo a 1. A continuación se puede ver el nuevo histograma con de la cantidad de palabras por dirección:

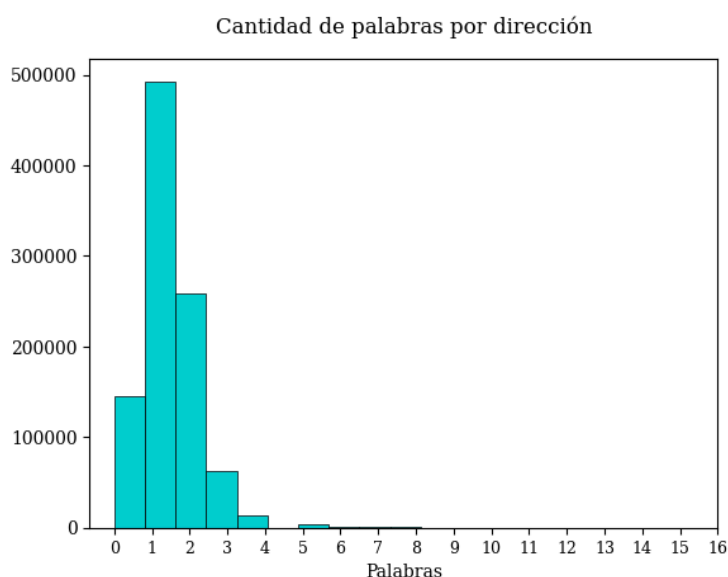


Figura 15 - Histograma de palabras por dirección.

El máximo fue de 13 palabras y mínimo de 0. Las direcciones de largo 0 se dieron porque, por ejemplo, si la dirección era “San Martín 455”, en el primer paso de la limpieza se eliminó “455” y en el tercero “San Martín”. Esto no es un problema debido a que el propósito de esta limpieza es buscar palabras que puedan servir para identificar las partes de una dirección. También se realizó por segunda vez la nube de palabras, donde se evidenció cómo aumenta la importancia de palabras como “pasaje”, “calle bis” y “avenida”.



Figura 16 - Nube de palabras sobre las direcciones.

Para ver con claridad los resultados de esta nube de palabras, se volvieron a cuantificar las veinte palabras más repetidas en las direcciones y fueron las siguientes:

Palabra	Cantidad	Porcentaje
CALLE	60637	4.68
AVENIDA	23364	1.80
Y	9545	0.74
AV	8778	0.68
AV.	8702	0.67
PASAJE	7928	0.61
BIS	7764	0.60
ENTRE	5391	0.42
PISO	4361	0.34
ALEM	4329	0.33
ALVEAR	4148	0.32
MENDOZA	4093	0.32
CORRIENTES	4089	0.32
JUSTO	4204	0.32
FRANCISCO	4210	0.32
B.	4064	0.31
ROCA	4043	0.31
RUTA	4042	0.31
LAVALLE	4003	0.31
SANTIAGO	3992	0.31

Figura 17 - Palabras más utilizadas en las direcciones.

Hipótesis

La hipótesis del análisis exploratorio de datos es que limpiando las direcciones, eliminando números, *stopwords* y nombres de calles frecuentes, pueden emerger patrones de

lenguaje coloquial utilizados por las personas al describir direcciones. Estos patrones pueden diferenciarse del lenguaje formal que se encuentra en registros nacionales o plataformas como Google Maps.

Esta hipótesis sugiere que al realizar una limpieza de las direcciones, se espera encontrar un conjunto de palabras y expresiones que reflejan cómo las personas suelen describir direcciones de manera informal. La diferenciación entre lenguaje informal y formal es relevante para este proyecto, ya que las direcciones se ingresan en un campo de texto libre y es posible que las personas utilicen un lenguaje menos estructurado y más coloquial al hacerlo.

Conclusiones del análisis exploratorio de datos

Los resultados del EDA evidencian que la hipótesis planteada anteriormente es así. Luego de quitar números, *stopwords* y nombres de calles comunes las palabras más frecuentes en una dirección son:

- | | |
|-----------|----------|
| ● Calle | ● Pasaje |
| ● Avenida | ● Bis |
| ● Y | ● Entre |
| ● Av | ● Piso |
| ● Av. | ● Ruta |

Esta variabilidad de palabras refuerza la necesidad de tener un enfoque basado en modelos complejos como NLP y no algo más simple como regex. A partir de estos resultados se obtuvieron algunos ejemplos de las palabras usadas en el lenguaje coloquial para separar las distintas partes de una dirección. Estas contribuyen a definir las etiquetas a utilizar para entrenar al modelo de NER o spancat. Las etiquetas son:

- | | |
|-----------------|------------------------------------|
| ● nombreCalle | ● POI (<i>point of interest</i>) |
| ● tipoCalle | ● localidad |
| ● numeroCalle | ● entreCalles |
| ● tipoDomicilio | |

Análisis de la muestra etiquetada

La muestra está formada por 1.436 registros, aproximadamente 900 registros fueron extraídos de la base original de manera aleatoria. Esto resultó en una muestra desbalanceada. Luego se corrigió agregando alrededor de 500 registros de las clases minoritarias.

Es importante aclarar que esta muestra es etiquetada manualmente por el equipo del proyecto. Este etiquetado es un proceso que consume mucho tiempo de trabajo. Por esto, al detectar que la muestra estaba desbalanceada, se optó por mejorarla, en lugar de tomar una nueva y etiquetar de cero.

Los principales estadísticos de la muestra, comparados con el conjunto de datos completo, son los siguientes:

Principales estadísticos	Muestra	Dataset de OCASA
Palabras promedio por dirección	4	3
Largo máximo de una dirección	14	16
Largo mínimo de una dirección	2	1
Largo de dirección más frecuente (moda)	3	2
Cantidad de registros	1.436	982.285

Tabla 3 - Principales estadísticos de la muestra y de la base de datos de OCASA.

Adicionalmente, se graficó un histograma que muestra la distribución de la cantidad de palabras por dirección.

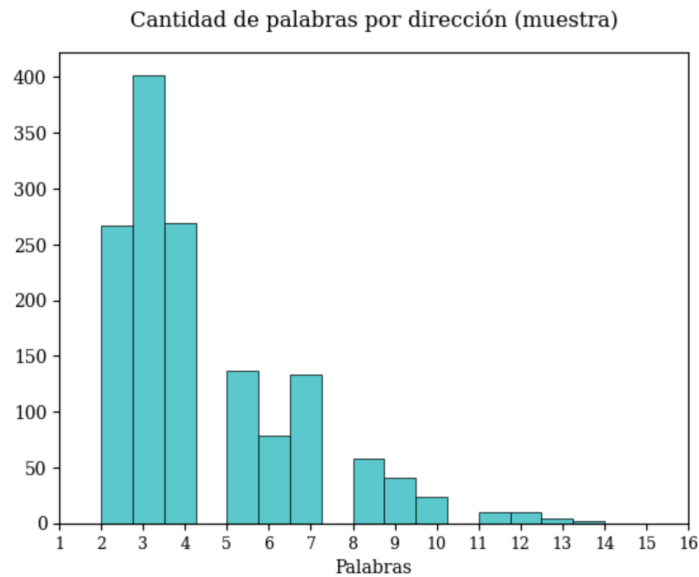


Figura 18 - Histograma de la cantidad de palabras por dirección en la muestra.

Por último, se muestra la frecuencia con la que se utilizó cada label para el etiquetado de datos. Se ve como los labels `numeroCalle` y `nombreCalle` están presentes en todas las direcciones, esto es porque son fundamentales para la geolocalización de una dirección. Por otra parte, el resto de las etiquetas aparecen en una menor cantidad de casos porque se usan para describir direcciones inusuales.

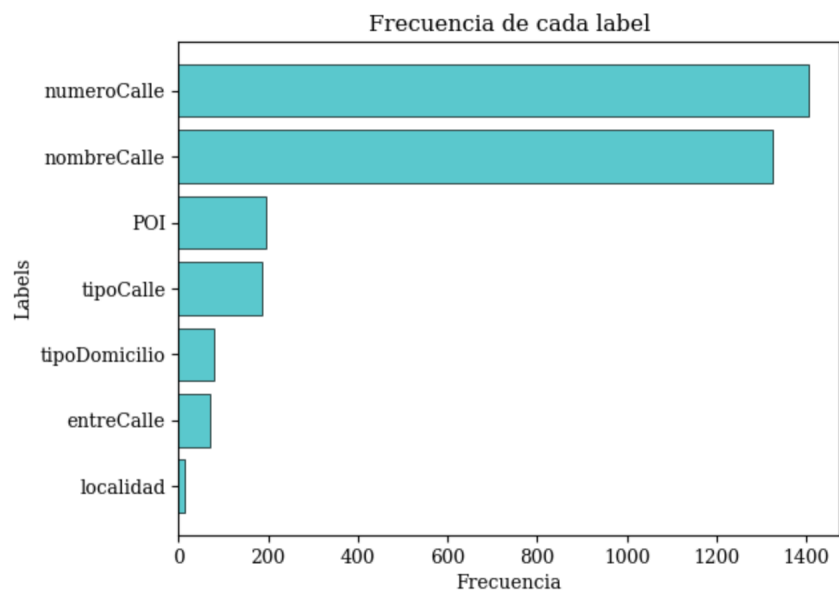


Figura 19 - Gráfico de barras sobre la frecuencia de cada label en las direcciones de muestra.

Caso de negocio

Dentro del proceso de geolocalización de direcciones en OCASA existe una posibilidad de mejora. Actualmente, se trabaja con Excel en donde se busca en la base de datos de la empresa si una dirección nueva coincide de manera exacta con las existentes. El nivel de coincidencia es muy estricto, generando una sobrecarga de revisión manual (RM). El proceso resulta en un 35% de revisión manual, es decir, de todas las direcciones que ingresan a OCASA por día, el 35% deben ser geolocalizadas manualmente por un analista. En consecuencia, el porcentaje de geolocalizaciones automáticas es del 65%.

El trabajo de geolocalización de direcciones es realizado por grupos de analistas con un salario de \$510.000, estos son datos proporcionados por OCASA. Se tendrá en cuenta el salario por hora de los mismos, el cual es 3,19 usd/hr.

Según datos de OCASA, los envíos que figuran con motivo de “No se ubica domicilio/faltan datos” y “Mala geolocalización” componen cerca de un 4% del total de los envíos. En promedio, OCASA procesa 1.000.000 de envíos mensuales, resultando en 40.000 envíos con mala geolocalización.

Teniendo en cuenta que una visita a un domicilio cuesta 0,7 usd, el costo de no entregar estos paquetes es de 28000 usd mensuales.

Con la solución propuesta de automatización de geolocalización de direcciones se plantean tres escenarios posibles de mejora:

- Escenario Conservador
- Escenario Moderado
- Escenario Optimista

El escenario conservador propone una reducción de la revisión manual al 25% y un aumento de geolocalización automática el 75%. En consecuencia, los envíos con mala geolocalización resultan en un 2,9%.

Luego, se plantea un escenario moderado, esto es a lo que se espera llegar con el proyecto. La revisión manual sería de un 15%, resultando en un 1,7% de envíos con mala geolocalización y la geolocalización automática sería de un 85%

Por último, se presenta un escenario optimista, en el cual la revisión manual sería del 5%, esto significa que solo un 0,6% de los envíos no serían entregados por mala geolocalización y la geolocalización automática sería de un 95%.

En las siguientes tablas se compara la reducción de costos del estado actual de OCASA y los tres escenarios propuestos:

Escenarios	RM	% Geolocalización automática	Costo RM	Costo de envío diario	% mala geo
Actual	35%	65%	\$491,26	\$1.540,00	4,0%
Conservador	25%	75%	\$350,90	\$1.100,00	2,9%
Moderado	15%	85%	\$210,54	\$660,00	1,7%
Optimista	5%	95%	\$70,18	\$220,00	0,6%

Tabla 4 - Escenarios propuestos.

Enfoque de Solución

El resultado esperado del proyecto es la optimización de procesos logísticos y la mejora de la calidad de datos de direcciones para OCASA. El objetivo principal es desarrollar una herramienta basada en Python que integre:

- Búsqueda de coincidencias de direcciones.
- Geolocalización de las mismas.

Búsqueda de coincidencias entre direcciones (Modelos NLP)

Se buscarán coincidencias entre las nuevas direcciones y las pertenecientes a OCASA. La forma elegida para hacer esto es a través de un modelo de NLP. El modelo etiqueta entidades dentro de un texto. El objetivo es etiquetar tanto en las direcciones existentes como las nuevas y poder realizar comparaciones entre etiquetas con el fin de encontrar coincidencias.

El paso previo a entrenar el modelo es la definición de etiquetas y el etiquetado de las direcciones con las mismas. Estas etiquetas son:

- **nombreCalle:** representa el nombre de la calle (generalmente es una palabra pero también puede ser un número).
- **tipoCalle:** representa el tipo de calle, por ejemplo: avenida, pasaje, ruta, la palabra calle en sí, etc.
- **numeroCalle:** representa el número del domicilio.
- **tipoDomicilio:** representa el tipo de domicilio, por ejemplo: edificio, casa, departamento.

- **POI (*point of interest*):** representan lugares destacados que se incluyen cuando se ingresa la dirección, por ejemplo comercios cercanos, plazas, instituciones.
- **localidad:** representa la localidad.
- **entreCalles:** representa las calles entre las cuales está la dirección, incluye los nombre de las dos calles y la palabra que se utiliza para separarlas, por ejemplo “entre”, “y”, etc.

Una vez definidas las etiquetas, se procede al etiquetado manual de una muestra de direcciones. Este proceso de etiquetado manual es fundamental para enseñar a los modelos a identificar entidades en las direcciones. Este etiquetado se realiza con la herramienta LabelStudio. En la figura 20, se ejemplifica como es el proceso de etiquetado manual.



Figura 20 - Ejemplos de direcciones etiquetadas manualmente en LabelStudio.

Una vez que la muestra fue etiquetada, se prosigue a entrenar los modelos: uno con *spancat* y otro con NER. Se hace una partición de la muestra en tres grupos distintos: el conjunto de entrenamiento (*train*), el conjunto de prueba (*test*) y validación (*dev*).

Conjunto de datos	Porcentaje	Cantidad
Entrenamiento (train)	70%	1.005
Prueba (test)	20%	287
Validación (dev)	10%	144
Muestra (total)	100%	1.436

Tabla 5 - Descripción de los datasets.

Para evaluar el rendimiento de los modelos, se tendrá en cuenta el *recall*, *precision* y *F1*. Si bien, la exactitud suele ser la métrica principal de modelos de clasificación, en casos como este donde existen muchas categorías (y con un gran desbalance entre ellas) no es suficiente.

El *recall* mide la capacidad de un modelo para identificar y recuperar correctamente todos los casos positivos existentes en un conjunto de datos. Esta es una métrica adecuada para el proyecto debido a que el objetivo del modelo es recuperar la mayor cantidad de etiquetas de una dirección, incluso si algunas etiquetas están mal posicionadas (falsos positivos). El *recall* es la métrica principal a la hora de evaluar modelos. Se calcula de la siguiente manera:

$$Recall = \frac{TP}{TP + FN}$$

TP: True Positive

FN: False Negative

En segundo lugar, se mide la *precision* de los modelos. Se calcula como la proporción de predicciones correctas sobre el total de predicciones positivas en el conjunto de datos:

$$Precision = \frac{TP}{TP + FP}$$

TP: True Positive

FP: False Positive

F1 es la métrica que combina la *precision* y el *recall*, es útil como medida global y se calcula de la siguiente manera:

$$F1 = 2 \times \frac{Precision + Recall}{Precision * Recall}$$

Es importante mencionar que *recall* y *precision* se complementan, el primero ayuda a entender el porcentaje de positivos que el modelo encuentra y el segundo para entender la cantidad de predicciones correctas que hace.

Geolocalización de direcciones

El objetivo de esta parte es reducir lo más posible el porcentaje de revisión manual. Las herramientas y procesos que se implementan para esta sección están ligados a la performance del modelo de NLP.

Si la performance del modelo de NLP es formidable, el proceso será el siguiente:

1. Se etiquetan las direcciones.
2. Se comparan sus etiquetas.
3. Se calcula un score para cada comparación.
4. Se agrega la geolocalización de score más alto.

Parte del desafío consiste en encontrar la mejor manera de traducir las comparaciones de direcciones (fuzzy matching y etiquetas) en un score que cuantifique que tan parecidas son dos direcciones entre sí, con el fin de geolocalizar una de ellas. En principio, se apunta a usar:

- Similarity score del fuzzy matching para direcciones completas.
- Comparaciones textuales con funciones de python para la comparación de etiquetas.

Antes de calcular el score se implementarán filtros para asegurarse que las comparaciones se hagan entre direcciones que pertenecen a la misma localidad y código postal. Adicionalmente, reduce la complejidad computacional del proceso.

A su vez, se incorporará una penalización, como requerimiento de OCASA. Esta impide que se aplique la misma latitud y longitud a direcciones que estén a más de 300 metros de distancia (en valor absoluto). Estas direcciones son tomadas como “malas”.

Además, se considera la posibilidad de que el modelo de NLP no cubra por completo las variantes de direcciones. Para esto se plantea que algunas geolocalizaciones puedan lograrse con recursos externos, como la API de Google Maps.

Metodologías a Implementar

Se utilizarán varias metodologías analíticas y no analíticas para abordar el proyecto. Estas metodologías son: Named entity recognition (NER), Fuzzy Matching y Spancat. A continuación, se explicarán cada una de estas metodologías.

Named Entity Recognition (NER)

El Named Entity Recognition es una técnica de procesamiento de lenguaje natural. Se utiliza para identificar y clasificar entidades en un texto. En este proyecto, NER se aplicará para identificar elementos clave en las direcciones, como nombres de calles, números de edificios y otras entidades relevantes, lo que facilitará la normalización y geolocalización de las direcciones.

Spancat

Spancat clasifica y categoriza datos de texto. Spancat se utilizará para etiquetar y categorizar partes específicas de las direcciones, estas etiquetas pueden superponerse. El beneficio de *spancat* es que puede aprender combinaciones como: tipo de calle + nombre de calle. Esta técnica puede ayudar a estandarizar y mejorar las direcciones.

Fuzzy Matching

Fuzzy Matching se utiliza para buscar y encontrar coincidencias aproximadas en cadenas de texto. En este proyecto, se aplicará para comparar direcciones en busca de similitudes incluso cuando existan errores tipográficos o variaciones en el formato. Esto es especialmente útil para identificar direcciones similares en la base de datos y mejorar la geolocalización de destinos.

Seteo de Experimentación

Modelos NLP

El primer paso para comenzar la experimentación con los modelos de SpaCy es obtener los datos etiquetados manualmente. Estos se descargan de la plataforma LabelStudio en formato JSON.

Se particionó la base de datos en tres: entrenamiento, testeo y validación. El conjunto de entrenamiento consiste en el 70% de los datos de la base original mientras que el conjunto de testeo está compuesto por el 20% y el de validación por el 10% restante.

Una vez particionado los datos deben ser formateados para ser compatibles con el modelo de SpaCy y este pueda identificar las etiquetas *custom* propuestas. Esto se realiza con Python.

Se experimentó con dos tipos de modelos: *spancat*, que permite etiquetas superpuestas y NER que no las permite. Cabe aclarar que el etiquetado manual debe ser

distinto para cada modelo, a continuación se muestra un ejemplo de las diferencia entre uno y otro.



Figura 21 - Etiquetado para el modelo spancat.



Figura 22 - Etiquetado para el modelo NER.

Se entrenan los modelos para que puedan identificar y clasificar entidades. Se logró mediante un algoritmo de procesamiento de lenguaje natural, donde su objetivo es extraer y etiquetar para poder entender mejor la información contenida en el texto. Ambos están configurados en español. Uno preparado para reconocer *spancat* y el otro no (NER).

Una vez entrenados, se probaron en el conjunto de testeo para evaluar su capacidad para reconocer y etiquetar correctamente las entidades. La medición del rendimiento se hará con el *recall*, la *precision* y el *F1*. A partir de estas métricas se decidirá cuál de los modelos es el mejor para implementar en la herramienta final.

Clasificación de coincidencias

Se experimentó con distintos tipos de clasificación de coincidencia de direcciones. La finalidad es lograr la mayor automatización del proceso posible, limitando la revisión manual hecha por los analistas sólo a los casos esenciales. Es importante recordar que se partió de una clasificación binaria, resultado de un buscarV de Excel, este era:

- 0 = no hay coincidencia, revisión manual.
- 1 = hay coincidencia, geolocalización OK.

En primer lugar, se intentó una clasificación a partir del similarity score del fuzzy matching, que resultaba en una escala de 0 a 100. A partir del mismo se armaron umbrales:

- **Inexacta:** similarity score entre 0 y 59.
- **Semi-exacta:** similarity score entre 60 y 89.
- **Exacta:** similarity score entre 90 y 100.

Esta clasificación fue descartada por diversas razones:

1. **No cuantifica de manera correcta algunas diferencias.** Uno de los errores más importantes es que considera como parecidas direcciones como “Calle 23 ...” y “Calle 37 ...”, por contener “Calle” y “3”. En realidad, estas direcciones son distintas y es un grave error si se llega a geolocalizar de esta manera.
2. **No tiene en cuenta el etiquetado de los modelos de NLP.** El similarity score compara el texto de la dirección completa. Si se toma esta medida se está dejando de lado el trabajo realizado con las etiquetas.
3. **No reduce la carga de revisión manual de los analistas,** ya que, tanto las semi-exactas como las inexactas precisan revisión manual.

Finalmente, se definió una categorización que se centra en eliminar lo más posible la revisión manual de direcciones. Esto se logra descartando casos donde la coincidencia es muy buena o muy mala. En los casos intermedios se automatiza la búsqueda de coincidencias con recursos externos como APIs. Las categorías resultantes son:

- **Exacta:** se encuentra coincidencia, geolocalización OK (no se revisa manualmente).
 - a. Exacta: constituye a las direcciones que se encuentran con coincidencia exacta dentro de la base de OCASA.
 - b. Exacta NORMA: integra la comparación de etiquetas nombreCalle y numeroCalle, con límites de diferencia de altura menor a 300.
 - c. Exacta API: forma una nueva dirección a partir de las etiquetas y busca la geolocalización en la API de Google Maps.
- **Geointeractiva** (no se revisa manualmente): estas direcciones contienen falta de información importante, como por ejemplo, el nombre de la calle. Aquí, la

dirección es derivada a otro sector de OCASA, donde se encargan de contactar al cliente para recolectar estos datos. Además incluye las direcciones que tienen una diferencia mayor a 300 metros en la altura.

- **Revisión Manual:** estas son las direcciones que no cumplieron las condiciones para ser geolocalizadas automáticamente. Esto puede darse porque:
 - a. La dirección del cliente no tiene las etiquetas suficientes para ser geolocalizada.
 - b. La dirección encontrada por la API no pertenece al mismo código postal/localidad o no tiene el formato nombre de calle y número. Un ejemplo de este último caso es que la API devuelva el nombre de una ciudad o pueblo, en lugar de una dirección concreta.

Desarrollo de la solución

El desarrollo de la solución comprende tres aristas:

- Selección de un modelo de NLP.
- Diseño del proceso de geolocalización.
- Desarrollo de una interfaz de Usuario.

Modelos de NLP

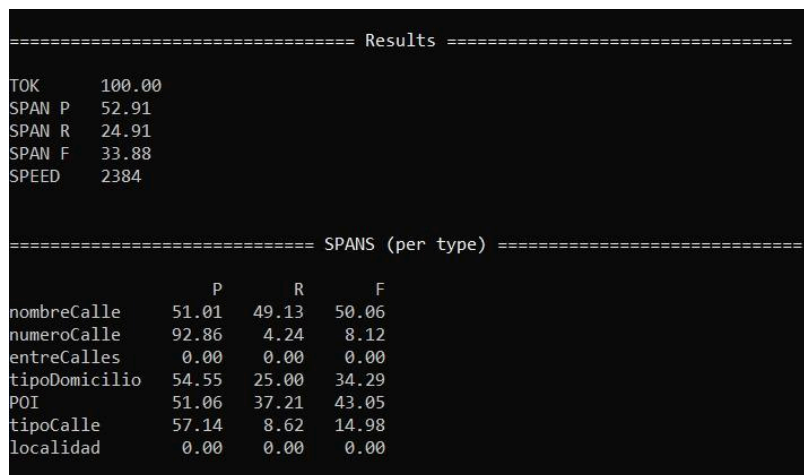
La implementación del modelo apunta a mejorar la precisión con la que se pueden geolocalizar las direcciones aproximadas, aquellas categorizadas como exactas no tienen necesidad de ser comparadas por etiquetas.

Para reducir la intervención humana, se necesita automatizar el proceso y así mejorar la calidad de los datos. Al ser campos de texto abierto, las direcciones llegan en muchas ocasiones con mal formato, o con información extra. Sin embargo, esto no implica que la información proporcionada sea irrelevante.

Además de mejorar la calidad de los datos, el proyecto tiene como objetivo mejorar el tiempo del proceso. Se reemplazan las macros de excel por código en Python, el cual es considerado más rápido. Los archivos procesados en estos códigos son los mismos inputs que recibirá el programa de Geolocalización.

Entrenamiento Modelo Spancat

El modelo de *spancats* calcula las tres métricas principales para evaluar modelos de clasificación: *precision*, *recall* y *F1*. Estos cálculos se hacen tanto para el modelo en general como para cada etiqueta (*span*). El entrenamiento del modelo *spancats* dió los siguientes resultados:



```
===== Results =====
TOK      100.00
SPAN P    52.91
SPAN R    24.91
SPAN F    33.88
SPEED    2384

===== SPANS (per type) =====
          P      R      F
nombreCalle  51.01  49.13  50.06
numeroCalle  92.86   4.24   8.12
entreCalles   0.00   0.00   0.00
tipoDomicilio 54.55  25.00  34.29
POI           51.06  37.21  43.05
tipoCalle     57.14   8.62  14.98
localidad     0.00   0.00   0.00
```

Figura 23 - Métricas del modelo *spancats*

El *recall* del modelo es 24,91%, un valor insuficiente para un modelo de clasificación. A su vez, la precisión del modelo tiene un valor de 52,19%. El F1 es de 33,88%, equilibrando los dos *scores* anteriores.

Las mismas métricas para cada etiqueta dan una mayor comprensión del funcionamiento del modelo:

- **nombreCalle:** esta es la etiqueta de éxito de este modelo, tiene un valor relativamente alto tanto de *recall* como precisión. Esto indica que predice los nombres de calle con moderada exactitud. Sin embargo, existe un porcentaje de falsos positivos y negativos que afectan esta métrica. Este éxito puede deberse a la gran cantidad de ejemplos de esta etiqueta en la muestra, ya que el nombre de la calle está en todas las direcciones, ya sea en número o palabras.
- **numeroCalle:** esta etiqueta no está adecuadamente clasificada por el modelo. El *recall* es de 4,24%, lo que significa que predice un exceso de falsos negativos. La precisión es bastante alta, 92,86%, lo que demuestra una poca presencia de falsos positivos. Un ejemplo de esto es: etiquetar cualquier número de una dirección como *numeroCalle* cuando en realidad son éstos pueden ser *nombreCalle* como es el caso de la ciudad de La Plata.

- **entreCalles**: esta etiqueta no es reconocida por el modelo, esto puede deberse a fallas en el etiquetado o en el formateo de datos y deberá ser corregido con urgencia.
- **tipoDomicilio**: esta etiqueta tiene un *recall* de 25% el cual es un valor bastante bajo. En comparación con otras etiquetas, se podría decir que el modelo “entiende” los tipos de domicilio posibles.
- **POI**: esta etiqueta se asemeja a tipoDomicilio pero, con una leve mejora en el *recall*.
- **tipoCalle**: la etiqueta tipoCalle tiene un valor muy bajo de *recall* 8,62% indicando que el modelo predice una gran cantidad de falsos negativos. Es decir, una dirección posee tipos de calle (AV, AV., CALLE, AVENIDA, BIS, PASAJE, PJE) pero el modelo no los identifica. Adicionalmente, la precisión no es muy elevada (57,14%) por lo cual la etiqueta en sí es defectuosa. Esto puede deberse a diferencias en el etiquetado manual.
- **localidad**: esta etiqueta no pudo ser reconocida por el modelo, esto es probablemente por los pocos casos que la contienen, como se muestra en la *figura 19*.

Entrenamiento Modelo NER

El modelo NER fue entrenado utilizando diferentes ajustes en las métricas para ver cómo era el rendimiento del mismo. En primer lugar, se le dió mayor importancia al *recall*. Se obtuvieron los siguientes resultados:

Results			
TOK	100.00		
NER P	41.04		
NER R	27.98		
NER F	33.27		
SPEED	2721		
NER (per type)			
	P	R	F
numeroCalle	44.59	62.91	52.19
nombreCalle	25.00	0.34	0.67
tipoCalle	25.00	4.00	6.90
POI	50.00	6.45	11.43
tipoDomicilio	10.71	25.00	15.00
entreCalles	29.03	64.29	40.00
localidad	0.00	0.00	0.00

Figura 24 - Métricas del modelo NER, optimizado para *recall*

El *recall* general de este modelo es de 3 pp. superior al del modelo *spancats*. Sin embargo, al analizar cada etiqueta por separado, se observan cambios en el rendimiento del modelo:

- **numeroCalle:** a diferencia del modelo anterior, esta etiqueta es la mejor predicha por el NER, tiene un nivel aceptable de *recall* (62,91%), acompañado de una precisión equilibrada (44,59%).
- **nombreCalle:** en este caso, el modelo no puede identificar nombres de calles, a diferencia del modelo *spancats*. Las tres métricas tienen valores bajos para esta etiqueta.
- **tipoCalle:** esta etiqueta no clasifica los tipos de calle correctamente al igual que el modelo *spancats*.
- **POI:** es similar al modelo anterior, pero con menor *recall*.
- **tipoDomicilio:** para esta etiqueta el *recall* (25%) es más alto que la precisión (10,71%). Si bien esto es deseado, los valores numéricos son bajos.
- **entreCalles:** el caso de éxito de este modelo es la clasificación de la etiqueta entreCalles, no solo la identifica sino que con un *recall* del 64,29%, lo cual es un valor destacable.
- **localidad:** esta etiqueta no pudo ser reconocida por el modelo, al igual que en el caso anterior.

Estos resultados sugieren que tanto el modelo *spancat* como el NER tienen dificultades para clasificar correctamente las diferentes etiquetas en las direcciones.

Por esta razón, se investigaron posibles problemas de configuración del modelo o conversión de los datos. El paso en el cual se intuye que existen problemas es cuando el motor de SpaCy lee los datos provenientes de LabelStudio. Así fue como se optó por probar un output distinto a JSON, que fuera compatible tanto con LabelStudio como en SpaCy.

Finalmente, se usó un archivo de tipo CoNLL2003 para el proceso *convert*. Este resolvió el problema y dió resultados muy favorables para el modelo NER en particular. Todas las métricas generales del modelo superan los 80 puntos. Si se desglosa por etiqueta se puede ver que:

- La etiqueta nombreCalle tiene un nivel alto tanto de *precision* (77,58%) como de *recall* (81,79%).
- La etiqueta numeroCalle es la que mejor identifica este modelo con todas las métricas sobre 90 puntos.
- tipoCalle y tipoDomicilio mejoraron significativamente en comparación con el modelo anterior.

- Otra gran diferencia de este modelo es la identificación de la etiqueta entreCalles, la cual no fue identificada por el spancat anterior. Además con muy buenas métricas de *precision* (69,44%) y *recall* (69,44%).
- La etiqueta POI también es clasificada muy bien por el modelo, alcanzando el puntaje máximo de *precision* (100%) y uno moderado en *recall* (57,14%).

```

===== Results =====
TOK      -
NER P    81.33
NER R    84.81
NER F    83.03
SPEED    5796

===== NER (per type) =====

```

	P	R	F
nombreCalle	77.58	81.79	79.63
numeroCalle	92.22	96.34	94.23
tipoCalle	37.50	42.86	40.00
localidad	0.00	0.00	0.00
POI	100.00	57.14	72.73
entreCalles	69.44	69.44	69.44
tipoDomicilio	50.00	28.57	36.36

Figura 25 - Métricas del modelo NER mejorado

Dado su buen nivel de performance, este es el modelo que se implementará en el entregable del proyecto.

Geolocalización

A continuación se define el proceso de automatización de geolocalización de direcciones. Este proceso contempla cinco outputs, de los cuales solo dos requieren intervención humana:

- **EXACTA:** coincidencia 100%, no requiere intervención.
- **EXACTA NORMA:** coincidencia a través de etiquetas, no requiere intervención.
- **EXACTA API:** construcción de una dirección a partir de etiquetas y geolocalización con recurso externo (API Google Maps), no requiere intervención.
- **GEOINTERACTIVA:** dirección con falta de información o que incumple los requerimientos de OCASA (altura menor 300 metros), requiere intervención humana por parte de *customer experience*.

- **REVISIÓN:** dirección que no obtuvo las etiquetas suficientes para ser geolocalizada o que el resultado de API no es satisfactorio, requiere intervención humana por parte de *analistas*.

El proceso por el cual se decide cómo asignar una latitud y longitud a una dirección es el siguiente:

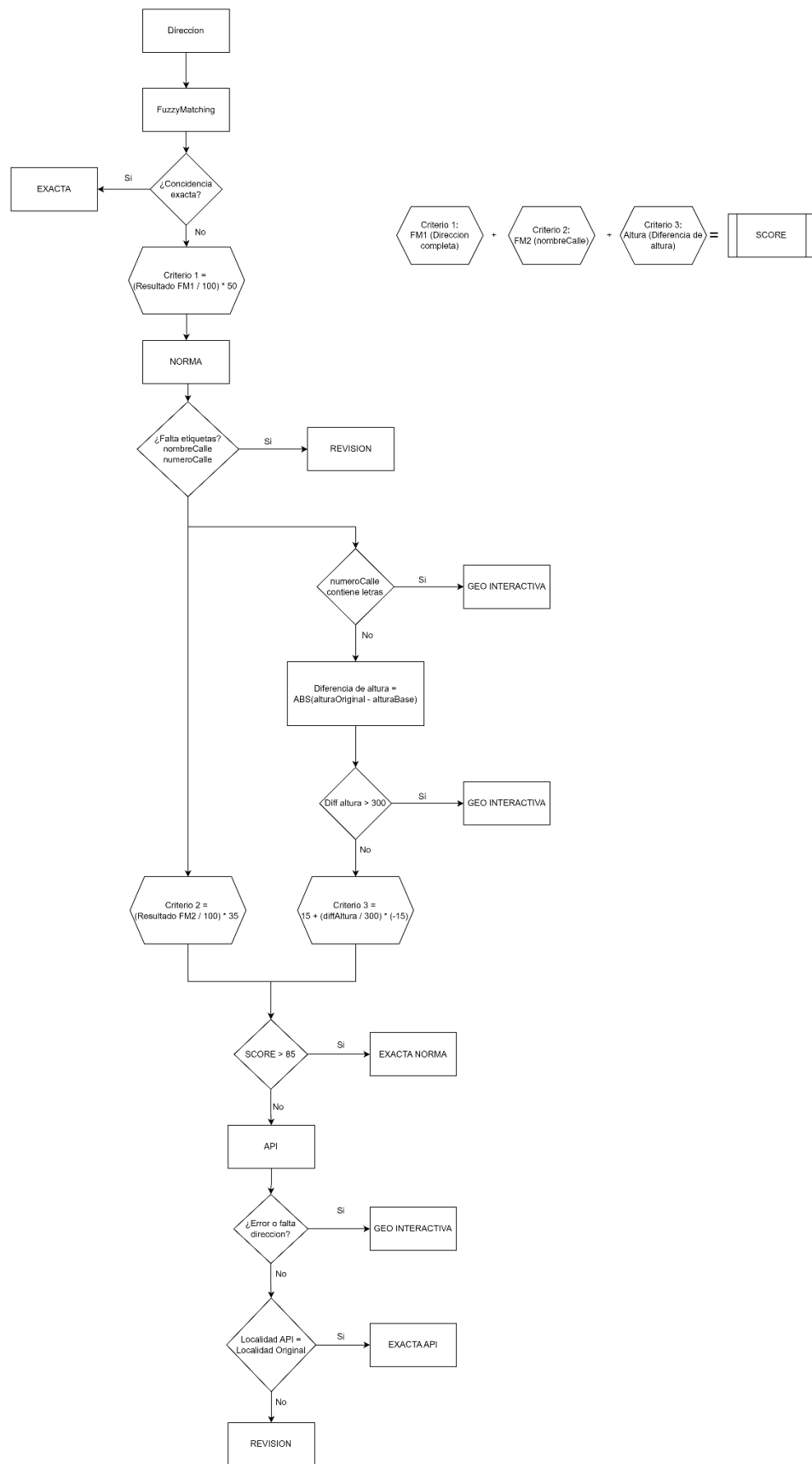


Figura 26 - Proceso de geolocalización.

1. Se estandariza el formato de las direcciones. Todas las direcciones se pasan a mayúscula y se eliminan las tildes.
2. Todas las direcciones son etiquetadas por NORMA (este es el nombre del modelo NLP elegido) y se obtienen etiquetas.
3. Se filtran las direcciones con el mismo código postal en la base de direcciones y se agregan a una lista.
4. Se busca la mejor coincidencia entre la dirección a geolocalizar y la lista de direcciones del mismo código postal con fuzzy matching (FM). El resultado de este matcheo (valor numérico entre 0 y 100) define el valor del criterio 1 (CR1) de la siguiente forma: $CR1 = \frac{\text{resultado}}{100} \times 50$.
5. Si se encuentra una coincidencia 1:1 en la lista o si el valor del resultado del primer FM es 100 se etiqueta como **EXACTA**, asignando latitud y longitud de la dirección de la base.
6. Las que no cumplieron con esta coincidencia, se procesan con NORMA. Se etiqueta la dirección encontrada por el primer FM y se comparan las etiquetas de *nombreCalle* y *numeroCalle* de la dirección que se está intentando geolocalizar.
 - a. Para los *nombreCalle* se realiza un segundo FM en el cual se ve qué tan similar es la calle que el modelo encontró en ambas direcciones. Con esto asigna valor al criterio 2 (CR2), usando la misma fórmula que para CR1 pero con un valor tope de 35 en vez de 50.
 - b. Mientras que para *numeroCalle* lo que se calcula es la diferencia entre las alturas, con una diferencia máxima aceptada de 300 para ser considerada EXACTA NORMA o EXACTA API. La diferencia da valor al criterio 3 (CR3) de la siguiente manera: $CR3 = 15 + \frac{\text{diffAltura}}{300} * (-15)$
7. En caso de que NORMA no identifique alguna de las etiquetas en alguna de las direcciones, el caso se clasifica como **REVISIÓN** ya que no podrá ser evaluada con esta metodología.
8. Se calcula score a partir de los tres criterios: $SCORE = CR1 + CR2 + CR3$
9. Si SCORE es mayor a 85 y la diferencia de altura es menor a 300 (nótese que para lograr este puntaje es necesario al menos obtener 1 punto de CR3) la dirección se considera como **EXACTA NORMA** y se le asigna latitud y longitud de la dirección encontrada.

10. Si SCORE es mayor a 85 y la diferencia de altura es mayor a 300, por requerimiento de OCASA, la dirección es clasificada como **GEOINTERACTIVA**.
11. Para el resto de los casos, se procede a realizar una búsqueda por API. Dado que muchas veces las direcciones traen información de más como "entre calle", "esquina", "y" que confunde a los motores de búsqueda. Es por eso que se arma una dirección a partir de etiquetas y esta se busca en la API. El formato de armado es el siguiente: *nombreCalle, numeroCalle, localidad, provincia*
12. Si la dirección encontrada por la API, tiene la misma localidad que informa el cliente y trae una calle con altura, la dirección se considera como **EXACTA API**. Si la dirección trae calle pero no coinciden las localidades, va a **REVISIÓN**. Si no trae una dirección, es clasificada como **GEOINTERACTIVA**.

Interfaz de Usuario

Se desarrolló una interfaz de usuario, GeoCenter, para facilitar y simplificar la ejecución de los programas a usuarios sin conocimientos previos de programación. Esta interfaz centraliza y simplifica los procesos de geolocalización.

El resultado final para las direcciones procesadas en esta herramienta va al motor de ruteo Unigis, parte de la operación de OCASA. Si la precisión de la ruta es buena, se necesita de menor cantidad de correcciones reduciendo nuevamente el trabajo de los analistas.

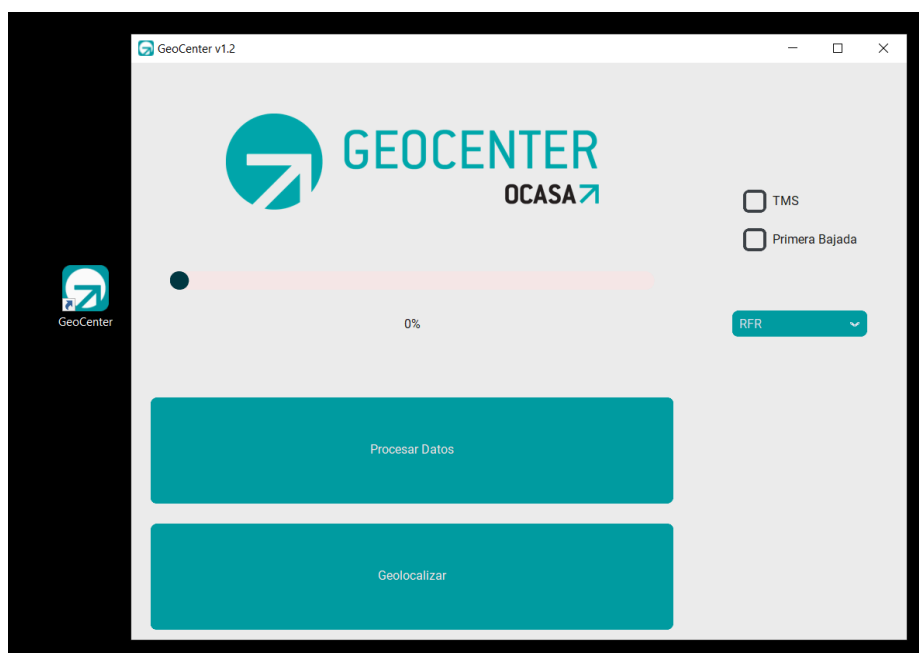


Figura 27 - Interfaz de usuario de GeoCenter.

Resultados

En este apartado se presentan los resultados obtenidos del proceso de geolocalización de direcciones, así como la validación del mismo. La transparencia y precisión de estos resultados son fundamentales para garantizar la confiabilidad del sistema y proporcionar una base sólida para futuras mejoras.

Proceso

El sistema de automatización de geolocalización de direcciones ha sido evaluado utilizando un conjunto de datos de un día en la operación normal de OCASA. El conjunto de prueba está compuesto por aproximadamente 55.000 direcciones que se clasificaron usando el sistema. A continuación, se presenta una tabla a modo de resumen con los resultados obtenidos:

Clasificación	Cantidad	Porcentaje
EXACTA	34.851	64%
EXACTA NORMA	6.642	12%
EXACTA API	1.019	2%
GEOINTERACTIVA	4.386	8%
REVISIÓN	7.290	13%
Total	54.189	

Tabla 6 - Clasificaciones del proceso.

En primer lugar, el porcentaje de direcciones con coincidencia **EXACTA** es del 64%. Esto es igual a el porcentaje de coincidencia que se encontraban con el proceso anterior de OCASA. La empresa informó que el único método de coincidencia era la exacta, hecha con un buscarV de Excel, lo que resultaba en un 35% de revisión manual. Por lo tanto, el porcentaje de coincidencias con ese sistema era del 65%. Aquí se ven dos aspectos positivos:

1. La información proporcionada sobre el proceso actual de OCASA es consistente con los datos provistos.
2. El nivel de conciencia exacta se mantiene al migrar la comparación de Excel a Python.

En segundo lugar, se analiza el porcentaje de direcciones geolocalizadas como **EXACTA NORMA**. Este porcentaje demuestra la performance del modelo NER. El porcentaje de direcciones que se geolocalizan usando el modelo es del 12%.

Esto es un gran logro del proyecto, debido a que se utilizó un modelo de NLP que no fue originalmente creado para la geolocalización de direcciones y este etiquetó de manera confiable más del 10% de las direcciones. Una de las hipótesis de este proyecto es comprobar si un modelo de este tipo puede ser útil para automatizar el proceso y se ve que efectivamente es así.

En tercer lugar, las direcciones geolocalizadas como **EXACTA API** son el 2%. Este es un uso alternativo del modelo, ya que la dirección que se busca en la API de Google Maps es armada con las etiquetas extraídas del modelo. Por lo tanto, se puede decir que el modelo ayuda a geolocalizar el 14% de las direcciones.

En cuarto lugar se observan las direcciones clasificadas como **GEOINTERACTIVA**, estas son las que los analistas redirigen al área de *customer experience* porque no tienen la suficiente información como para ser geolocalizadas. Por ejemplo, falta el nombre de la calle. Esto se da el 8% de las veces.

Finalmente, se encuentra la categoría **REVISIÓN**, el porcentaje de direcciones de la misma es 13%. Aquí se presentan las direcciones que necesitan revisión manual por parte de los analistas. Estas fueron etiquetadas por el modelo NER y obtuvieron un score medianamente alto. Es decir, aparentemente, contienen toda la información necesaria para ser geolocalizadas (este no es el caso de las direcciones categorizadas como **GEOINTERACTIVA**). Las direcciones son categorizadas como **REVISIÓN** cuando no se encontró coincidencia con las direcciones del base de OCASA. Probablemente sea una dirección nueva a la que OCASA no ha entregado nunca un paquete. Esta categoría evidencia la reducción del trabajo manual que provee el sistema propuesto en este proyecto.

Es importante aclarar que entre la operación actual de OCASA y la del proyecto propuesto hay una diferencia en lo que se considera como geolocalización automática y revisión manual. Esto se da al agregar categorías a ambas clases.

Validación

Con el objetivo de validar la clasificación hecha por proceso, se ha seleccionado una muestra estadísticamente significativa de 2.670 direcciones, a las cuales se le verificó la categoría asignada. Este tamaño de muestra asegura los resultados con un nivel de confianza del 99% y un margen de error de 2,5%.

La verificación se hizo manualmente por los miembros del equipo. Este proceso de validación permite calcular un margen de error y establecer un nivel de confianza para el proceso propuesto en este proyecto.

Se verificaron 534 direcciones aleatorias de cada categoría. La cantidad y el porcentaje de clasificación correctas se muestran en el siguiente cuadro:

	CORRECTAS TOTALES		
EXACTA	534	534	100.00%
EXACTA NORMA	533	534	99.81%
EXACTA API	481	534	90.07%
GEO INTERACTIVA [RUTA 501]	417	534	78.09%
REVISION	524	534	98.13%
TOTAL	2489	2670	93.22%

Figura 28 - Validación del proceso.

La exactitud general del proceso es del 93,22%, un valor altamente deseable. Los procesos más confiables son: EXACTA, EXACTA NORMA y REVISIÓN. A su vez estos son los procesos que abarcan la mayor cantidad direcciones en la muestra de direcciones de un día de OCASA (Tabla 6). Acumulan el 90% de las mismas.

Los procesos que presentan un nivel moderado de exactitud son: EXACTA API y GEOINTERACTIVA. Sin embargo, solo el 10% de las direcciones pertenecen a estas categorías (Tabla 6).

En resumen, el proceso propuesto en este proyecto tiene una exactitud entre un 90,72% y un 95,72% con un nivel de confianza del 99%. Si se implementa este proceso puede asegurar un 90% de precisión.

Conclusiones

El objetivo de este proyecto es reducir la sobrecarga de revisión manual a la hora de geolocalizar direcciones. Se planteó la hipótesis de que esto se podía lograr automatizando el proceso e incluyendo un modelo de Natural Language Processing (NLP). Para medir el cumplimiento de este objetivo se establecieron los siguientes KPIs

- Porcentaje de revisión manual.
- Porcentaje de geolocalizaciones automáticas.
- Costo de revisión manual.

- Costo por envío.

A lo largo de este proyecto se experimentó con distintos modelos de NLP, centrándose en los del tipo NER o afines. A su vez, se adaptó la coincidencia exacta que ya era parte de la operación existente en OCASA y se agregó un recurso externo para poder abarcar la mayor cantidad de casos posibles (API de Google Maps).

Todo esto resultó en un proceso que no solo geolocaliza direcciones, sino que también redirige las que no logra geolocalizar a las áreas adecuadas según su calidad. Es decir, el proceso realiza un doble trabajo: geolocalización y clasificación.

- La clasificación es realizada con un 90% - 95% de exactitud.
- La geolocalización la realiza con un 96.63% de exactitud. En particular:
 - La geolocalización por coincidencia exacta tiene un 100% de exactitud.
 - La geolocalización por modelo NER tiene un 98,81% de exactitud.
 - La geolocalización con modelo NER más API de Google Maps tiene un 90,07% de exactitud.

El impacto de este proceso en los KPIs establecidos es el siguiente. En primer lugar, la revisión manual (RM) se reduce de un 35% a un 13%, superando el escenario moderado planteado (15%) por 2 puntos porcentuales. El costo de revisión manual es de \$182,47 , alcanzando un ahorro del 37%. A su vez, los costos de envío diario son de \$572, alcanzando el mismo nivel de ahorro.

El porcentaje de geolocalización automática aumenta 14 puntos porcentuales de 65% a 79%. Finalmente, el porcentaje de envíos no entregados por mala geolocalización es estimadamente del 1,5%, esto representa una reducción de 2,5 puntos porcentuales.

Escenarios	RM	% Geolocalización automática	Costo RM	Costo de envío diario	% mala geo
Actual	35%	65%	\$491,26	\$1.540,00	4,0%
Moderado	15%	85%	\$350,90	\$660,00	1,7%
Logrado	13%	79%	\$182,47	\$572,00	1,5%

Tabla 7 - Resultados logrados por el proceso propuesto.

Como conclusión de este análisis, se evidencian los beneficios sustanciales que tendría la implementación del sistema propuesto en este proyecto para OCASA.

Potenciales Próximos Pasos

Como potenciales próximos pasos, se recomienda guardar las ubicaciones geolocalizadas en la base de direcciones mediante API que han sido entregadas correctamente para mejorar la precisión del proceso en el futuro. Lo que permitirá aumentar la cantidad de registros disponibles para comparaciones posteriores, incrementando el número de resultados encontrados de forma EXACTA y EXACTA NORMA. Esta medida disminuye los costos al reducir la necesidad de realizar llamados adicionales a la API para geolocalizar direcciones.

Del mismo modo pueden generarse reportes de uso de la aplicación al subir información sobre los procesos ejecutados, al igual que los tiempos de ejecución, la cantidad de direcciones procesadas, para tener mejores insights acerca de la calidad de direcciones con las que OCASA geolocaliza y su evolución en el tiempo con la implementación del proyecto.

Anexo

[Presentación](#)

Bibliografia

- Mattingly, William. *Introduction to Named Entity Recognition*, 2021 (2nd ed.). ner.pythonhumanities.com
- Nishanth, N. *Training Custom NER*, 2020. towardsdatascience.com
- Singh, Taranjeet. *Natural Language Processing With spaCy in Python*, 2023. realpython.com