

CHATBOT PARA A DOCUMENTACIÓN E NORMATIVA DA UDC

Marcelo Ferreiro Sánchez
Marcos Grobas Martínez
José Romero Conde

14 de decembro do 2025

Resumo

O obxectivo do proxecto é desenvolver un chatbot capaz de solventar dúbidas acerca do funcionamento dos procesos burocráticos e de documentación da Universidade da Coruña (UDC). Para conseguilo optamos por unha arquitectura xenerativa aumentada por recuperación (RAG), técnica moi empleada para mellorar o desempeño de chatbots baseados en LLM cando buscan información específica dun dominio sobre o que o modelo de linguaxe orixinal non foi entrenado.

Índice xeral

1. Introducción	2
2. Solución proposta	3
2.1. A Arquitectura	3
2.2. O <i>Crawler</i>	3
2.2.1. Selección de páxinas relevantes	3
2.2.2. Procesado dos contidos de cada páxina	4
2.2.3. Funcionamento interno do <i>Crawler</i>	6
2.3. O <i>RAGsystem</i>	7
2.3.1. Construcción da base de vectores	7
2.3.2. O sistema de recuperación	7
2.4. Os modelos	8
2.5. Ferramentas usadas	8
3. Instalación e uso	9
4. Resultados	10
5. Conclusions	11
A. Lista completa de <i>keywords</i>	13
B. Saída OCR da táboa de exemplo	15

1. Introducción



burocracia de calquera campo pode chegar a ser moi complexa e pode chegar a consumir moito tempo e recursos ás persoas que teñen que lidiar con ela. Os procesos universitarios non son unha excepción e moitas das persoas involucradas neles (sexan, alumnos, profesores ou persoal administrativo) os poden atopar inabarcables ou imposibles de navegar sen axuda.

Neste contexto, apreciouse como podería ser de gran utilidade o desenvolvemento dun chatbot capaz de responder a preguntas relacionadas coa documentación e normativa da Universidade seguindo a gran tendencia da actualidade de empregar chatbots para diversas tarefas.

O obxectivo deste proxecto é desenvolver un chatbot que poida simplificar e explicar **referindo sempre ás fontes burocráticas oficiais** os procesos burocráticos e de documentación da Universidade da Coruña (UDC).

O sistema é resultado da unión de compoñentes e técnicas xa ben coñecidas, sempre tendo en mente o obxectivo a cumprir. Polo tanto, tratouse máis ben dunha tarefa de aplicación e adaptación de técnicas e ferramentas xa existentes nun caso particular, antes que de deseño ou resolución de novos problemas.

Se ben este traballo está circunscrito no contexto da asignatura de Técnicas Avanzadas de Procesamiento de Linguaxe Natural (TAPLN), debido á súa natureza e obxectivo, gran parte do tempo invertido no proxecto dedicouse á tarefa de recolección de información para o RAG.

Non existindo unha 'base de datos' oficial da UDC sobre a que un usuario poda descargar toda a documentación relativa ao centro, senón que atópase esta repartida nas páxinas web dos seus diferentes centros, foi necesario deseñar unha solución que nos permita extraela a partir dos seus portais oficiais.

Este tipo de tarefas non son novas no mundo da informática, de feito pertencen a unha área máis que consolidada chamada Recuperación de Información (IR) e tales métodos que buscan, extraen e organizan información disposta en webs html son coñecidos como *crawlers*, parte esencial do traballo final pois para asegurar que o sistema sexa capaz de responder á maioría de dúbidas dos usuarios, é necesario ter un corpus de toda información mínimamente relevante acerca do funcionamento interno da universidade.

2. Solución proposta



ostrárase nesta sección a arquitectura xeral do sistema e posteriormente describirase cada un dos seus compoñentes, sendo estes principalmente o *RAGsystem* e o *Crawler*.

2.1 A Arquitectura

Fundamentalmente e como indicouse antes, o sistema componse de dúas partes relativamente independentes: *RAGsystem*, que será un modelo de linguaxe grande (LLM) xa adestrado que recibirá xunto con cada consulta de usuario, unha serie de documentos (ou fragmentos dos mesmos) para responder mellor á mesma, e por outra parte, un sistema de *crawling* que ocuparase de navegar as diversas páxinas e portais da UDC recolectando aquelas que considere que poden conter información relevante.

Nas seguintes subseccións explicaranse en detalle o funcionamento interno de cada parte e cómo interactúan entre elas. Optouse por unha orde de exposición que siga o fluxo de execución do sistema, é dicir, comezando polo *Crawler* e seguindo polo *RAGsystem*. Sopesouse seguir a orde cronolóxica do desenvolvemento do sistema mais concluíuse que introduciría demasiadas exégesis e reviraría demasiado e sen utilidade a narrativa desta memoria. De calquera forma, explicarase a evolución das partes que máis cambios sufriron ao longo da súa implementación.

2.2 O *Crawler*

Considerouse que a posesión dun bo volume de datos sería crucial para resolver o problema a tratar, na maioría dos casos as dúbidas burocráticas resólvense encontrando o documento adecuado, e de non telo, o sistema veríase obrigado a comunicarlle ao usuario que non ten acceso á tal ou cal información, voltándose completamente inútil. É por iso que a Recuperación de Información xoga un papel crucial para o RAG.

2.2.1 Selección de páxinas relevantes

Sendo así, o primeiro problema a solucionar é o de deseñar un bo criterio do que é un **documento relevante** para a aplicación. Se ben tal tarefa pódese complicar *ad infinitum* e é un área de estudo activa aínda nestes días [Pezzuti, MacAvaney e Tonellotto 2025], neste caso optouse por un enfoque moito máis sinxelo, baseándose primariamente na premisa de que esta tarefa non require de conxuntos masivos de datos. Só é necesario navegar un certo número de URLs (principalmente do directorio raíz de cada facultade),

non a totalidade da web, polo que pódese permitir descargar páxinas e documentos potencialmente pouco relevantes. A información a colleitar ten un límite de tamaño finito e razoable: nun escenario extremo, mesmo descargando todas as URLs da UDC, estaríase lonxe de requirir terabytes de almacenamento, como sí acontecería nun *crawler* de propósito xeral.

Deste xeito, a solución proposta sitúase como un método arcaico aínda que funcional, unha asignación de relevancia binaria (relevante ou non relevante) mediante *keywords*. Este enfoque, se ben simple, remite aos métodos de *focused crawling* pioneiros [Chakrabarti, Berg e Dom 1999] que sentaron as bases da recuperación de información temática na web.

As palabras clave utilizadas son fixas e propias do ámbito académico e do vocabulario burocrático, aparecendo tanto en galego, coma en castelán e inglés (véxase o Apéndice A para a lista completa).

2.2.2 Procesado dos contidos de cada páxina

Unha vez unha páxina estímase como relevante, é necesario procesar seu contido, para este caso, procurar presentalo en texto plano, evitando ao máximo posible decoradores. Mais nunha páxina atópase máis que texto, especialmente neste caso de uso, moita información burocrática útil para un usuario final atoparase nun documento PDF ligado ao URL, é por iso que o *crawler* tamén os procesa e converte en texto plano listo para que posteriormente un LLM poda leelos sen maior complicación.

Non obstante, este enfoque presenta unha limitación importante: información que se atope unicamente en imaxes (capturas de pantalla, carteis dixitalizados ou documentos escaneados) permanece invisible para o *crawler*.

Tal debilidade do sistema fíxose obvia nas primeiras fases de testeo do *chatbot*, pois este non era capaz de responder a unha pregunta moi sinxela e básica para os estudantes: "Cantos libros pódense emprestar?". Tras investigar a casuística e comprobarse que o URL onde figura tal información foi efectivamente analizada polo *crawler*, caéuse na conta de que, se ben a información figura nunha táboa dentro da páxina, esta está en formato imaxe [ver 2.1], polo que próbase que é necesario aproveitar mellor a información de cada páxina para cumprir os obxectivo establecido.

Tipos de usuarios	Nº de documentos en préstamo	Días de préstamo	Renovaciones	Reservas
GRUPO 1				
Estudiantado de Grado de centros propios y adscritos	10	10 días	Indefinidas	Límite 10 docs.
Estudiantado de programa de movilidad (Erasmus, Sicue-Séneca)				
Estudiantado de doble Grado, de simultaneidad				
Estudiantado de grados interuniversitarios				
Estudiantado da Universidade Sénior				
GRUPO 2				
Estudiantado de MASTERES e posgrados propios, incluyendo los de la Fundación Universidade da Coruña	15	21 días	Indefinidas	Límite 15 docs.
Estudiantado de Trabajos de fin de grado				
Personal de administración en servicio (PTGAS)				
GRUPO 3				
Estudiantado de Doctorado	35	30 días	Indefinidas	Límite 35 docs.
Personal investigador visitante: visitante senior/ visitante predoctoral o postdoctoral	35	30 días	Indefinidas	Límite 35 docs.
GRUPO 4				
Becarios/as de investigación	35	Curso académico (cada biblioteca podrá excluir de este tipo de préstamo documentos por razón de uso y disponibilidad)	Indefinidas	Límite 35 docs.
Personal contratado investigador				
GRUPO 5				
PDI da UDC (incluyendo centros adscritos), de la Fundación UDC	100	Curso académico (cada biblioteca podrá excluir de este tipo de préstamo documentos por razón de uso y disponibilidad)	Indefinidas	Límite 100 docs.
Profesorado emérito, jubilado incentivado y honorario				
Profesorado visitante				
Lectores/as				
GRUPO 6				
Cualquier persona ajena a la comunidad universitaria de la UDC que sea autorizada por la Biblioteca Universitaria	6	10 días	Indefinidas	Límite 6 docs.

Figura 2.1 Táboa de préstamos

Procesado de imaxes mediante OCR

O recoñecemento óptico de caracteres (OCR polas súas siglas en inglés) é unha técnica que permite extraer texto a partir de imaxes dixitais, se ben o seu uso máis típico é a dixitalización de documentos físicos escaneados, serve para calquera imaxe que conteña caracteres, como é o caso.

Neste caso, implementouse mediante a librería *pytesseract* unha interface en *Python* do motor OCR Tesseract, creado inicialmente por HP e mantido actualmente por Google.

Procesar cada imaxe presente nun URL pode semexar moi custoso computacionalmente, mais a extracción de texto mediante OCR é verdadeiramente rápida, isto sumado a que a maioría de imaxes son irrelevantes e de moi pequeno tamaño, otorga a capacidade de poder procesar todas as presentes en calquera documento sen apenas engadir latencia ao sistema. Como exemplo, o tempo de procesado de OCR na imaxe da táboa de préstamos [2.1] foi de 0,6280 segundos.

Demostrado que é posible e relativamente sinxelo á vez que barato procesar arquivos de imaxe, é necesario agora demostrar que tal texto é fidedigno ao realmente existente na mesma.

Un exemplo que demostra á perfección un caso de uso real é precisamente a táboa mencionada anteriormente, o texto extraído en crú pódese ver no apéndice B. Revisándoo, o texto extraído non asemexa nada parecido á táboa orixinal, mais hai que ter en mente que non é necesario que sexa intelixible para o usuario final, senón para o LLM que o debe de analizar. Unha forma rápida de comprobar que o chatbot poida ser capaz

de entender o texto extraído (coa complicación de que neste caso está en formato táboa) é pedirlle mediante un prompt no portal online dalgún LLM comercial, que refaga a táboa por nós con esa información. Fíxose tal experimento usando ChatGPT-5 e a táboa que reconstruíu foi a seguinte:

Grupo	Tipo de usuarios	Nº docs	Días	Renov.
1	Estud. Grao (centros propios/adscritos), Grao, Graos interuniv., Univ. Sénior	10	10 días	Indef. (Lím. 10)
2	Estud. másteres/posgraos propios (Fund. UDC), Estud. TFG, PTGAS	15	21 días	Indef. (Lím. 15)
3	Estud. Doutorado, Inv. visitante/senior, Inv. predoc./posdoc	35	30 días	Indef. (Lím. 35)
4	Becarios inv., Pers. contr. inv. (<i>Exclusión posible</i>)	35	Curso acad.	Indef. (Lím. 35)
5	PDI UDC (inc. adscritos), Fund. UDC, Prof. emérito/xub./hon., Prof. visitante, Lectores (<i>Exclusión posible</i>)	100	Curso acad.	Indef. (Lím. 100)
6	Persoal externo á UDC, Pers. autoriz. Biblioteca	6	10 días	Indef. (Lím. 6)

Cadro 2.1 Táboa reconstruída mediante LLM (ChatGPT-5)

Se ben existe certa información que o LLM obviou por ser incompleta, a información importante está presente, demostrando que os LLMs son capaces de reconstruír e interpretar a información extraída mediante OCR, incluso cando esté representada en formatos máis complexos como o é unha táboa.

2.2.3 Funcionamento interno do *Crawler*

A implementación proposta usa da librería *Python BeautifulSoup* para navegar a rede e extraer información das páxinas. O proceso comeza cunha lista de URLs semente que representa os portais principais da universidade (concretamente a *homepage* de cada facultade). Para cada URL, o sistema descarga o contido HTML, extrae todos os enlaces e imaxes presentes na páxina, e identifica documentos PDF que conteñan termos relacionados coa burocracia universitaria segundo a lista de *keywords* antes exposta. Os recursos descargados almacénanse de forma organizada no sistema de ficheiros local, mentres que un sistema de metadatos rexistra información sobre cada recurso para optimizar futuras execucións, concretamente garda o *etag* e a data de última modificación de cada páxina, ademais da data de descarga.

En canto ás imaxes, aplícase OCR a todas as presentes en cada documento e decídese se gardar o texto extraído ou non segundo se presenta máis dun mínimo de caracteres. Todo este proceso realízase ao final de *crawlear* cada páxina, os arquivos de imaxe vanse engadindo a unha pila, e esta é procesada cando xa extraéronse as hiperreferencias a outras páxinas e os documentos PDF.

Para acelerar o proceso de *crawling* aplicáronse técnicas de programación concurrente. É posible establecer un número de traballadores aos cales se lles asignará unha URL

como base desde onde iniciar o crawler, deste xeito, é posible ter múltiples crawlers simultaneamente navegando a rede da universidade. Para non xerar problemas de lectura/escritura nos ficheiros onde se escriben os metadatos ou o rexistro de páxinas visitadas, implementáronse *locks*, deste xeito dous *crawlers* non podrán escribir á vez nun mesmo documento.

2.3 O RAGsystem

O sistema RAG (Retrieval-Augmented Generation) implementado constitúe o núcleo do chatbot, combinando técnicas de recuperación de información con modelos de linguaxe xerativos. O proceso de resposta a unha consulta desenvólvese en varias etapas: primeiro, a pregunta do usuario convértese nun vector mediante embeddings semánticos que capturan o seu significado; a continuación, este vector empregase para buscar no vectorstore os documentos máis relevantes mediante similitude coseno, recuperando os k fragmentos de texto máis próximos semanticamente á consulta. Para mellorar a calidade da recuperación, o sistema incorpora un compoñente de reranking baseado en TF-IDF que permite reordenar os documentos recuperados segundo a súa relevancia léxica, ou ben combinarse de forma híbrida coas puntuacións vectoriais mediante pesos configurables. Unha vez identificados os fragmentos máis relevantes, estes incorpóranse como contexto nunha plantilla de prompt xunto co historial recente da conversación, permitindo que o modelo de linguaxe (Claude, ChatGPT ou calquera compatible con *Langchain*) xere unha resposta fundamentada na documentación oficial da universidade. O sistema mantén un historial de conversación de lonxitude configurable que permite responder a preguntas que fan referencia a interaccións previas, dotando ao chatbot de capacidade contextual e conversacional. Ademais, cada documento recuperado enriquecese con metadatos que inclúen a súa puntuación de relevancia tanto vectorial como TF-IDF, facilitando a trazabilidade e verificación das fontes empregadas para xerar cada resposta.

2.3.1 Construcción da base de vectores

2.3.2 O sistema de recuperación

O sistema polo que o RAG recibe os documentos relevantes para cada consulta recibiu múltiples iteracións e cambios tras varias fases de testeo e replanteamento teórico. Trátase dun compoñente clave do sistema e non ten unha solución trivial, pois é un área de investigación moi activa [Author e Author 2025], poderíase dicir que se ben a área de IR parecía estar algo estancada nos últimos anos, debido aos bos resultados que alcanzaron os buscadores web, agora volve a verse moi activa grazas ao xurdimento dos RAGsystems.

Comezou empregando un sistema de recuperación baseado exclusivamente en similitude coseno de vectores FAISS. Mais tras comprobar que tal *approach* podía non

dar os resultados esperados, pois da a mesma importancia a todas as palabras da consulta, non ten en conta cal pode ser a palabra clave máis relevante. A maioría das consultas (especialmente burocráticas) van ter moitas palabras moi comúns e repetidas ao longo da colección de documentos (proceso, documento, normativa) que poden facer que o sistema recupere documentos pouco relevantes.

A primeira solución implementada foi aplicar MMR (Maximal Marginal Relevance) no canto da similaridade coseno. O fundamento detrás desta idea era que deste xeito non desenvolveríanse documentos moi redundantes entre sí, xa que no caso de existir varios documentos moi relevantes entre sí, súa información útil para a consulta sería menor, mais durante a experimentación comprobouse que na realidade non se cumpre tal suposición.

É moi común ver diferentes documentos oficiais que repiten a mesma información ou varias versións e revisións dun proceso en diferentes documentos. En tal caso é probable que o sistema devolva un só deles, perdéndose información relevante ou directamente indicando un proceso obsoleto, xa que prioriza diversidade ante exhaustuvudade [Carbonell e Goldstein 1998]. Por outra banda, esta problemática acerca da vixencia temporal dos documentos é unha das maiores problemáticas neste proxecto (e en calquera RAG que manexe información cambiante ao longo do tempo [Grofsky 2025]) e tratarase máis adiante, xa que require modificación en diversas partes do sistema.

2.4 Os modelos

2.5 Ferramentas usadas

3. Instalación e uso

4. Resultados



Results presentation here.

5. Conclusions

ola castro

Bibliografía

- Author, A. e B. Author (2025). “A Systematic Review of Key Retrieval-Augmented Generation (RAG) Systems: Progress, Gaps, and Future Directions”. En: *arXiv preprint* arXiv:2507.18910. Consultado: Diciembre 2025.
- Carbonell, Jaime e Jade Goldstein (1998). “The use of MMR, diversity-based reranking for reordering documents and producing summaries”. En: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '98. New York, NY, USA: ACM, pp. 335–336.
- Chakrabarti, Soumen, Martin Van den Berg e Byron Dom (1999). “Focused crawling: a new approach to topic-specific Web resource discovery”. En: *Computer Networks* 31.11-16, pp. 1623–1640. DOI: [10.1016/S1389-1286\(99\)00052-3](https://doi.org/10.1016/S1389-1286(99)00052-3).
- Grofsky, Mark (2025). “Solving Freshness in RAG: A Simple Recency Prior and the Limits of Heuristic Trend Detection”. En: *arXiv preprint* arXiv:2509.19376. Consultado: Diciembre 2025.
- Pezzuti, Francesca, Sean MacAvaney e Nicola Tonellotto (xul. de 2025). “Neural Prioritisation for Web Crawling”. En: *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*. ICTIR '25. ACM, 307–314. DOI: [10.1145/3731120.3744597](https://doi.org/10.1145/3731120.3744597). URL: <http://dx.doi.org/10.1145/3731120.3744597>.

A. Lista completa de *keywords*

A continuación móstrase o conxunto completo de palabras clave utilizadas polo *crawler* para determinar a relevancia das páxinas:

- | | | |
|------------------|-----------------|----------------------|
| ■ regulation | ■ programa | ■ bases |
| ■ reglamento | ■ requirements | ■ anexo |
| ■ normativa | ■ requisitos | ■ catalog |
| ■ procedure | ■ regulamento | ■ catalogo |
| ■ procedimiento | ■ regulación | ■ catálogo |
| ■ proceso | ■ procedimiento | ■ library |
| ■ form | ■ solitudine | ■ biblioteca |
| ■ formulario | ■ guía | ■ collection |
| ■ solicitud | ■ docente | ■ coleccion |
| ■ guideline | ■ asignatura | ■ colección |
| ■ guia | ■ política | ■ acquisition |
| ■ manual | ■ matrícula | ■ adquisicion |
| ■ policy | ■ inscripción | ■ adquisición |
| ■ politica | ■ académico | ■ loan |
| ■ norma | ■ convocatoria | ■ prestamo |
| ■ enrollment | ■ prazo | ■ préstamo |
| ■ matricula | ■ prazos | ■ reserve |
| ■ inscripcion | ■ documentación | ■ reserva |
| ■ administrative | ■ tramite | ■ interlibrary |
| ■ administrativo | ■ trámite | ■ interbibliotecario |
| ■ academic | ■ ordenanza | ■ reference |
| ■ academico | ■ resolución | ■ referencia |
| ■ calendar | ■ circular | ■ circulation |
| ■ calendario | ■ instrucciones | ■ circulacion |
| ■ syllabus | ■ instrucciones | ■ circulación |

▪ periodical	▪ fondos	▪ notification
▪ periodico	▪ serials	▪ notificacion
▪ periódico	▪ publicaciones seriadas	▪ notificación
▪ journal	▪ special collections	▪ registration
▪ revista	▪ colecciones especiais	▪ registro
▪ archive	▪ reading room	▪ protocol
▪ archivo	▪ sala de lectura	▪ protocolo
▪ arquivos	▪ stacks	▪ statute
▪ repository	▪ depósito	▪ estatuto
▪ repositorio	▪ microfilm	▪ ordinance
▪ classification	▪ microficha	▪ decree
▪ clasificacion	▪ digital library	▪ decreto
▪ clasificación	▪ biblioteca dixital	▪ resolution
▪ indexing	▪ opac	▪ resolucion
▪ indexacion	▪ marc	▪ official
▪ indexación	▪ application	▪ oficial
▪ cataloging	▪ deadline	▪ office
▪ catalogacion	▪ plazo	▪ oficina
▪ catalogación	▪ documentation	▪ department
▪ dewey	▪ documentacion	▪ departamento
▪ isbn	▪ certification	▪ service
▪ issn	▪ certificado	▪ servicio
▪ bibliographic	▪ certificación	▪ servizo
▪ bibliografico	▪ authorization	▪ unit
▪ bibliográfico	▪ autorizacion	▪ unidad
▪ holdings	▪ autorización	▪ unidade

B. Saída OCR da táboa de exemplo

Texto OCR

[Tipos de usuarios
Nº de documentos en préstamo
Días de préstamo
Renovaciones
Reservas
GRUPO 1
Estudiantado de Grado de centros propios y adscritos
Estudiantado de programa de movilidad (Erasmus, Sicue-Séneca)
Estudiantado de doble Grado, de simultaneidad 10 10 días Indefinidas | Límite 10 docs.
Estudiantado de grados interuniversitarios Estudiantado da Universidade Sénior GRUPO 2
Estudiantado de MASTERES e posgrados propios, incluyendo los de la Fundación Univer-
sidade da Coruña . , 15 21 días Indefinidas | Límite 15 docs. Estudiantado de Trabajos de
fin de grado. Personal de administración en servicio (PTGAS) GRUPO 3 Estudiantado de
Doctorado 35 30 días Indefinidas | Límite 35 docs. Personal investigador visitante; visitant
tant 'ersonal investigador visitante: visitante senior] visitante 35 30 días indefinidas | Límite
38 docs predoctoral o postdoctoral GRUPO 4 Becarios/as de investigación Curso académico
Personal contratado investigador (cada biblioteca podrá excluir 35 de este tipo de préstamo
Indefinidas | Límite 35 docs. documentos par razón de uso y disponibilidad) GRUPO 5 PDI
da UDC (incluyendo centros adscritos), de la Fundación UDC Curso académico Profesorado
emérito, jubilado incentivado y honorario (cada biblioteca podrá excluir Profesorado visi-
tante 100 de este tipo de préstamo Indefinidas | Límite 100 docs. Lectores/as documentos
par razón de uso y disponibilidad) GRUPO 6 Cual ! idad taria de la UDC cualquier persona
ajena a la comunidad universitaria dela 6 10 días indefinidas | Límite 6 docs
que sea autorizada por la Biblioteca Universitaria