Plataformas de Generación automática de Preguntas y Respuestas a partir de una entrada de datos.

Nota: Las ventajas y desventajas es en correspondencia con el fin o tarea en que lo vamos a usar, puede que para otras personas lo que es una desventaja sea una ventaja.

**Generación Automática de Preguntas (Usada actualmente)**

Repositorio: [indrajithi/genquest: Generación automática de preguntas mediante PNL (github.com)](https://github.com/indrajithi/genquest)

Este programa sencillo toma un archivo de texto como entrada y genera preguntas analizando cada oración.

Usa la biblioteca de Python textBlob para el análisis de textos, extraer las oraciones mediante la función parse(texto) y formular la pregunta con la función genQuestion(oración).

Ventajas

* Permite formular rápidamente preguntas a partir de un texto.
* No necesita de centro de datos (datasets) para realizar procesos de análisis, comprensión, etc. Por lo que no hay que descargar nada. Y el proceso es rápido.
* Al usar pocas dependencias no se tuvo problemas con incompatibilidad a la hora de trabajar en conjunto con otros paquetes de python que ya teníamos instalados. Incluso actualizando algunos seguía funcionando.
* Se trabaja en Python.

Desventajas

* Solo usa idioma inglés.
* Solo genera preguntas con ¿Qué es?, por lo que a veces se generan preguntas sin mucho sentido. Para el trabajo que estamos realizando es una gran desventaja.

Haystack (Se está probando para ser usada)

Documentación: [Haystack - Haystack (deepset.ai)](https://haystack.deepset.ai/overview/intro)

Haystack es un marco de código abierto para crear sistemas de búsqueda que funcionan de manera inteligente en grandes colecciones de documentos. Los avances recientes en PNL han permitido la aplicación de la respuesta a preguntas, la recuperación y el resumen a entornos del mundo real y Haystack está diseñado para ser el puente entre la investigación y la industria. (Documetación)

Esta plataforma posee varias herramientas para realizar diversos trabajos como convertir ficheros, traductor, procesamiento y clasificación de documentos, etc.

La biblioteca que usaremos es **farm-haystack** en su versión básica que trae lo necesario para lo que es generación de preguntas y respuestas a partir de una entrada de datos. La versión completa (Libre, al menos en la documentación no aparece que se sea de pago) trae muchas funciones más avanzadas dentro para hacer otros tipos de procesamientos y tareas.

Mediante la función **QuestionGenerator().generate(texto)** se generan las preguntas según el texto entrado. Si se muestra el resultado se ve una lista de preguntas (***list [preguntas…])***.

Ventajas

* Permite generar con muy buenos resultados preguntas a partir de una entrada de datos. Las preguntas generadas se ajustan y con sentido a los datos entrados.
* Usa centro de datos para realizar los procesos de análisis, comprensión, etc. Por lo que se tarda un poco, pero arroja buenos resultados.
* Genera todo tipo de preguntas, ¿Qué?, ¿Quién?, ¿Cuál?, etc.
* Buena documentación para aprender a usar las herramientas.
* Se trabaja en Python.

Desventajas

* Si bien usar un centro de datos proporciona mejores resultados para nosotros fue una desventaja debido a la conexión a internet pues hay que descargar casi 1gb de datos, se necesitaría de una buena velocidad de conexión y en caso de datos móviles contar con un paquete de más de 1gb. Lo que si una vez descargado ya no es necesario volver a hacerlo a menos que se desinstale el paquete **farm-haystack**.
* El idioma que maneja es el inglés.
* Aunque tiene buena documentación hay algunos casos donde puede resultar complicado entender algunas cosas al menos para alguien con poca experiencia.

Revisar esto e incluirlo:

<https://www.rws.com/language-weaver/blog/Issue-150-Automated-Question-Generation/>

**Español:**

<https://metatext.io/models/mrm8488-bert2bert-spanish-question-generation>

<https://huggingface.co/mrm8488/bert2bert-spanish-question-generation?text=Manuel+vive+en+Murcia%2C+Espa%C3%B1a>

Revisar este artículo:

<https://www.sciencedirect.com/science/article/abs/pii/S0957417418307565>

**Estados del arte:**

<https://link.springer.com/article/10.1007/s40593-019-00186-y>

### [Automatic question generation and answer assessment: a survey](https://telrp.springeropen.com/articles/10.1186/s41039-021-00151-1)

### [Biomedical question answering: A survey of approaches and challenges](https://dl.acm.org/doi/abs/10.1145/3490238)

### [Automatic question-answer pairs generation and question similarity mechanism in question answering system](https://link.springer.com/article/10.1007/s10489-021-02348-9)

**Buscar proyectos en GitHub**

https://github.com/search?l=Python&q=Automatic+Question+Generation&type=Repositories