

**DOCUMENTACIÒN DEL PROYECTO EDUQUETS**

**CURSO: CIENCIA DE DATOS**

**ENTREGADO POR: DUGLAS ALTAMAR GOMEZ**

**ROGER DARIO ESPITIA PEREZ**

**NIXON ANDRES PADILLA JIMENEZ**

**KATHERIN YULISA TOBIO DORIA**

**VIVETH SOFIA SEGOVIA**

**ARGUMEDO**

**PRESENTADO A: ALEXANDER TOSCANO**

**FACULTAD DE EDUCACIÒN Y CIENCIAS HUMANAS**

**LICENCIATURA EN INFORMATICA**

**UNIVERSIDAD DE CORDOBA**

## DESCRIPCIÓN GENERAL DEL PROYECTO



**EDUQUEST** es una innovadora plataforma educativa basada en inteligencia artificial que transforma contenidos escolares en experiencias narrativas interactivas. A través de **mundos generativos**, cada tema académico se convierte en una historia envolvente donde los estudiantes asumen un rol activo, enfrentando desafíos, tomando decisiones y aplicando conocimientos para avanzar en la trama.

En lugar de recibir una clase tradicional, el estudiante es introducido en un entorno ficticio original, con personajes y conflictos diseñados en función del contenido a aprender. Esta metodología favorece el aprendizaje por exploración, ensayo y error, y resolución de problemas, promoviendo una comprensión profunda, significativa y emocionalmente conectada del conocimiento.

## BUSQUEDA

### Prueba de relleno de historias y corpus de ROCStories

¡Noticias!

**Febrero de 2017:** La primera tarea compartida de la Prueba de Relatos Completos de Historias ha finalizado. ¡Puedes leer el resumen [aquí](#)! [Visita la página](#) de Tareas Compartidas para más detalles.

**Diciembre de 2016:** ¡Ya está disponible la edición de invierno de ROCStories 2017 con 52 666 historias nuevas! Accede al conjunto de datos a continuación.

**Noviembre de 2016:** La Prueba de Relatos Completos es ahora la tarea compartida para LSDSem'17. ¡Inscríbete antes del 15 de diciembre! Para más información, visita la [página](#) de Tareas Compartidas.

**Octubre de 2016:** ¿Tienes un sistema para el Story Cloze Test? ¡Ahora puedes enviar tus resultados a [CodaLab](#) para ayudar a la comunidad a mantenerse al día con el SOTA!

Complete [este](#) formulario para acceder al conjunto de datos, que es gratuito. La última versión incluye 98 159 ROCStories y 3744 instancias de Story Cloze Test. Si no recibe el correo electrónico en unos minutos, ¡contáctenos! **Manténgase al tanto de los próximos conjuntos de Story Cloze Test.**

El Test de Cloze Narrativo es un nuevo marco de razonamiento basado en el sentido común para evaluar la comprensión, la generación y el aprendizaje de guiones. Este test requiere un sistema para elegir el final correcto para una historia de cuatro oraciones. Proponemos el Test de Cloze Narrativo para reemplazar el método más avanzado para evaluar el aprendizaje de la estructura narrativa, el Test de Cloze Narrativo (Chambers y Jurafsky, 2008).

Para habilitar la Prueba de Relatos Completos, creamos un nuevo corpus de relatos de sentido común de cinco oraciones, «ROCStories». Este corpus es único por dos razones: (1) captura un amplio conjunto de relaciones causales y temporales de sentido común entre eventos cotidianos, y (2) es una colección de alta calidad de relatos de la vida cotidiana que también puede utilizarse para la generación de relatos.

### Ejemplos

Prueba de relleno de la historia

Realizamos una búsqueda y encontramos un trabajo similar: [ROCStories](#), que se trata de una **colección de historias breves de sentido común** desarrollada por el Laboratorio de Lenguaje Natural de la Universidad de Rochester. El proyecto tiene como objetivo entrenar y evaluar modelos de inteligencia artificial en la comprensión narrativa y el razonamiento causal a través de historias de cinco oraciones.

Las historias están diseñadas para representar situaciones cotidianas que contienen una progresión lógica de eventos, permitiendo a los modelos predecir desenlaces plausibles o detectar incoherencias narrativas. Aunque el enfoque de ROCStories no está orientado directamente a la educación, comparte con *EDUQUEST* el interés en la estructura narrativa como medio para representar el conocimiento y facilitar el razonamiento.

## ¿Cómo puede servir al proyecto EDUQUEST?

### Ejemplos

Prueba de relleno de la historia

Contexto	Final derecho	Final equivocado
A Karen le asignaron una compañera de piso durante su primer año de universidad. Su compañera le propuso ir a un concierto en una ciudad cercana. Karen aceptó encantada. El concierto fue absolutamente emocionante.	Karen se hizo buena amiga de su compañera de cuarto.	Karen odiaba a su compañera de cuarto.
Jim obtuvo su primera tarjeta de crédito en la universidad. No tenía trabajo, así que compró todo con su tarjeta. Después de graduarse, acumuló una deuda de \$10,000. Jim se dio cuenta de que había sido una tontería gastar tanto dinero.	Jim decidió idear un plan de pago.	Jim decidió abrir otra tarjeta de crédito.
Gina perdió su teléfono en casa de sus abuelos. No estaba en la sala. Se dio cuenta de que había estado en el coche antes. Tomó las llaves de su padre y salió corriendo.	Ella encontró su teléfono en el auto.	Ella ya no quería su teléfono.

### Historias de ROCS

Título	Historia de cinco oraciones
La prueba	Jennifer tiene un examen importante mañana. Se estresó tanto que se quedó despierta toda la noche. Al día siguiente fue a clase agotada. Su profesora le dijo que el examen se posponía para la semana que viene. Jennifer se sintió agriludece.
El huracán	Morgan y su familia vivían en Florida. Oyeron que se avecinaba un huracán. Decidieron evacuar a casa de un familiar. Llegaron y se enteraron por las noticias de que era una tormenta terrible. Se sintieron afortunados de haber evacuado en ese momento.
Salsa de espagueti	Tina preparó espaguetis para su novio. Le costó mucho trabajo, pero estaba muy orgullosa. Su novio se comió todo el plato y dijo que estaba bueno. Tina lo probó y se dio cuenta de que estaba asqueroso. Le conmovió que él fingiera que estaba bueno para no herirla.

### 1. Evaluación de comprensión narrativa

El *Story Cloze Test* permite medir si un modelo de IA es capaz de seleccionar el desenlace correcto de una historia, evaluando su capacidad de comprensión del lenguaje, razonamiento causal y uso del sentido común. Esto es especialmente útil para *EDUQUEST*, donde los modelos deben entender y generar narrativas coherentes vinculadas a contenidos académicos.

### 2. Entrenamiento de modelos generativos

El corpus *ROCStories*, compuesto por historias breves de cinco oraciones, es ideal para entrenar modelos generativos que puedan crear narrativas lógicas, estructuradas y humanas. *EDUQUEST* podría beneficiarse de este corpus como base para entrenar modelos que generen mundos educativos de forma autónoma y con coherencia narrativa.

### 3. Datos anotados para aprendizaje supervisado

Los datos del *Story Cloze Test* incluyen tanto desenlaces correctos como incorrectos, lo cual es útil para entrenar modelos en tareas de clasificación binaria y mejora de la coherencia textual. Esto podría fortalecer la capacidad de *EDUQUEST* para generar múltiples rutas narrativas con decisiones significativas que impacten el desarrollo de la historia.

## Acceso a ROCStories y Story Cloze Test

Tras diligenciar el formulario de solicitud, recibimos por correo electrónico los **enlaces oficiales de descarga** del corpus **ROCStories** y del marco de evaluación **Story Cloze Test**, recursos fundamentales en el campo del procesamiento de lenguaje natural. A continuación, se detallan los conjuntos disponibles:

### ROCStories – Historias completas de cinco oraciones

- Winter 2017 set: <https://goo.gl/00YkPK>
- Spring 2016 set: <https://goo.gl/7R59b1>

Estos conjuntos incluyen cientos de microhistorias diseñadas para capturar el sentido común narrativo y sirven como base para el entrenamiento de modelos de generación o comprensión de texto.

## **Story Cloze Test – Evaluación de comprensión narrativa**

### **Winter 2018 (conjunto recomendado):**

Validación (val set): <https://goo.gl/XWjas1> Prueba (test set):

<https://goo.gl/BcTtB4> **Spring 2016 set:**

Validación (val set): <https://goo.gl/cDmS6I> Prueba (test set): <https://goo.gl/iE31Qm>

## 1. Descripción detallada del Proyecto

El Proyecto de Inferencia Narrativa con IA tiene como objetivo entrenar un modelo de aprendizaje automático para clasificar el nivel de inferencia necesario para comprender una historia narrativa. Los niveles de inferencia son:

- **Literal (Nivel 1):** Comprensión directa de los hechos explícitos.
- **Causal (Nivel 2):** Entender relaciones de causa y efecto en la narrativa.
- **Crítica o Evaluativa (Nivel 3):** Interpretar emociones, intenciones o implicaciones más profundas.

El proyecto utiliza un dataset basado en el formato *Story Cloze Test*, que incluye historias cortas con un final correcto y un final incorrecto, junto con su nivel de inferencia. El modelo entrenado puede predecir el nivel de inferencia de un nuevo texto a través de una API Flask.

### 1.1. Aplicaciones

- **Educación:** Evaluar la complejidad de textos para estudiantes.
- **Procesamiento de Lenguaje Natural (PLN):** Clasificación y generación de narrativas.
- **Evaluación Automática:** Determinar el nivel de comprensión necesario para textos.

## 2. Estructura del Proyecto

- `/datos:` Contiene el archivo `ROCStories_Ejemplo_Combinado.xlsx` con los datos de entrenamiento.
- `/modelo:` Almacena el modelo entrenado (`modelo_inferencia.pkl`) y el vectorizador (`vectorizador.pkl`).
- `/api:` Incluye el microservicio Flask (`microservicio_flask.py`) para realizar predicciones.

## 3. Requisitos

Para ejecutar el proyecto, necesitas:

- Python 3.8 o superior.
- Dependencias (instálalas con pip):

```
1 pip install pandas scikit-learn joblib flask
```

- Un entorno con acceso al archivo Excel en /datos.

## 4. Instalación

1. **Clona el repositorio** (si aplica) o asegúrate de tener los archivos del proyecto.
2. **Crea un entorno virtual** (opcional, pero recomendado):

```
1 python -m venv venv
2 source venv/bin/activate    % En Windows: venv\Scripts\activate
```

3. **Instala las dependencias:**

```
1 pip install -r requirements.txt
```

Si no tienes un archivo requirements.txt, instala manualmente:

```
1 pip install pandas scikit-learn joblib flask
```

4. **Verifica la estructura:** Asegúrate de que las carpetas /datos, /modelo y /api estén en el directorio raíz, junto con los archivos entrenar\_modelo.py y microservicio\_flask.py.

## 5. Uso

### 5.1. Entrenar el Modelo

El script entrenar\_modelo.py entrena un modelo de regresión logística usando el dataset en /datos. Sigue estos pasos:

1. Asegúrate de que el archivo ROCStories\_Ejemplo\_Combinado.xlsx esté en /datos.
2. Ejecuta el script:

```
1 python entrenar_modelo.py
```

3. Esto genera dos archivos en /modelo:

- modelo\_inferencia.pkl: El modelo entrenado.
- vectorizador.pkl: El vectorizador TF-IDF.

El script combina las oraciones de cada historia, las vectoriza usando TF-IDF y entrena un modelo de regresión logística para predecir el nivel numérico (1, 2 o 3).

### 5.2. Ejecutar la API

El microservicio Flask permite realizar predicciones en tiempo real. Para usarlo:

1. Asegúrate de que los archivos modelo\_inferencia.pkl y vectorizador.pkl estén en /modelo.
2. Ejecuta el script:

```
1 python api/microservicio_flask.py
```

3. La API se ejecutará en <http://localhost:5000>.

### 5.3. Hacer Predicciones

Envía una solicitud POST a la ruta /inferir con un JSON que contenga el texto de la historia. Ejemplo:

```
1 curl -X POST http://localhost:5000/inferir -H "Content-Type: application/json" -d '{"texto": "Pedro se quedó callado durante la reunión. Todos notaron su comportamiento extraño. Él no levantó la mirada ni una vez. Cuando terminó, salió sin decir nada. Estaba decepcionado con la decisión tomada."}'
```

Respuesta esperada:

```
1 { "nivel": 3 }
```

Esto indica que el texto requiere una inferencia crítica (nivel 3).

## 6. Detalles Técnicos

### 6.1. Dataset

El archivo ROCStories\_Ejemplo\_Combinado.xlsx contiene historias con las siguientes columnas:

- **ID\_Historia:** Identificador único.
- **Oración\_1 a Oración\_4:** Partes de la historia.
- **Final\_Correcto:** Final lógico de la historia.
- **Final\_Incorrecto:** Final ilógico (no usado en el entrenamiento).
- **Nivel\_Numérico:** Nivel de inferencia (1 = Literal, 2 = Causal, 3 = Crítica).
- **Nivel\_Eduquest:** Descripción del nivel.

Ejemplo:

ID	Oración 1	Oración 2	Oración 3	Oración 4	Final Correcto
1	Luis se despertó tarde...	Saltó de la cama...	Salió corriendo...	Tomó el primer bus...	Llegó a tiempo...

### 6.2. Proceso de Entrenamiento

1. **Carga de Datos:** Se lee el archivo Excel con pandas.
2. **Preprocesamiento:** Las oraciones y el final correcto se concatenan en un solo texto.
3. **Vectorización:** Se usa TfidfVectorizer para convertir los textos en vectores numéricos.
4. **Entrenamiento:** Se entrena un modelo de regresión logística (LogisticRegression) para clasificar el nivel numérico.
5. **Guardado:** El modelo y el vectorizador se guardan con joblib.

### 6.3. API Flask

- **Endpoint:** /inferir (POST)
- **Entrada:** JSON con un campo texto (string).
- **Salida:** JSON con el campo nivel (entero: 1, 2 o 3).
- **Funcionamiento:** El texto se vectoriza con el vectorizador guardado, y el modelo predice el nivel de inferencia.

## 7. Ejemplo Completo

1. Entrena el modelo:

```
1 python entrenar_modelo.py
```

2. Inicia la API:

```
1 python api/microservicio_flask.py
```

3. Envía una solicitud:

```
1 curl -X POST http://localhost:5000/inferir -H "Content-Type :  
application/json" -d '{"texto": "Marta encontró una billetera en  
la calle. La abrió y vio una cédula adentro. Buscó el número en  
redes sociales. Llamó al dueño para devolvérsela. El dueño la  
agradeció con una sonrisa."}'
```

4. Recibe:

```
1 { "nivel": 1 }
```

## 8. Notas Adicionales

- **Escalabilidad:** Para datasets más grandes, considera optimizar el modelo (por ejemplo, usar redes neuronales) o ajustar los hiperparámetros de TfidfVectorizer o LogisticRegression.
- **Limitaciones:** El modelo depende de la calidad y tamaño del dataset. Un dataset pequeño puede limitar la generalización.
- **Mejoras Futuras:**
  - Incorporar modelos más avanzados (e.g., transformers como BERT).
  - Ampliar el dataset con más historias.
  - Agregar validación cruzada para mejorar la robustez del modelo.

## 9. Preguntas Frecuentes

**¿Qué pasa si el texto enviado no sigue el formato de cinco oraciones?** El modelo puede procesar textos de cualquier longitud, pero su precisión depende de que el texto sea similar a los datos de entrenamiento (historias narrativas claras).

**¿Cómo puedo contribuir al proyecto?**

- Agrega más datos al archivo Excel.



- Experimenta con otros algoritmos de clasificación en `entrenar_modelo.py`.
- Mejora la API con nuevas funcionalidades, como validación de entrada.