

GUION DE PRESENTACION

Sistema de Optimizacion de Siembra
para la Mixteca Oaxaqueña

Equipo de Inteligencia Artificial

Diciembre 2025

Integrante	Fase	Tiempo Est.
Ossiel Alejandro Acevedo Herrera	Fase 0: Datos	~8 min
Aneli Arce Jimenez	Fase 1: Red LSTM	~5 min
Ramon Aragon Toledo	Fase 2: Logica Difusa	~7 min
Cristian Rodriguez Gomez	Fase 3: Optimizacion	~10 min
	Total estimado:	30 min

Índice

1. PARTE 1: OSSIEL ALEJANDRO ACEVEDO HERRERA	3
1.1. Diapositiva 1: Portada	3
1.2. Diapositiva 2: Introduccion	3
1.3. Diapositiva 3: Objetivo del Proyecto	3
1.4. Diapositiva 4: Arquitectura del Sistema	4
1.5. Diapositiva 5: Estructura del Proyecto	4
1.6. Diapositiva 6: Fase 0 - Preparacion de Datos	4
1.7. Diapositiva 31: Pipeline Completo de Datos	5
1.8. Diapositiva 32: Preparacion de Datos (NASA POWER)	5
2. PARTE 2: ANELI ARCE JIMENEZ	6
2.1. Diapositiva 7: Red Neuronal LSTM - Concepto	6
2.2. Diapositiva 8: Pronostico Climatico 2026	6
2.3. Diapositiva 9: Introduccion al Codigo - Gestor Climatico	6
2.4. Diapositiva 10: Codigo - Gestor Climatico	7
3. PARTE 3: RAMON ARAGON TOLEDO	8
3.1. Diapositiva 11: Sistema de Logica Difusa - Concepto	8
3.2. Diapositiva 12: Variables de Entrada	8
3.3. Diapositiva 13: Variable de Salida (Amplitud)	9
3.4. Diapositiva 14: Introduccion al Codigo - Sistema Difuso	9
3.5. Diapositiva 15: Codigo - Sistema Difuso	9
3.6. Diapositiva 16: Introduccion al Codigo - Test Visual	9
3.7. Diapositiva 17: Codigo - Test Visual Difuso	10
4. PARTE 4: CRISTIAN RODRIGUEZ GOMEZ	11
4.1. Diapositiva 18: Algoritmo Genetico - Concepto	11
4.2. Diapositiva 19: Evolucion del Fitness	11
4.3. Diapositiva 20: Introduccion al Codigo - Algoritmo Genetico	11
4.4. Diapositiva 21: Codigo - Algoritmo Genetico	12
4.5. Diapositivas 22-27: Codigos Adicionales	12
4.6. Diapositiva 28: Resultados del Sistema	12
4.7. Diapositiva 29: Alternativa PSO	13
4.8. Diapositiva 30: Codigo PSO	13
4.9. Diapositiva 33: Conclusiones	13
4.10. Diapositiva 34: Gracias	13
5. ANEXO: PREGUNTAS FRECUENTES Y RESPUESTAS	14

1 PARTE 1: OSSIEL ALEJANDRO ACEVEDO HERRERA

(Introduccion General + Fase 0: Preparacion de Datos)

1.1 Diapositiva 1: Portada

OSSIEL dice:

“Buenos dias/tardes. Somos el equipo conformado por Aneli, Cristian, Ramon y un servidor Ossiel. El dia de hoy les presentaremos nuestro proyecto: **Sistema de Optimizacion de Siembra para la Mixteca Oaxaqueña**, el cual integra tres tecnicas de Inteligencia Artificial: Redes Neuronales, Logica Difusa y Algoritmos Geneticos.”

1.2 Diapositiva 2: Introduccion

OSSIEL dice:

“Para contextualizar el problema: La Mixteca Oaxaqueña es una region que enfrenta **desafios climaticos muy significativos** para la agricultura tradicional de maiz.

Determinar la **fecha optima de siembra** es crucial por tres razones principales:

- Primero, para **maximizar el rendimiento** del cultivo.
- Segundo, para **minimizar el riesgo climatico** ante sequias o lluvias excesivas.
- Y tercero, para **aprovechar al maximo** las condiciones de temperatura y precipitacion.

Nuestro sistema combina **tres tecnicas de IA** para encontrar automaticamente la mejor fecha de siembra, sin depender unicamente del conocimiento tradicional.”

1.3 Diapositiva 3: Objetivo del Proyecto

OSSIEL dice:

“El objetivo principal del proyecto es **desarrollar un sistema inteligente** que determine la ventana de siembra optima para maiz en la Mixteca.

Para lograrlo, integramos tres tecnicas complementarias:

- **Redes Neuronales LSTM** para la predicion climatica.
- **Logica Difusa** para evaluar las condiciones de siembra.
- **Algoritmos Geneticos** para optimizar y encontrar la mejor fecha.

El resultado final es proveer al agricultor una **recomendacion clara y fundamentada**, basada completamente en datos.”

1.4 Diapositiva 4: Arquitectura del Sistema

OSSIEL dice:

“La arquitectura del sistema se divide en un **pipeline de 4 fases**:

- **Fase 0** que es la que yo desarrolle: Se encarga de la **recolección de datos** históricos de fuentes como CONAGUA y NASA POWER, incluyendo el filtrado y limpieza.
- **Fase 1** que desarrollo Aneli: Utiliza una **Red Neuronal LSTM** para predecir la temperatura y precipitación de todo el año 2026.
- **Fase 2** que desarrollo Ramon: Implementa un **Sistema de Logica Difusa** que evalúa qué tan apto es cada día para sembrar, con un puntaje de 0 a 100.
- **Fase 3** que desarrollo Cristian: Aplica un **Algoritmo Genético** (y también PSO) para buscar el día óptimo de siembra.

El flujo va de Fase 0 hasta Fase 3, y el resultado final es la **fecha óptima recomendada**.”

1.5 Diapositiva 5: Estructura del Proyecto

OSSIEL dice:

“A nivel técnico, el proyecto tiene una **organización modular**:

- El archivo `main.py` es el punto de entrada.
- La carpeta `data/processed/` contiene los pronósticos en formato CSV.
- En `src/neural/` está la implementación de la Red LSTM.
- En `src/fuzzy/` está el sistema de lógica difusa.
- Y en `src/optimization/` están los algoritmos genéticos y PSO.

El flujo de datos es: el CSV con 365 días de pronóstico es leído por el gestor climático, luego el sistema difuso evalúa la aptitud, el algoritmo genético optimiza, y finalmente obtenemos el día óptimo del año.”

1.6 Diapositiva 6: Fase 0 - Preparación de Datos

OSSIEL dice:

“Ahora les explico la **Fase 0**, que fue mi responsabilidad.

El origen de los datos proviene de la estación meteorológica de **Huajuapan de León**. Contamos con un **periodo histórico de 19 años** de registros, hasta 2024. Es importante mencionar que no se incluyen datos de 2025 porque aún no están disponibles completamente.

El **procesamiento** que realicé incluyó tres pasos:

1. **Limpieza**: Filtrado de ruido y tratamiento de valores nulos en el dataset.
2. **Selección del modelo**: Evalúe múltiples arquitecturas de redes neuronales y seleccione la que presentó el **menor error** en las predicciones.
3. **Entrenamiento**: El modelo ganador se entrenó con los datos depurados para generar el archivo de pronósticos 2026.

”

1.7 Diapositiva 31: Pipeline Completo de Datos

OSSIEL dice:

“Para que tengan una vision mas clara del flujo de archivos, les muestro el **pipeline completo**:

Del lado izquierdo tenemos los **scripts de procesamiento**:

1. `preparacion_datos.py`: Descarga los datos de la API de NASA POWER, los limpia y prepara las variables ciclicas.
2. `entrenamiento_modelo.py`: Diseña y entrena la LSTM usando Keras.
3. `generar_pronostico.py`: Usa el modelo entrenado para predecir el clima de 2026.

Del lado derecho estan los **archivos generados**:

- `Reporte_Humano_Huajuapan.csv`: Datos historicos limpios para validacion visual.
- `Dataset_Entrenamiento_IA.csv`: Ventanas de 15 dias formateadas para la LSTM.
- `mejor_modelo_clima.h5`: La red neuronal entrenada con todos sus pesos.
- `Pronostico_2026_IA.csv`: La prediccion diaria que consume el algoritmo genetico.

”

1.8 Diapositiva 32: Preparacion de Datos (NASA POWER)

OSSIEL dice:

“Profundizando en la **fuente de datos**:

Utilizamos la **API de NASA POWER**, que proporciona datos satelitales de temperatura y precipitacion. La ubicacion especifica es Huajuapan de Leon, Oaxaca, y el periodo abarca 19 anos de historia.

El **procesamiento aplicado** incluye:

1. **Limpieza**: Eliminacion de valores nulos y datos anomalos.
2. **Codificacion Ciclica**: Transformamos las fechas a valores de Seno y Coseno para que la red neuronal pueda capturar la estacionalidad del clima.
3. **Normalizacion**: Escalamos todas las variables para que esten en rangos compatibles con la red neuronal.
4. **Ventaneo**: Creamos secuencias de 15 dias consecutivos, que son la entrada que la LSTM espera recibir.

Con esto, le paso la palabra a Aneli para que nos explique la Fase 1.”

NOTA:

Transicion: Ossiel cede el microfono a Aneli.

2 PARTE 2: ANELI ARCE JIMENEZ

(Fase 1: Red Neuronal LSTM)

2.1 Diapositiva 7: Red Neuronal LSTM - Concepto

ANELI dice:

“Gracias Ossiel. Ahora les explicare la **Fase 1**, que corresponde a la red neuronal. **Que es una Red LSTM?** Las siglas significan **Long Short-Term Memory**, y es un tipo especial de red neuronal recurrente. Esta especializada en aprender **patrones temporales** en secuencias de datos, y lo mas importante: es capaz de ‘recordar’ dependencias a largo plazo.

En nuestro proyecto, la LSTM funciona asi:

- **Entrada:** Recibe los datos climaticos historicos de temperatura y lluvia.
- **Salida:** Genera una prediccion del clima para cada dia de 2026.
- El resultado se guarda en el archivo **Pronostico_2026_IA.csv** con 365 registros, uno por cada dia del año.

”

2.2 Diapositiva 8: Pronostico Climatico 2026

ANELI dice:

“Aqui pueden observar el **resultado visual** de la prediccion de la red LSTM. La grafica muestra la **prediccion de temperatura y precipitacion** para todo el año 2026. Pueden notar como el modelo captura los patrones estacionales: temperaturas mas altas en primavera-verano y el periodo de lluvias concentrado en los meses de junio a septiembre.

Esta informacion es crucial porque alimenta directamente al sistema de logica difusa que explicara Ramon.”

2.3 Diapositiva 9: Introduccion al Codigo - Gestor Climatico

ANELI dice:

“Ahora les explico el **Gestor Climatico**, que es un modulo clave de la Fase 1. Su proposito es actuar como el **puente** entre los datos crudos y el sistema inteligente. Se encarga de leer el archivo CSV que genero la red neuronal y preparar los datos en el formato que necesita el sistema difuso.

El archivo se encuentra en **src/neural/gestor_climatico.py**.”

2.4 Diapositiva 10: Codigo - Gestor Climatico

ANELI dice:

“Aqui vemos el pseudocodigo del gestor climatico.

La funcion `obtener_clima_real` recibe dos parametros: el dia de inicio y la duracion del ciclo de cultivo, que por defecto son 120 dias.

El proceso es:

1. Calcula los indices de las fechas que necesita extraer.
2. Lee el archivo CSV generado por la LSTM.
3. Extrae la ventana de tiempo correspondiente.
4. Formatea la salida como una lista de diccionarios, donde cada dia tiene su temperatura y lluvia.

Este modulo es fundamental porque permite que el algoritmo genetico consulte el clima de cualquier periodo del año. Ahora le cedo la palabra a Ramon para la Fase 2.”

NOTA:

Transicion: Aneli cede el microfono a Ramon.

3 PARTE 3: RAMON ARAGON TOLEDO

(Fase 2: Sistema de Logica Difusa)

3.1 Diapositiva 11: Sistema de Logica Difusa - Concepto

RAMÓN dice:

“Gracias Aneli. Yo les explicare la **Fase 2**, que corresponde al sistema de Logica Difusa. **Que es la Logica Difusa?** Es una extension de la logica booleana tradicional que maneja **grados de verdad**. En lugar de solo ‘verdadero’ o ‘falso’, podemos tener valores intermedios. Esto nos permite modelar conceptos imprecisos como ‘temperatura optima’ o ‘lluvia adecuada’, que no tienen limites exactos.

Los **componentes de nuestro sistema** son:

- **Variables de entrada:** Temperatura en grados Celsius y Precipitacion en milimetros.
- **Variable de salida:** Amplitud de siembra, con un puntaje de 0 a 100.
- **Funciones de membresia:** Usamos formas trapezoidales y triangulares.
- **Reglas de inferencia:** Definimos 9 reglas del tipo IF-THEN que capturan el conocimiento experto sobre agricultura.

”

3.2 Diapositiva 12: Variables de Entrada

RAMÓN dice:

“Aqui pueden ver las **funciones de membresia** de las variables de entrada.

Para la **Temperatura**, definimos tres categorias:

- ‘Baja’: cuando esta entre 5 y 18 grados.
- ‘Optima’: el pico esta alrededor de 25 grados.
- ‘Alta’: temperaturas superiores a 32 grados.

Para la **Precipitacion**, tambien tenemos tres categorias:

- ‘Escasa’: muy poca lluvia, lo cual es malo para el cultivo.
- ‘Adecuada’: el rango ideal de precipitacion.
- ‘Excesiva’: demasiada lluvia, que puede danar los cultivos.

Estas funciones permiten que valores numericos como ‘28 grados’ se traduzcan a terminos linguisticos como ‘60 % optima y 40 % alta.’”

3.3 Diapositiva 13: Variable de Salida (Amplitud)

RAMÓN dice:

“La variable de **salida** es la Amplitud de Siembra, que va de 0 a 100.

La interpretacion del puntaje es:

- **0 a 35 (Baja):** Condiciones adversas, no se recomienda sembrar.
- **25 a 75 (Media):** Condiciones aceptables pero con cierto riesgo.
- **65 a 100 (Alta):** Condiciones ideales para la siembra.

Este puntaje es lo que el algoritmo genetico va a acumular para cada dia, sumando la aptitud de los 120 dias del ciclo de cultivo.”

3.4 Diapositiva 14: Introduccion al Codigo - Sistema Difuso

RAMÓN dice:

“Ahora veamos el codigo del sistema difuso.

Este modulo define el ‘**cerebro**’ de **evaluacion** del sistema. Aqui configuramos las variables linguisticas con sus rangos, y establecemos las **reglas de inferencia** que determinan que tan bueno es un dia para sembrar.

El archivo esta en `src/fuzzy/fuzzy_system.py`.”

3.5 Diapositiva 15: Codigo - Sistema Difuso

RAMÓN dice:

“El pseudocodigo muestra los cuatro pasos principales:

1. **Definir Antecedentes y Consecuente:** Creamos las variables de temperatura, lluvia y amplitud con sus rangos.
2. **Definir Funciones de Membresia:** Por ejemplo, para temperatura definimos ‘baja’ como trapezoidal de 5 a 18 grados, ‘optima’ como triangular centrada en 25, y ‘alta’ de 32 en adelante.
3. **Definir las Reglas:** La regla ideal dice: SI la lluvia es adecuada Y la temperatura es optima, ENTONCES la amplitud es alta. La regla mala dice: SI la lluvia es escasa Y la temperatura es alta, ENTONCES la amplitud es baja. En total tenemos 9 reglas que cubren todas las combinaciones.
4. **Crear el Sistema de Control:** Combinamos todas las reglas y creamos una simulacion que podemos ejecutar.

”

3.6 Diapositiva 16: Introduccion al Codigo - Test Visual

RAMÓN dice:

“Tambien desarrolle un script de **prueba visual** llamado `test_fuzzy_visual.py`.

Su proposito es **verificar visualmente** que el sistema difuso funciona correctamente. Permite injectar valores manuales de temperatura y lluvia para ver que puntaje genera el sistema, y ayuda a entender que reglas se estan activando en casos especificos.”

3.7 Diapositiva 17: Código - Test Visual Difuso

RAMÓN dice:

“El código del test es sencillo:

Primero importamos el sistema difuso global. Luego definimos valores de prueba, por ejemplo, lluvia de 20 mm y temperatura de 25 grados.

Inyectamos estos valores al sistema, ejecutamos el cálculo que hace la fuzzificación, inferencia y defuzzificación, y obtenemos el resultado.

Finalmente, podemos visualizar qué reglas se activaron. Esto fue muy útil durante el desarrollo para verificar que las reglas estuvieran correctamente configuradas.

Ahora le paso la palabra a Cristian para la Fase 3.”

NOTA:

Transicion: Ramon cede el microfono a Cristian.

4 PARTE 4: CRISTIAN RODRIGUEZ GOMEZ

(Fase 3: Algoritmos de Optimizacion + Cierre)

4.1 Diapositiva 18: Algoritmo Genetico - Concepto

CRISTIAN dice:

“Gracias Ramon. Finalmente, les explicare la **Fase 3**, que es donde todo el sistema se integra para encontrar la fecha optima.

Que es un Algoritmo Genetico? Es una tecnica de optimizacion **inspirada en la evolucion natural**. Trabajamos con una poblacion de soluciones que ‘evolucionan’ hacia el optimo mediante operadores de seleccion, cruce y mutacion.

En nuestro proyecto:

- El **cromosoma** es simplemente un dia del año, del 1 al 365.
- El **fitness** o aptitud se calcula sumando los puntajes del sistema difuso para los 120 dias del ciclo de cultivo.
- El **objetivo** es encontrar el dia que tenga el maximo fitness acumulado.

”

4.2 Diapositiva 19: Evolucion del Fitness

CRISTIAN dice:

“Esta grafica muestra la **evolucion del fitness** a lo largo de las generaciones.

Pueden observar como la solucion mejora progresivamente. En las primeras generaciones hay mucha variabilidad porque el algoritmo esta explorando el espacio de busqueda. Conforme avanzan las generaciones, la curva se estabiliza porque ya encontro una buena region.

Esta convergencia nos indica que el algoritmo esta funcionando correctamente.”

4.3 Diapositiva 20: Introduccion al Codigo - Algoritmo Genetico

CRISTIAN dice:

“El algoritmo genetico es el **motor de optimizacion** del proyecto.

En lugar de probar cada dia del año uno por uno, lo cual seria fuerza bruta, evolucionamos una poblacion de fechas candidatas. Cada candidato se evalua usando la funcion de fitness que consulta el clima de los 120 dias siguientes.

El archivo esta en `src/optimization/algoritmo_genetico.py`.”

4.4 Diapositiva 21: Codigo - Algoritmo Genetico

CRISTIAN dice:

“Veamos el pseudocodigo:

La funcion `fitness_func` recibe una solucion (un dia de siembra). Primero verifica restricciones: si el dia es mayor a 240, lo penaliza porque no habria tiempo de cosechar antes de fin de año.

Luego obtiene el clima para los siguientes 120 dias usando el gestor climatico de Aneli. Para cada dia, calcula la aptitud con el sistema difuso de Ramon y suma todo.

La funcion `correr_optimizacion` configura el algoritmo genetico con 50 generaciones, 20 individuos por poblacion, y lo ejecuta. Al final retorna la mejor solucion encontrada.”

4.5 Diapositivas 22-27: Codigos Adicionales

CRISTIAN dice:

“Brevemente menciono los modulos adicionales:

Graficar Panorama (`graficar_panorama.py`): Herramienta de validacion que calcula la aptitud para **todos** los dias del año usando fuerza bruta. Esto nos permite verificar si el algoritmo genetico realmente encontro el optimo global.

Mocks (`src/mock.py`): Modulos de prueba que generan datos simulados. Son esenciales para desarrollo y pruebas unitarias.

Main (`main.py`): Punto de entrada de la aplicacion. Orquesta todo el flujo y presenta los resultados al usuario.”

4.6 Diapositiva 28: Resultados del Sistema

CRISTIAN dice:

“Pasemos a los **resultados observados**:

El algoritmo genetico **converge consistentemente** hacia fechas en la temporada de lluvias, generalmente entre mayo y junio. Esto coincide con el conocimiento tradicional campesino de la region.

Las **ventajas del sistema** son:

- Es **automatizado**: no requiere intervencion manual.
- Esta **fundamentado**: se basa en pronosticos climaticos reales.
- Es **adaptable**: puede actualizarse cada ano con nuevos datos.
- Es **interpretable**: gracias a la logica difusa, podemos explicar por que se recomienda cierta fecha.

”

4.7 Diapositiva 29: Alternativa PSO

CRISTIAN dice:

“Como alternativa al algoritmo genetico, tambien implemente el algoritmo de **Enjambre de Particulas** o PSO.

Que es PSO? Es un algoritmo inspirado en el comportamiento de bandadas de aves o cardumenes. Cada ‘particula’ representa una fecha candidata y se mueven por el espacio de busqueda, influenciadas por su mejor posicion historica y la mejor del grupo.

Las **ventajas de PSO** sobre el algoritmo genetico son:

- Tiene menos parametros de configuracion.
- La convergencia es mas suave y predecible.
- No requiere operadores de cruce ni mutacion.

Ambos algoritmos llegan a resultados similares, lo cual valida que el optimo encontrado es robusto.”

4.8 Diapositiva 30: Codigo PSO

CRISTIAN dice:

“El codigo del PSO usa la libreria Mealpy. La funcion objetivo es practicamente identica al fitness del algoritmo genetico: valida restricciones, obtiene el clima, y suma la aptitud de cada dia.

La configuracion usa 50 epochas y 20 particulas. El resultado se obtiene llamando a `modelo.solve()`.”

4.9 Diapositiva 33: Conclusiones

CRISTIAN dice:

“Para cerrar, nuestros **logros tecnicos** fueron:

- Implementamos exitosamente un sistema hibrido que integra **3 tecnicas de IA**.
- La **Red LSTM** logra predicciones climaticas para todo el año.
- El **Sistema Difuso** traduce condiciones climaticas en aptitud de siembra.
- El **Algoritmo Genetico** (y PSO) encuentran eficientemente la fecha optima.

El **impacto potencial** de este sistema incluye:

- Ser una herramienta de apoyo real para agricultores de la Mixteca.
- Contribuir a la reduccion del riesgo de perdida de cosechas.
- Ser un modelo replicable para otras regiones agricolas de Mexico.

”

4.10 Diapositiva 34: Gracias

CRISTIAN dice:

“Con esto concluimos nuestra presentacion. Agradecemos su atencion.

Tienen alguna pregunta?”

5 ANEXO: PREGUNTAS FRECUENTES Y RESPUESTAS

Possible Pregunta	Respuesta Sugerida
Por que eligieron 120 dias como ciclo de cultivo?	El ciclo de 120 dias es el periodo tipico de maduracion del maiz en la region de la Mixteca Oaxaqueña.
Que tan precisa es la prediccion de la LSTM?	La LSTM tiene un error promedio de $\pm 2\text{-}3^{\circ}\text{C}$ en temperatura y $\pm 5\text{mm}$ en precipitacion, basado en validacion con datos historicos.
Por que usaron 9 reglas difusas y no mas?	Las 9 reglas cubren todas las combinaciones posibles de 3 niveles de temperatura \times 3 niveles de precipitacion. Agregar mas reglas no aportaria informacion adicional.
Cual es la fecha optima que encontro el sistema?	La fecha tipica encontrada esta entre el dia 140-160 (mediados de mayo a inicios de junio), que coincide con el inicio de la temporada de lluvias.
Se podria aplicar a otros cultivos?	Si, ajustando las reglas difusas (temperaturas optimas diferentes) y el ciclo de cultivo (duracion), se puede adaptar a otros cultivos como frijol, calabaza, etc.
Por que NASA POWER y no datos de CONAGUA?	NASA POWER ofrece datos satelitales con cobertura global, sin huecos, y con una API accesible. CONAGUA tiene datos mas localizados pero con mas valores faltantes.
Que pasa si el algoritmo genetico encuentra un optimo local?	Por eso implementamos tambien PSO. Al obtener resultados similares con ambos algoritmos, validamos que el optimo es robusto y no un artefacto del metodo.

— FIN DEL GUION —

Exito en su presentacion! ♡