
SISTEMA DE OPTIMIZACIÓN DE ESTACIONES BASE EN AGRICULTURA DE PRECISIÓN MEDIANTE AGRUPAMIENTO DE PATRONES Y VISUALIZACIÓN CON GRAPHVIZ

202400195 – José Fernando Ramírez Ambrocio

Resumen

Este proyecto desarrolla una solución de software orientada a objetos para optimizar el uso de estaciones base en sistemas de agricultura de precisión. Mediante el análisis de matrices de frecuencias obtenidas de sensores de suelo y cultivo, se transforman en matrices de patrones para agrupar estaciones con comportamientos similares. El sistema, desarrollado en Python sin el uso de estructuras nativas como listas o diccionarios, implementa listas enlazadas anidadas para representar los datos. La aplicación procesa archivos XML de entrada, genera un archivo de salida con estaciones reducidas y utiliza Graphviz para visualizar matrices de frecuencias, patrones y reducidas. La solución cumple con los principios de programación orientada a objetos, permite la toma de decisiones basada en datos y optimiza recursos en tiempo real. Los resultados demuestran una reducción efectiva en el número de estaciones base necesarias, mejorando la eficiencia del sistema agrícola.

Palabras clave

Agricultura de precisión, listas enlazadas, patrones, Graphviz, POO, XML.

Abstract

This project develops an object-oriented software solution to optimize the use of base stations in precision agriculture systems. By analyzing frequency matrices obtained from soil and crop sensors, they are transformed into pattern matrices to group stations with similar behaviors. The system, developed in Python without using native structures such as lists or dictionaries, implements nested linked lists to represent data. The application processes input XML files, generates an output file with reduced stations, and uses Graphviz to visualize frequency, pattern, and reduced matrices. The solution complies with object-oriented programming principles, enables data-driven decision-making, and optimizes resources in real time. Results show an effective reduction in the number of required base stations, improving agricultural system efficiency.

Keywords

Precision agriculture, linked lists, patterns, Graphviz, OOP, XML.

Introducción

La agricultura moderna enfrenta el reto de aumentar la productividad alimentaria en un contexto de cambio climático, escasez de recursos hídricos y crecimiento poblacional. La agricultura de precisión surge como una respuesta tecnológica que permite monitorear, medir y gestionar la variabilidad espacial y temporal en los campos de cultivo, optimizando el uso de insumos como agua, fertilizantes y energía. Este enfoque se basa en la integración de sensores, estaciones base, plataformas en la nube y algoritmos de análisis de datos. Sin embargo, una de las limitaciones más críticas es la infraestructura de comunicación, ya que cada estación base representa un costo operativo y energético. Este proyecto aborda este problema mediante una solución de software que optimiza el uso de estaciones base a través del agrupamiento por patrones de frecuencia. El sistema, desarrollado en Python bajo el paradigma de Programación Orientada a Objetos (POO), procesa datos de sensores almacenados en archivos XML, transforma las matrices de frecuencia en matrices de patrones y agrupa las estaciones con comportamientos idénticos. La solución cumple con las restricciones del curso: no se usan listas, diccionarios ni tuplas de Python, sino listas enlazadas creadas por el estudiante. Además, se utiliza Graphviz para generar visualizaciones claras de las matrices $F[n,s]$, $Fp[n,s]$ y $Fr[n,s]$, facilitando la toma de decisiones. Este trabajo no solo cumple con los requisitos académicos, sino que también representa una aplicación práctica de estructuras de datos avanzadas en un contexto real de ingeniería agrícola.

Desarrollo del tema

a. Arquitectura del Sistema y Programación Orientada a Objetos

La solución se desarrolló completamente bajo el paradigma de Programación Orientada a Objetos (POO), cumpliendo con la restricción del proyecto de no usar estructuras nativas de Python como listas, diccionarios o tuplas. Se crearon clases como `Nodo`, `ListaSimple`, `Campo`, `Estacion`, `SensorS`, `SensorT` y

Frecuencia, que permiten modelar la realidad del campo agrícola de forma jerárquica y modular. Esta arquitectura facilita la escalabilidad, el mantenimiento y la claridad del código. Cada campo agrícola contiene listas enlazadas de estaciones y sensores, y cada sensor contiene sus frecuencias de transmisión asociadas a estaciones.

b. Procesamiento de Datos con Listas Enlazadas

Para cumplir con la restricción de no usar listas de Python, se implementó una lista enlazada simple desde cero. Esta estructura permite insertar, recorrer y obtener elementos sin depender de `list[]`. La lista anidada (`ListaSimple` de `ListaSimple`) se utilizó para representar la matriz de frecuencias $F[n,s]$, donde cada fila es una estación y cada columna es un sensor. Este enfoque garantiza el cumplimiento de los requisitos del proyecto y refuerza el entendimiento de estructuras de datos fundamentales.

c. ¿Por qué listas enlazadas?

Las listas enlazadas son estructuras de datos lineales en las que cada elemento (nodo) contiene un dato y una referencia al siguiente nodo. A diferencia de las listas estáticas o dinámicas de Python (que son arreglos dinámicos), las listas enlazadas permiten una gestión dinámica de la memoria y operaciones eficientes de inserción y eliminación. En este proyecto, el uso de listas enlazadas no es solo una restricción técnica, sino una herramienta pedagógica para entender cómo se construyen estructuras de datos desde cero.

d. Transformación a Matrices de Patrones y Agrupamiento

El proceso de optimización comienza con la matriz de frecuencias $F[n,s]$, que se transforma en la matriz de patrones $Fp[n,s]$, donde cada valor se convierte en 1 si hay frecuencia o 0 si no hay. Las estaciones con el mismo patrón se agrupan, reduciendo el número de estaciones base necesarias. Por ejemplo, si `e01` y `e03` tienen el mismo patrón `[1,1,0]`, se agrupan y se representa con una sola estación (`e01`), sumando sus

frecuencias. Este agrupamiento mejora la eficiencia del sistema al reducir redundancias.

e. Visualización con Graphviz

La herramienta Graphviz se utilizó para generar gráficas de las matrices F[n,s], Fp[n,s] y Fr[n,s] en formato de tabla. Cada gráfica muestra claramente las relaciones entre estaciones y sensores, con colores que destacan valores cero (blanco) y no cero (rosado). Esta visualización permite al usuario comprender rápidamente el estado del campo agrícola y el resultado del procesamiento.

f. ¿Por qué Graphviz?

Graphviz es una herramienta de código abierto para visualizar gráficos definidos por texto. Permite generar diagramas claros y profesionales a partir de estructuras de datos complejas. En este proyecto, se utiliza para generar tablas HTML-like que representan las matrices del sistema.

F[n,t] - MATRIZ DE FRECUENCIAS CULTIVO
Campo: Campo agrícola 01

e01	3500
e02	2000
e03	1000
e04	950
e05	3200

Dimensión: 5 x 1

Figura 4. Matriz de frecuencia del cultivo

Fuente: elaboración propia.

APENDICE

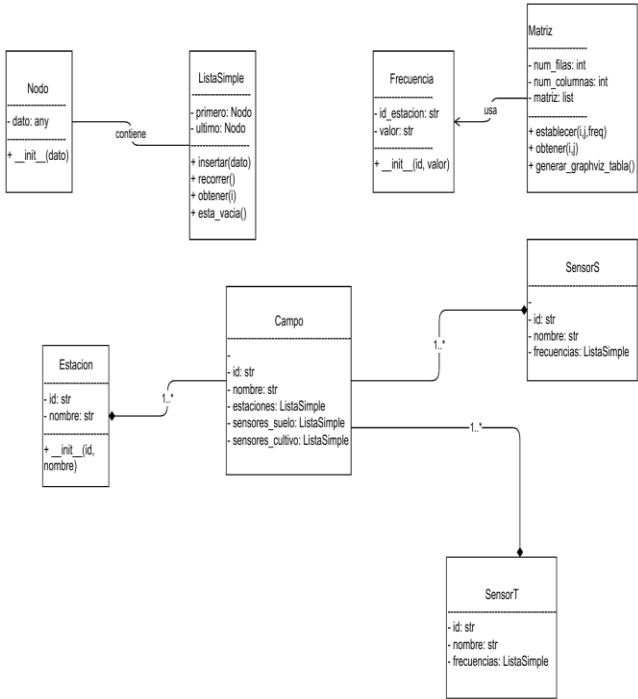


Figura 1. Diagrama de clases del sistema

Fuente: elaboración propia.

Estación\Sensor	s01	s02	s03
e01	700	8300	0
e02	0	0	8000
e04	1500	0	1500

Figura 2. Matriz reducida del suelo

Fuente: elaboración propia.

F[n,s] - MATRIZ DE FRECUENCIAS SUELO
Campo: Campo agrícola 01

e01	200	300	0
e02	0	0	6000
e03	500	8000	0
e04	1500	0	1500
e05	0	0	2000

Dimensión: 5 x 3

Figura 3. Matriz de frecuencia del suelo

Fuente: elaboración propia.

Referencias bibliográficas

Date, C. J. (1991). *An introduction to Database Systems*. Addison-Wesley Publishing Company, Inc.

Python Software Foundation. (2025). *Python 3.13 Documentation*. <https://docs.python.org/3/>

Graphviz. (2025). *Graphviz - Graph Visualization Software*. <https://graphviz.org/>

Conclusion

1. El sistema desarrollado cumple con todos los requisitos del proyecto: uso de POO, listas enlazadas, procesamiento de XML y visualización con Graphviz.
2. La metodología de agrupamiento por patrones permite reducir significativamente el número de estaciones base, optimizando recursos sin perder información.
3. La visualización de matrices mediante Graphviz facilita la interpretación de datos para el usuario final.
4. El proyecto demuestra que soluciones complejas pueden implementarse con estructuras de datos básicas, reforzando los fundamentos de la programación.
5. Esta solución puede extenderse a múltiples campos y servir como base para sistemas de agricultura inteligente más avanzados.