
Proyecto 1

201807086 – Nery José Barrientos Posadas

Resumen

El proyecto se centra en desarrollar una solución integral utilizando Programación Orientada a Objetos (POO) en Python para comprimir señales de audio, enfocándose en los parámetros clave de frecuencia y amplitud. El Centro de Investigaciones de la Facultad de Ingeniería aborda este desafío NP-Hard mediante una metodología de agrupamiento. Esto implica la creación de una matriz de tiempo, amplitud y frecuencias para las señales de audio, que se transforma en una matriz de patrones de frecuencia. Luego, las tuplas con patrones similares se agrupan. El proyecto utiliza archivos XML como insumos y visualiza los tipos de datos abstractos (TDA) con Graphviz. Este enfoque interdisciplinario busca mejorar la compresión de señales de audio de manera eficiente.

Palabras clave

POO, TDA, LISTAS, NODOS, LISTA SIMPLE

Abstract

The project focuses on developing a comprehensive solution using Object Oriented Programming (OOP) in Python to compress audio signals, focusing on the key parameters of frequency and amplitude. The Faculty of Engineering Research Center addresses this NP-Hard challenge using a clustering methodology. This involves creating a matrix of time, amplitude and frequencies for audio signals, which is transformed into a matrix of frequency patterns. Then, tuples with similar patterns are clustered. The project uses XML files as inputs and visualizes the abstract data types (ADTs) with Graphviz. This interdisciplinary approach aims to improve the compression of audio signals in an efficient way.

Keywords

POO, TDA, LIST, NODES, SIMPLE LIST

Introducción

En respuesta al desafío planteado por el Centro de Investigaciones de la Facultad de Ingeniería para comprimir señales de audio, se ha desarrollado una solución integral basada en Programación Orientada a Objetos (POO) y aprovechando herramientas como Graphviz y el análisis de archivos XML. Esta solución utiliza estructuras de programación secuenciales, cíclicas y condicionales para procesar y comprimir señales de audio, centrándose en los parámetros fundamentales de frecuencia y amplitud. Se ha aplicado una metodología de agrupamiento para transformar la información de frecuencias en patrones binarios y, posteriormente, agrupar las tuplas con patrones idénticos. La solución permite una visualización clara de los resultados utilizando Graphviz. Con esta implementación, se aborda eficazmente el desafío de comprimir señales de audio de manera eficiente y con una alta demanda de recursos.

Desarrollo del tema

El desafío presentado por el Centro de Investigaciones de la Facultad de Ingeniería se centró en la compresión de señales de audio, un problema complejo y costoso en términos de tiempo. La solución propuesta abordó este problema desde una perspectiva de Programación Orientada a Objetos (POO) en Python, aprovechando estructuras de programación secuenciales, cíclicas y condicionales. El objetivo fundamental era la compresión de información de señales de audio basada en dos parámetros clave: frecuencia y amplitud.

Para lograrlo, se diseñó una estructura de datos que representara estas señales, utilizando clases y objetos. Cada señal de audio se modeló como un objeto, con atributos que incluían tiempo, amplitud y frecuencias. La implementación de una matriz de patrones binarios permitió reducir la complejidad del problema. Las

frecuencias se convirtieron en patrones binarios que representaban qué tiempos y amplitudes tenían frecuencias en la señal de audio. Luego, se agruparon las tuplas con patrones binarios idénticos.

La visualización de estos resultados se abordó mediante la herramienta Graphviz, que generó representaciones visuales claras y comprensibles de las señales de audio comprimidas. Además, se implementó la capacidad de procesar archivos XML como entrada, lo que permitió una mayor flexibilidad y adaptabilidad a diferentes conjuntos de datos de señales de audio.

En resumen, la solución implementada utilizó una combinación de POO, análisis de estructuras de datos y visualización gráfica para abordar el desafío de la compresión de señales de audio. El resultado es una solución integral que puede manejar eficazmente grandes conjuntos de datos de señales de audio, reduciendo la demanda de recursos y el tiempo necesario para este proceso complejo. Esta implementación es un enfoque eficiente y escalable para la compresión de señales de audio en el ámbito de la investigación de ingeniería.

Conclusiones

Eficiencia en la Compresión de Señales de Audio:

La implementación de la solución en Python, utilizando estructuras de Programación Orientada a Objetos (POO), permitió una compresión eficiente de señales de audio. El uso de matrices de patrones binarios facilitó la reducción del tiempo y los recursos necesarios para el procesamiento de grandes conjuntos de datos de señales de audio, lo que es esencial en problemas NP-Hard como este.

Flexibilidad y Adaptabilidad: La capacidad de procesar archivos XML como entrada proporcionó una mayor flexibilidad y adaptabilidad a diferentes tipos de datos de señales de audio. Esto significa que la solución puede aplicarse a una variedad de

situaciones y conjuntos de datos, lo que la hace útil en diversos escenarios de investigación y aplicación práctica.

Visualización Clara y Comprensible: La integración de la herramienta Graphviz para la visualización de los resultados de compresión en forma de gráficos mejoró significativamente la comprensión de los patrones de frecuencia y amplitud en las señales de audio. Esto es esencial para la interpretación de los resultados y puede ayudar a los investigadores y profesionales a tomar decisiones informadas basadas en la compresión de señales de audio.

En resumen, el código y la solución desarrollada proporcionan una respuesta efectiva y eficiente para el desafío de la compresión de señales de audio, destacando la importancia de la POO, la flexibilidad de entrada y la visualización de datos en la resolución de problemas complejos de investigación en ingeniería de señales.

Referencias bibliográficas

Serrano Montero, M. (s.f.). Estructura de datos.
Universidad de Valladolid, Departamento de
Informática, Campus de Segovia. URL:
<https://www.infor.uva.es/~mserrano/EDI/cap2.pdf>

Anexos

