ISSN 1870-4069

Comparación del nivel de precisión de los clasificadores Support Vector Machines, k Nearest Neighbors, Random Forests, Extra Trees y Gradient Boosting en el reconocimiento de actividades infantiles utilizando sonido ambiental

Diego M. Blanco-Murillo[1], Antonio García-Domínguez[1], Carlos E. Galván-Tejada[1], José M. Celaya-Padilla[2]

[1] Universidad Autónoma de Zacatecas, Unidad Académica de Ingeniería Eléctrica, México

[2] CONACyT - Universidad Autónoma de Zacatecas, Unidad Académica de Ingeniería Eléctrica, Zacatecas, México

{diegomurillo,antonio.garcia,ericgalvan}@uaz.edu.mx,jose.celaya@uaz.edu.mx

Resumen. La información de audio desempeña un papel importante en el creciente contenido digital disponible hoy en día, de tal manera que esto, da como resultado la necesidad de desarrollar sistemas o aplicaciones que analicen de forma automática dicho contenido. Algunas de las aplicaciones más comunes en esta área son: reconocimiento de eventos de audio para domótica y sistemas de vigilancia automática, reconocimiento de voz, recuperación de información musical, análisis multimodal, reconocimiento de actividades humanas, entre otras más en el área de la Inteligencia Ambiental y la Inteligencia Artificial. Sin embargo, los estudios en algunas de las áreas anteriormente mencionadas son escasos, principalmente en el área de reconocimiento de actividades humanas mediante el sonido ambiental. Es por ello, que en este trabajo se realiza una evaluación y comparación del rendimiento de los clasificadores Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET) y Gradient boosting (GB) aplicados al reconocimiento de actividades realizadas por infantes en el rango de edad de 12 a 36 meses.

Palabras clave: Reconocimiento de actividades infantiles,sonido ambiental, support vector machines, K nearest neighbors, random forests, extra trees, gradient boosting.

Precision Level Comparison of the Support Vector Machines, K Nearest Neighbors, Random Forests, Extra Trees and Gradient Boosting

Diego M. Blanco-Murillo, Antonio García-Domínguez, Carlos E. Galván-Tejada, et al.

Classifiers in the Recognition of Children's Activities Using Environmental Sound

Abstract. Audio information plays an important role in the growing digital content that is available nowadays, in this manner, this leads to an outcome that shows the necessity of developing systems or applications that automatically analyze this content. Some of the most common applications in this area are: recognition of audio events for home automation and automatic surveillance systems, voice recognition, music information retrieval, multimodal analysis, recognition of human activities, among others in the area of Environmental Intelligence and Artificial Intelligence. Nevertheless, studies in some of the areas mentioned above are scarce, mainly in the area of recognition of activities through environmental sound. It is therefore, in this paper, an evaluation and comparison of the performance of the Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET) and Gradient boosting (GB) classifiers applied to the recognition of activities carried out by infants in the age range of 12 to 36 months is made.

Keywords: Child activity recognition, environmental sound, support vector machines, K nearest neighbors, random forests, extra trees, gradient boosting.

1. Introducción

El análisis, clasificación y predicción automática de actividades humanas es un tema de gran interés en diferentes áreas de la inteligencia artificial, tanto por su dificultad, como por sus aplicaciones. El reconocimiento de actividades puede ser la base para que sistemas realicen tareas complejas, por ejemplo, sistemas para dar seguimiento a pacientes, cuidado de ancianos, como el presentado por Jalal, Kamal y Kim en [1], cuidado de infantes, sistemas de rehabilitación, entrenamiento físico, vigilancia inteligente, como el presentado por Mun Sim, Lee y Ohbyung Kwon en [2], robots inteligentes, entre muchas otras.

Existen muchos trabajos orientados al cuidado de infantes que se inclinan por utilizar aparatos o dispositivos colocados directamente sobre el cuerpo de la persona, entre estos se encuentran sistemas portátiles, acelerómetros, dispositivos de radiofrecuencia, sensores barométricos, como los presentados por Onkar y Agrawal en [3], Anusha, Belagali, Maheendrachari y Prashant en [4], Boughorbel, Breebaart, Bruekers, Flinsenberg y Kate en [5] y Nam y Park en [6], que se enfocan en el monitoreo de grupos de niños con la finalidad de evitar un accidente y salvaguardar la integridad física de los mismos. Por el contrario, existen pocos trabajos que se basan en utilizar el sonido ambiental como fuente de datos para reconocer actividades de niños, como el presentado por García-Domínguez y Galván-Tejada en [7].

La orientación de los trabajos desarrollados hasta hoy en día sobre reconocimiento de actividades en niños resultan ser completamente invasivos, ya que se

centran en utilizar sensores colocados directamente sobre el cuerpo de la persona o en alguna prenda que lleve puesta, provocando que éstos puedan interferir con las actividades que los niños realizan, obteniendo datos imprecisos que no permitan realizar un análisis coherente y detallado de la actividad a analizar.

El enfoque de utilizar sonido como fuente de datos para reconocer y clasificar las actividades resulta no invasivo y más cómodo para los infantes, ya que el dispositivo de grabación puede estar a una distancia considerablemente alejada para captar un audio de buena calidad y sin intervenir en las actividades realizadas por el niño.

En la actualidad, existen varios clasificadores de datos que son aplicados al reconocimiento de actividades, entre los cuales se destacan: Naive Bayes (NB), Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET), Gradient boosting (GB) y Neural Networks (NN). Un ejemplo de estos clasificadores, es la utilización de Random Forest (RF) y Neural Networks (NN), como el presentado por Carlos Galván-Tejada, Jorge Galván-Tejada, Celaya-Padilla, Delgado-Contreras, Magallanes-Quintanar, Martinez-Fierro, Garza-Veloz, López-Hernández y Gamboa-Rosales en [8], con el objetivo de realizar un análisis de las características de los audios para desarrollar un modelo de reconocimiento de actividades humanas.

El objetivo de este estudio es evaluar el nivel precisión de los clasificadores SVM, kNN, RF, ET, GB, a la hora de catalogar grabaciones de audio de actividades comunes en niños de 12 a 36 meses de edad, tales como caminar, correr, jugar y llorar.

Tabla 1. Descripción de las actividades.

Actividad	**Descripción**
Caminar	Recorrer a pie determinada distancia a velocidad media
Correr	Recorrer a pie con rapidez una distancia determinada
Jugar	Manipular bloques de plástico de manera que éstos produzcan ruido al golpearse
Llorar	Producir un sonido de llanto como reacción a algún suceso

2. Materiales y métodos

En la presente sección se describe la extracción de características que se realizó con *pyAudioAnalysis* para clasificar audios de actividades infantiles, así como los métodos y clasificadores utilizados para llevar a cabo el presente estudio.

2.1. Descripción del data-set

El Data-Set que se utiliza para este trabajo, está conformado por grabaciones de audio de cuatro actividades realizadas por niños en el rango de edad de 12

a 36 meses, tales como, caminar, correr, jugar y llorar. La Tabla 1 muestra la descripción de cada una de las actividades que fueron consideradas en el análisis. **Fuente de los archivos de audio.** Para establecer el Data-Set, los audios de las diferentes actividades, fueron recolectados directamente del estudio realizado sobre el Reconocimiento de actividades infantiles utilizando sonido ambiental [7], y descargados directamente de una página web [9].

2.2. Procesamiento de audio

Para realizar procesamiento de audio *pyAudioAnalysis* [10] es una biblioteca abierta de Python que proporciona una amplia gama de funcionalidades relacionadas con el audio que se centran en problemas de extracción, clasificación, segmentación y visualización de características.

2.3. Extracción de características (análisis a corto plazo)

La señal de audio primero se divide en ventanas de corto plazo (cuadros) y para cada cuadro se calculan las 34 características mencionadas en la Tabla 2. Esto da como resultado una secuencia de vectores de características a corto plazo de 34 elementos cada uno. Los tamaños de ventana de corto plazo ampliamente aceptados son de 20 a 100 milisegundos. En *pyAudioAnalysis*, el proceso a corto plazo se puede llevar a cabo utilizando superposición de encuadre, es decir, el paso del cuadro es más corto que la longitud del cuadro, o no superposición de encuadre, es decir, el paso del cuadro es igual a la longitud del cuadro.

2.4. Modelo de clasificación

Una parte importante en el reconocimiento y clasificación de actividades humanas es el clasificador que se utiliza para catalogar dichas actividades. Cuando se analizan señales de audio, como es el caso del presente trabajo, la utilización de un clasificador adecuado para el análisis de las muestras ayuda a obtener un procesamiento adecuado, preciso y confiable.

Los modelos de clasificación utilizados para este trabajo implementan un procedimiento de validación cruzada para estimar el parámetro clasificador óptimo, la elección de estos modelos se hizo tomando en cuenta la forma en que estos modelos dividen la señal de audio en ventanas de corto plazo (cuadros) y después calculan una serie de características para cada cuadro. Este proceso conduce a una secuencia de vectores de características a corto plazo para toda la señal.

2.5. Clasificadores

En este trabajo, se utilizaron cinco clasificadores diferentes, Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET), Gradient boosting (GB). Una descripción detallada de los clasificadores utilizados se puede encontrar en esta sección.

Tabla 2. Características de audio.

Índice	**Nombre**	**Descripción**
1	Tasa de cruce cero	La tasa de cambio de signo de la señal durante la duración de un cuadro particular.
2	Energía	La suma de cuadrados de los valores de señal, normalizados por la longitud respectiva.
3	Entropía de energía	La entropía de las energías normalizadas de los subfotogramas. Se puede interpretar como una medida de cambios abruptos.
4	Centroide espectral	El centro de gravedad del espectro.
5	Extensión espectral	El segundo momento central del espectro.
6	Entropía espectral	Entropía de las energías espectrales normalizadas para un conjunto de subfotogramas.
7	Flujo espectral	La diferencia cuadrada entre las magnitudes normalizadas de los espectros de los dos cuadros sucesivos.
8	Desplazamiento espectral	La frecuencia por debajo de la cual se concentra el 90 % de la distribución de la magnitud del espectro.
9-21	CCFM	Los coeficientes cepstrales de la frecuencia Mel forman una representación cepstral donde las bandas de frecuencia no son lineales sino que se distribuyen de acuerdo con la escala de mel.
22-33	Vector de Croma	Una representación de 12 elementos de la energía espectral en la que los contenedores representan las 12 clases de tono temperamental de música de tipo occidental (espaciado de semitonos).
34	Desviación cromática	La desviación estándar de los 12 coeficientes de croma.

Support Vector Machines (SVM). El clasificador SVM es un algoritmo de aprendizaje supervisado basado en kernel que clasifica los datos en dos o más clases. SVM está especialmente diseñado para la clasificación binaria. Durante la fase de capacitación, SVM construye un modelo, mapea el límite de decisión para cada clase y especifica el hiperplano que separa las diferentes clases. Aumenta la distancia entre las clases, incrementando el margen del hiperplano con la finalidad de ayudar a la precisión de la clasificación. SVM se puede utilizar para realizar de manera efectiva la clasificación no lineal.

Como se mencionó anteriormente, el clasificador SVM es un clasificador basado en kernel. Una función Kernel es un procedimiento de mapeo realizado en el conjunto de entrenamiento para mejorar su semejanza con un conjunto de datos linealmente separable. El objetivo del mapeo es aumentar la dimensionalidad del conjunto de datos y se realiza de manera eficiente utilizando una función kernel. Algunas de las funciones de kernel comúnmente utilizadas son lineales, RBF (del inglés radial basis function), cuadráticas, kernel Perceptron multicapa y kernel polinomial.

k Nearest Neighbors (k-NN). En el reconocimiento de patrones, el algoritmo K-NN es un método de aprendizaje basado en instancias utilizado para clasificar objetos en función de sus ejemplos de entrenamiento más cercanos en el espacio de características. Un objeto se clasifica por el voto mayoritario de sus vecinos, es decir, el objeto se asigna a la clase que es más común entre sus k vecinos más cercanos, donde k es un número entero positivo. En el algoritmo k-nn, la clasificación de un nuevo vector de características de prueba está determinada por las clases de sus k-vecinos más cercanos.

Random forests (RF). El clasificador Random forests (RF) es un meta estimador que se adapta a una serie de clasificadores de árbol de decisiones en varias submuestras del conjunto de datos y utiliza un promedio para mejorar la precisión predictiva y controlar el ajuste excesivo. El tamaño de submuestra siempre es el mismo que el tamaño de muestra de entrada original.

En Random Forests, cada árbol en el conjunto se construye a partir de una muestra extraída con reemplazo, es decir, una muestra de arranque del conjunto de entrenamiento. Además, al dividir un nodo durante la construcción del árbol, la división que se elige ya no es la mejor división entre todas las características. En cambio, la división que se selecciona es la mejor división entre un subconjunto aleatorio de las características. Como resultado de esta aleatoriedad, el sesgo del bosque generalmente aumenta ligeramente (con respecto al sesgo de un solo árbol no aleatorio) pero, debido al promedio, su varianza también disminuye, generalmente más que compensando el aumento en el sesgo, lo que arroja un modelo global mejor.

Extra trees (ET). Extra trees, implementa un meta estimador (igual que Random forests) que se adapta a una serie de árboles de decisión aleatoria en varias submuestras del conjunto de datos y utiliza promedios para mejorar la precisión predictiva y controlar el ajuste excesivo.

El módulo *sklearn.ensemble* incluye dos algoritmos de promedio basados en árboles de decisión aleatoria: el algoritmo RandomForest y el algoritmo Extra-Trees. Ambos algoritmos son técnicas de perturbación y combinación diseñadas específicamente para árboles. Esto significa que se crea un conjunto diverso de clasificadores introduciendo aleatoriedad en la construcción del clasificador. La predicción del conjunto se da como la predicción promedio de los clasificadores individuales.

Gradient boosting (GB). Gradient boosting (GB) construye un modelo aditivo de forma progresiva; permite la optimización de funciones de pérdida diferenciables arbitrarias. En cada etapa, los árboles de n_classes_regression se ajustan al gradiente negativo de la función de pérdida de desviación binomial o multinomial. La clasificación binaria es un caso especial en el que solo se induce un solo árbol de regresión.

Gradient Tree Boosting o Gradient Boosted Regression Trees (GBRT) es una generalización de impulsar a las funciones de pérdida diferenciables arbitrarias.

GBRT es un procedimiento comercial preciso y efectivo que se puede usar tanto para problemas de regresión como de clasificación. Los modelos de Gradient Tree Boosting se utilizan en una variedad de áreas, incluidas la clasificación de búsqueda web y la ecología.
Las ventajas de GBRT son:

- Manejo natural de datos de tipo mixto (características heterogéneas)
- Poder predictivo
- Robustez a valores atípicos en el espacio de salida (a través de funciones de pérdida robustas)

Las desventajas de GBRT son:

- Escalabilidad, debido a la naturaleza secuencial de impulsar, difícilmente se puede paralelizar.

El módulo *sklearn.ensemble* proporciona métodos para la clasificación y la regresión a través de árboles de regresión potenciados por el gradiente.

3. Experimentación

Se utilizaron un total de 70 grabaciones, de las cuales 43 se utilizaron para realizar el entrenamiento del modelo, con el fin de obtener un entrenamiento robusto y preciso, y las restantes 27, para realizar pruebas de precisión a cada clasificador, dicha cantidad se consideró aceptable para no estresar al modelo, probandolo con una cantidad menor de grabaciones en comparación con la cantidad de grabaciones con las que se había creado. Las cantidades de archivos de audio que se tiene para cada actividad, y que se utilizan en cada una de las etapas (entrenamiento y pruebas) se muestran en las Tablas 3 y 4 respectivamente.

Tabla 3. Archivos de audio por actividad para entrenamiento.

Actividad	**Archivos de audio**
Caminar	11
Correr	11
Jugar	10
Llorar	11

El proceso de entrenamiento y pruebas fue llevado a cabo con la ayuda del lenguaje de programación Python, que es un lenguaje de programación de alto nivel que ha estado atrayendo un interés creciente, especialmente en la comunidad académica y científica durante los últimos años y en el cual se encuentra implementada la librería *pyAudioAnalysis*, que cubre una amplia gama de tareas de análisis de audio, tales como extraer características de audio, entrenar y aplicar clasificadores de audio, segmentar una secuencia de audio utilizando metodologías supervisadas o no supervisadas y visualizar relaciones

Tabla 4. Archivos de audio por actividad para pruebas.

Actividad	Archivos de audio
Caminar	7
Correr	7
Jugar	6
Llorar	7

de contenido. Para el análisis de los archivos, se trabajó con los clasificadores Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET) y Gradient boosting (GB) de la librerías de Python.

Con el propósito de crear un modelo preciso, cada clasificador fue entrenado aisladamente a partir de los datos de la Tabla 3, es decir, el conjunto de archivos de audio de dicha tabla fue colocado en carpetas separadas con el nombre de cada clasificador.

Los pasos a realizar en el proceso de entrenamiento son los siguientes:

- La extracción de características de cada audio (para este trabajo sería la extracción de características a corto plazo).
- La evaluación del clasificador y selección de parámetros.
- Obtener el parámetro clasificador óptimo.
- Almacenar el entrenamiento del modelo.

Además, también se crea un archivo ARFF (con el mismo nombre que el modelo), donde se almacena todo el conjunto de vectores de características y etiquetas de clase respectivas.

La obtención de este archivo se realizo con la funcion *featureAndTrain()* incluida en la librería *pyAudioAnalysis*.

La fase de pruebas para cada clasificador se realizó de manera separada para evitar una posible confusión en los resultados. La forma de comenzar el proceso de prueba para cada clasificador inicia cargando el clasificador, es decir el archivo que se generó en la etapa de entrenamiento, en seguida se leen los archivos de audio y se convierten en sonido mono, los dos últimos pasos consisten en realizar la extracción de características a corto plazo y la clasificación de la actividad de cada audio.

4. Resultados

Fueron un total de cinco pruebas las que se realizaron, una prueba por cada clasificador con el conjunto de datos de la Tabla 4.

En la Tabla 5 se pueden observar los datos que arrojaron las pruebas que se le aplicaron a cada modelo a cerca de la precisión con la que se llevó a cabo la clasificación de las actividades analizadas en el presente trabajo.

Observando la Tabla 5, se puede comprobar que los clasificadores k Nearest Neighbors (kNN) y Extra trees (ET) fueron los más acertados a la hora de

Tabla 5. Resultados.

	Clasificadores				
	SVM	**kNN**	**RF**	**ET**	**GB**
Actividad	**Porcentaje de precisión**	**Porcentaje de precisión**	**Porcentaje de precisión**	**Porcentaje de precisión**	**Porcentaje de precisión**
Caminar	40.31 %	100 %	76.50 %	100 %	99.93 %
Correr	55.53 %	100 %	71.50 %	100 %	99.95 %
Jugar	74.94 %	100 %	84.00 %	100 %	99.95 %
Llorar	41.93 %	100 %	73.50 %	100 %	99.95 %

clasificar un archivo de audio, obteniendo un 100 % de precisión, seguido por el clasificador Gradient boosting (GB) que obtuvo una precisión del 99.94 % en promedio. Por el contrario, se puede observar que los clasificadores Support Vector Machines (SVM) y Random forests (RF) clasifican a los archivos de audio con una precisión del 53.17 % y 76.37 % en promedio respectivamente.

5. Discusión y conclusiones

El enfoque principal de este trabajo de investigación es realizar un estudio comparativo sobre distintos clasificadores utilizados en el área de reconocimiento y clasificación de actividades humanas mediante sonido ambiental, específicamente en actividades realizadas por infantes de 12 a 36 meses (considerados biológicamente como bebés). Los clasificadores tomados en cuenta en el presente trabajo son: Support Vector Machines (SVM), k Nearest Neighbors (kNN), Random forests (RF), Extra trees (ET) y Gradient boosting (GB). De este análisis se puede discutir y concluir lo siguiente:

- Actividades como caminar y correr pudieron ser clasificadas de forma correcta por los clasificadores ya que contienen suficiente información para diferenciar actividades que generan un sonido ambiental similar.
- k Nearest Neighbors (kNN) y Extra trees (ET) resultaron ser los clasificadores más precisos a la hora de evaluar el archivo de audio de las diferentes actividades.
- En contraste con lo anterior, Support Vector Machines (SVM) y Random forests (RF) que resultaron ser los clasificadores menos acertados en esta experimentación.

No obstante, se tiene la inquietud de que un Data-Set más grande sería de bastante ayuda para el clasificador a la hora de su entrenamiento, permitiendo así obtener un modelo de estimación de actividades infantiles altamente preciso.

6. Trabajo futuro

Como trabajo futuro, más actividades habitualmente realizadas por infantes de 12 a 36 meses serán agregadas al Data-Set y se realizarán más pruebas con

los clasificadores presentados con anterioridad a dichas actividades, de manera adicional se propone los siguientes puntos específicos:

- Aplicar otros clasificadores de datos para hacer una comparación y lograr obtener el mejor clasificador enfocado al reconocimiento y clasificación de actividades realizadas por infantes.
- Agregar más actividades realizadas por infantes.
- Optimizar la clasificación de las actividades realizadas por infantes.

Referencias

1. Jalal, A., Kamal, S., Kim, D.: A Depth Video Sensor-Based Life-Logging Human Activity Recognition System for Elderly Care in Smart Indoor Environments. Sensors (14248220), 11735–11759 (2014)
2. Mun Sim, J., Lee, Y., Kwon O.: Acoustic Sensor Based Recognition of Human Activity in Everyday Life for Smart Home Services. International Journal of Distributed Sensor Networks, 1–11 (2015)
3. Onkar Nehete, J., Agrawal, D.G.: Real time Recognition and monitoring a Child Activity based on smart embedded sensor fusion and GSM technology. The International Journal Of Engineering, 35–40 (2015)
4. Anusha, A.M., Belagali R., Mahendrachari, Prashant: Child Activity Recognition Using Accelerometers. In: NCRIET-2015 & Indian, 007–012 (2015)
5. Boughorbel, Sabri, Breebaart, Jeroen, Bruekers, Fons, Flinsenberg, Warner Ten, K.: Child-activity recognition from multi-sensor data. In: Proceedings of the 7th International Conference on Methods and Techniques in Behavioral Research - MB 10, (2010)
6. Nam, Y., Wook Park, J.: Physical activity recognition using a single triaxial accelerometer and a barometric sensor for baby and child care in a home environment. Ambient Intelligence & Smart Environments, 381–402 (2013)
7. García-Domínguez, A., Galván-Tejada, C.E.: Reconocimiento de actividades infantiles utilizando sonido ambiental: Un enfoque preliminar. Research in Computing Science 139 (2017), 71–79 (2017)
8. Galván-Tejada, C.E., Galván-Tejada, J.I., Celaya-Padilla, J.M., Delgado-Contreras, J.R., Magallanes-Quintanar, R., Martinez-Fierro, M.L., Garza-Veloz, I., López-Hernández, Y., Gamboa-Rosales, H.: An Analysis of Audio Features to Develop a Human Activity Recognition Model Using Genetic Algorithms, Random Forests, and Neural Networks. Mobile Information Systems, 1–10 (2016)
9. Freesound, `https://freesound.org/`
10. Giannakopoulos, T.: pyAudioAnalysis: An Open-Source Python Library for Audio Signal Analysis. PloS one (2015)