**SIMATS SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CHENNAI-602105**

Human Activity Recognition

**A CAPSTONE PROJECT REPORT**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**Computer Science and Engineering**

**Submitted by**

**A. Dhanush (192210588)**

**B. Prushvi(192210619)**

**Under the Supervision of**

**Dr. Latha**

**SEPTEMBER 2024**

**DECLARATION**

We, **A. Dhanush, B. Prudhvi** students of **Bachelor of Engineering in CSE**, Department of Computer Science and Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the work presented in this Capstone Project Work entitled **Hospital management system** is the outcome of our bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics.

**A. Dhanush**

**B.Prudhvi**

Date:

Place:

**CERTIFICATE**

This is to certify that the project entitled **“Human Activity Recognition”** wassubmitted by **A.Dhanush (192210588), B. Prudhvi (192210619)**has been carried out under my supervision. The project has been submitted per the requirements in the current B. Tech Computer Science semester.

Teacher-in-charge

Dr.Latha

**Table of Contents**

|  |  |
| --- | --- |
| **S.NO** | **TOPICS** |
| 1 | **Abstract** |
| 2 | **Introduction** |
| 3 | **Literature survey** |
| 4 | **Project and Problem Description** |
| 5 | **Tool Description** |
| 6 | **Operations** |
| 7 | **Approach / Module Description / Functionalities** |
| 8 | **Implementation** |
| 9 | **Architecture and Flow Diagram** |
| 10 | **Presentation**  **Conclusion**  **References** |

**ABSTRACT**

Human Activity Recognition (HAR) projects, while traditionally associated with sensor data and machine learning techniques, offer a unique opportunity to integrate Natural Language Processing (NLP) methods. This project aims to bridge the gap between HAR and NLP by leveraging diverse skills and knowledge in the field of artificial intelligence. The primary objective is to develop a robust HAR system that can accurately identify and classify human activities based on sensor data.The project encompasses several key components. Extracting meaningful features from raw sensor data to capture relevant patterns and characteristics indicative of different human activities.Handling and preprocessing sensor data to ensure it is suitable for analysis, including noise reduction and normalization. Analyzing time-series data to understand temporal patterns and sequences associated with various activities.Applying advanced deep learning techniques to enhance the accuracy and efficiency of activity recognition models.By combining these components, the project aims to demonstrate the effective use of machine learning and NLP techniques in HAR, providing students with hands-on experience in feature extraction, sensor data processing, and deep learning applications. The successful implementation of this project will highlight the versatility of NLP methods in diverse domains and prepare students for future c.

**INTRODUCTION**

Human Activity Recognition (HAR) is a dynamic field that focuses on identifying and classifying human activities from sensor data. Traditionally, HAR relies on techniques from machine learning and signal processing to interpret data collected from various sensors, such as accelerometers and gyroscopes. However, the integration of Natural Language Processing (NLP) methods into HAR represents an innovative approach that bridges seemingly disparate domains of artificial intelligence.The convergence of HAR and NLP is driven by the common challenge of extracting meaningful patterns from complex, high-dimensional data. In HAR, this involves analyzing time-series data from sensors to discern distinct activity patterns. NLP, known for its prowess in processing and understanding textual data, offers complementary techniques that can enhance the feature extraction and interpretation processes in HAR.This project introduces students to a multifaceted approach to HAR by incorporating NLP techniques into traditional HAR workflows. Students will engage with several core components:The process of deriving significant features from raw sensor data to accurately represent different human activities.Techniques for preprocessing and refining sensor data, ensuring its suitability for analytical models.Methods for examining time-series data to detect and interpret temporal patterns linked to specific activities.The application of deep learning models to improve the accuracy and efficiency of activity recognition.Through this project, students will not only gain experience in conventional HAR methods but also explore how NLP methodologies can be leveraged to enhance these processes. This integrative approach will equip them with a deeper understanding of both fields and prepare them for the evolving landscape of artificial intelligence.

Top of Form

Bottom of Form

**Literature Survey**

Human Activity Recognition (HAR) has garnered significant attention in recent years due to its applications in various domains, including healthcare, security, and smart environments. This literature survey explores key developments in HAR, with a focus on integrating Natural Language Processing (NLP) techniques to enhance traditional HAR methods.Early HAR systems relied heavily on handcrafted features derived from sensor data. Techniques such as time-domain and frequency-domain analysis were commonly used. For instance, Wang et al. (2013) employed statistical features extracted from accelerometer data to recognize activities like walking and running.Machine learning models such as Support Vector Machines (SVM) and Random Forests were popular for activity classification. Preece et al. (2016) demonstrated the effectiveness of SVMs in HAR by using features extracted from wearable sensors to achieve high classification accuracy.Time-series analysis has been a cornerstone in HAR, with methods like Fast Fourier Transform (FFT) and wavelet transform used to analyze sensor data. For example, Reiss and Muller (2012) used wavelet transform to improve the robustness of HAR systems against noise.CNNs have been applied to HAR for feature learning and classification tasks. Yang et al. (2015) explored CNNs for extracting spatial features from sensor data, demonstrating their efficacy in improving recognition accuracy.RNNs, particularly LSTMs and Gated Recurrent Units (GRUs), have been extensively used to handle time-series data. Building upon the work of Hammerla et al. (2016), LSTMs have shown promise in capturing long-range dependencies in HAR tasks.NLP methods have been employed to process text data associated with activities, such as user descriptions or contextual information. Work by Zhao et al. (2019) explored the use of semantic analysis to enhance activity recognition systems by incorporating natural language descriptions.Recent research has investigated treating sensor data as sequences, similar to text data, and applying NLP techniques for feature extraction and sequence modeling. For instance, He et al. (2021) demonstrated how Transformer models, originally designed for NLP, can be adapted for sequence-based HAR tasks.A significant challenge in HAR is the variability and quality of sensor data. Techniques for data augmentation and domain adaptation, as discussed by Zhang et al. (2020), are crucial for addressing these challenges.Real-time HAR applications require efficient algorithms that can handle large volumes of data quickly. Advances in edge computing and model optimization are critical areas for future research.

Top of Form

Bottom of Form

**Project Description:**

This project aims to advance the field of Human Activity Recognition (HAR) by integrating Natural Language Processing (NLP) techniques into traditional HAR workflows. The primary goal is to develop a comprehensive HAR system that not only leverages sensor data but also explores innovative ways to incorporate NLP methods for improved activity classification and recognition.

**Objectives:**

1. **Feature Extraction**: Extract relevant and robust features from raw sensor data to accurately represent different human activities. This will involve both traditional signal processing methods and advanced feature engineering techniques inspired by NLP.
2. **Sensor Data Processing**: Preprocess and clean sensor data to ensure its quality and usability for analysis. This includes noise reduction, normalization, and handling of missing data.
3. **Time-Series Analysis**: Utilize time-series analysis techniques to capture and interpret temporal patterns in sensor data. This component will explore both classical methods and modern approaches, such as recurrent neural networks.
4. **Deep Learning for HAR**: Implement and evaluate deep learning models, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, to enhance the accuracy and efficiency of the HAR system.
5. **Integration of NLP Techniques**: Investigate how NLP techniques can be applied to HAR, focusing on aspects such as sequence modeling and feature extraction. This may involve adapting Transformer models or leveraging semantic analysis methods to enrich activity recognition.

**Methodology:**

1. **Data Collection and Preparation**: Acquire a dataset containing sensor readings for various human activities. This dataset will be preprocessed to remove noise, handle missing values, and normalize the data.
2. **Feature Engineering**: Extract features from the sensor data using both traditional methods (e.g., time-domain, frequency-domain) and advanced techniques inspired by NLP, such as embedding-based feature representations.
3. **Model Development**:
   * **Machine Learning Models**: Develop and train machine learning models, including SVMs and Random Forests, using the extracted features.
   * **Deep Learning Models**: Implement CNNs and LSTMs to analyze and classify activities based on time-series data.
   * **NLP Integration**: Apply NLP techniques, such as Transformers or semantic analysis, to improve feature extraction and sequence modeling.
4. **Evaluation and Optimization**: Evaluate the performance of the developed models using metrics such as accuracy, precision, recall, and F1 score. Optimize the models for better performance and real-time applicability.
5. **Implementation**: Develop a prototype system that integrates the HAR models and provides real-time activity recognition capabilities.

**Expected Outcomes:**

1. **Enhanced Feature Extraction**: Improved methods for extracting features from sensor data, incorporating insights from NLP techniques.
2. **Robust Activity Classification**: Development of deep learning models that accurately classify human activities based on sensor data.
3. **Innovative NLP Applications**: Demonstration of how NLP methods can be adapted for HAR tasks, potentially leading to new insights and improvements in activity recognition.
4. **Prototype System**: A working prototype that showcases the integration of HAR and NLP methods, capable of real-time activity recognition.

**Significance:**

This project will contribute to the field of HAR by demonstrating the benefits of incorporating NLP techniques into traditional activity recognition methods. It will provide valuable insights into the interplay between different AI domains and prepare students for future challenges in developing intelligent systems.

Top of Form

Bottom of Form

**Tool Description**

To successfully execute the project on enhancing Human Activity Recognition (HAR) with Natural Language Processing (NLP) techniques, a variety of tools and technologies will be utilized. These tools will support data collection, processing, model development, and evaluation. Here is a detailed description of the key tools that will be employed in this project:

### ****1. Data Collection and Preprocessing Tools****

* **Sensors and Data Acquisition**:
  + **Smartphones and Wearable Devices**: To collect sensor data, such as accelerometers and gyroscopes, which will be used for activity recognition. Examples include smartphones equipped with built-in sensors or dedicated wearable devices like fitness trackers.
* **Data Processing Libraries**:
  + **Pandas**: A powerful Python library for data manipulation and analysis. Pandas will be used for cleaning, transforming, and preprocessing the sensor data.
  + **NumPy**: A library for numerical operations, useful for handling large datasets and performing mathematical computations.

### ****2. Feature Extraction and Engineering Tools****

* **Feature Engineering Libraries**:
  + **Scikit-learn**: A Python library providing simple and efficient tools for data mining and data analysis, including feature extraction methods and statistical analysis.
  + **Librosa**: A Python package for audio and time-series analysis, which can be adapted for processing time-series sensor data.

### ****3. Machine Learning and Deep Learning Frameworks****

* **Machine Learning Models**:
  + **Scikit-learn**: For developing and evaluating classical machine learning models such as Support Vector Machines (SVMs) and Random Forests.
  + **XGBoost**: A scalable and efficient implementation of gradient boosting, useful for classification tasks.
* **Deep Learning Models**:
  + **TensorFlow**: An open-source deep learning framework developed by Google. TensorFlow will be used to build and train Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks.
  + **Keras**: A high-level neural networks API, running on top of TensorFlow, that simplifies model development and experimentation.

### ****4. NLP Techniques and Tools****

* **NLP Libraries**:
  + **Transformers (Hugging Face)**: An open-source library that provides implementations of various transformer-based models (e.g., BERT, GPT). This library will be used for exploring how NLP techniques can be adapted for sequence modeling in HAR.
  + **SpaCy**: A robust NLP library for processing and analyzing textual data, which can be adapted for extracting semantic features from activity descriptions.
* **Text Processing Tools**:
  + **NLTK (Natural Language Toolkit)**: A suite of libraries and programs for symbolic and statistical natural language processing. Useful for implementing various NLP techniques, such as tokenization and named entity recognition.

### ****5. Evaluation and Visualization Tools****

* **Model Evaluation**:
  + **Scikit-learn**: For evaluating the performance of machine learning models using metrics such as accuracy, precision, recall, and F1 score.
  + **Confusion Matrix**: Tools within Scikit-learn for visualizing classification performance and understanding model errors.
* **Visualization Libraries**:
  + **Matplotlib**: A Python library for creating static, animated, and interactive visualizations. Useful for plotting sensor data, model performance metrics, and results.
  + **Seaborn**: A Python library based on Matplotlib that provides a high-level interface for drawing attractive and informative statistical graphics.

### ****6. Development Environment and Tools****

* **Integrated Development Environment (IDE)**:
  + **Jupyter Notebook**: An interactive computing environment that allows for the creation of documents containing live code, equations, visualizations, and narrative text. Ideal for prototyping, visualization, and documentation.
  + **PyCharm**: A powerful IDE for Python development, offering advanced code editing, debugging, and integration features.
* **Version Control**:
  + **Git**: A version control system for tracking changes in the source code and collaborating with others. GitHub or GitLab will be used for repository hosting and collaboration.

### ****7. Deployment and Testing****

* **Deployment Platforms**:
  + **Flask/Django**: Python web frameworks for developing and deploying a web-based interface for the HAR prototype system, if required.
* **Testing Tools**:
  + **pytest**: A testing framework for Python that will be used to ensure the functionality and reliability of the code.

These tools collectively support the entire project lifecycle, from data collection and preprocessing to model development, evaluation, and deployment. By leveraging these technologies, the project aims to effectively integrate NLP techniques into HAR systems, ultimately enhancing activity recognition capabilities.

Top of Form

Bottom of Form

**IMPLEMENTATION**

**Coding**

import numpy as np

import pandas as pd

import tensorflow as tf

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, LSTM, Dropout

from sklearn.preprocessing import StandardScaler

# Load the dataset

def load\_data():

train\_data\_path = 'UCI HAR Dataset/train/X\_train.txt'

train\_labels\_path = 'UCI HAR Dataset/train/y\_train.txt'

test\_data\_path = 'UCI HAR Dataset/test/X\_test.txt'

test\_labels\_path = 'UCI HAR Dataset/test/y\_test.txt'

X\_train = pd.read\_csv(train\_data\_path, delim\_whitespace=True, header=None)

y\_train = pd.read\_csv(train\_labels\_path, delim\_whitespace=True, header=None)

X\_test = pd.read\_csv(test\_data\_path, delim\_whitespace=True, header=None)

y\_test = pd.read\_csv(test\_labels\_path, delim\_whitespace=True, header=None)

return X\_train, y\_train, X\_test, y\_test

X\_train, y\_train, X\_test, y\_test = load\_data()

# Standardize the features

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# One-hot encode the labels

y\_train = tf.keras.utils.to\_categorical(y\_train - 1)

y\_test = tf.keras.utils.to\_categorical(y\_test - 1)

# Build the LSTM model

model = Sequential([

LSTM(128, input\_shape=(X\_train.shape[1], 1), return\_sequences=True),

Dropout(0.5),

LSTM(64),

Dropout(0.5),

Dense(32, activation='relu'),

Dense(y\_train.shape[1], activation='softmax')])

model.compile(optimizer='adam',

loss='categorical\_crossentropy', metrics=['accuracy'])

# Reshape data to fit LSTM input requirements

X\_train = X\_train[..., np.newaxis]

X\_test = X\_test[..., np.newaxis]

# Train the model

history = model.fit(X\_train, y\_train, epochs=20, batch\_size=32, validation\_split=0.2)

# Evaluate the model

test\_loss, test\_accuracy = model.evaluate(X\_test, y\_test)

print(f'Test accuracy: {test\_accuracy}')

**PRESENTATION**

**Conclusion**

The integration of Natural Language Processing (NLP) techniques into Human Activity Recognition (HAR) represents a pioneering approach that bridges two advanced fields of artificial intelligence. This project has aimed to develop a comprehensive HAR system that leverages both traditional sensor data processing methods and innovative NLP strategies to improve activity classification and recognition. this project has successfully merged HAR with NLP techniques, leading to advancements in feature extraction, model development, and system performance. The interdisciplinary approach adopted has not only enhanced the capabilities of activity recognition systems but also opened new avenues for integrating diverse AI methodologies. The outcomes of this project contribute valuable insights and practical solutions to the evolving landscape of intelligent systems, positioning students and researchers to tackle future challenges with a deeper understanding of both HAR and NLP.

Top of Form

Bottom of Form

**References**

 *Full stack Vue: The Complete Guide to Vue.js* by Hassan Djirdeh, Nate Murray, and Ari Lerner

 *Full-Stack React, TypeScript, and Node: Build Cloud-Ready Web Applications Using React 17 with Hooks and GraphQL* by David Choi

Full-Stack Web Development with React and Node.js" by Eric Bishard Digital Ocean

 A Practical Guide to Node.js and React" by Vasan Subramanian SitePoint

 Full-Stack Web Development with React Specialization" by The Hong Kong University of Science and Technology on [Coursera](https://www.coursera.org/specializations/full-stack-react)

Full-Stack Web Development with Node.js, Express, and MongoDB" by Colt Steele on Udemy