



Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**



Grado en Ingeniería Informática

Trabajo Fin de Grado

**Aplicación de Reconocimiento de
Imágenes para Monitorización No
Intrusiva de Carga Eléctrica**

Autor: Hernán Calvo Aguiar
Tutor(a): Esteban García Cuesta

Madrid, Junio 2024

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado
Grado en Ingeniería Informática

Título: Aplicación de Reconocimiento de Imágenes para Monitorización
No Intrusiva de Carga Eléctrica

Junio 2024

Autor: Hernán Calvo Aguiar

Tutor: Esteban García Cuesta

Departamento de Inteligencia Artificial (DIA)

Escuela Técnica Superior de Ingenieros Informáticos

Universidad Politécnica de Madrid

Resumen

Aquí va el resumen del trabajo. Extensión máxima 2 páginas.

La Monitorización No Intrusiva de Carga, es una técnica que busca administrar los consumos de diferentes dispositivos eléctricos sin la necesidad de medir directamente cada uno de ellos.

El objetivo de este proyecto es desarrollar la pieza clave de este sistema. Una Inteligencia Artificial que permita identificar los dispositivos conectados y consumiendo a una red eléctrica, como podría ser una casa, un restaurante o una oficina.

El modelo a diseñar deberá ser capaz de identificar los dispositivos conectados a la red eléctrica, y estimar el tiempo de uso de cada uno. Para ello, se ha decidido hacer uso de CSPNet, un modelo potente, computacionalmente barato de entrenar

Este trabajo

Abstract

Abstract of the Final Degree Project. Maximum length: 2 pages.

Tabla de contenidos

1. Introducción	1
1.1. Motivación y Aplicaciones	1
1.2. Objetivos	2
1.3. Alcance del proyecto	2
1.4. Estructura de la Memoria	3
2. Fundamentos y Estado del Arte	5
2.1. Conceptos generales	5
2.2. Aprendizaje Automático	7
2.3. Redes Neuronales	9
2.3.1. Sparse Coding	9
2.3.2. Clasificadores de imágenes	10
2.3.3. Redes Recurrentes	10
2.4.	10
2.4.1. CSPNet	10
2.4.2. Codificación GAF	10
2.4.3. Validación de Modelos: NILMTK	10
3. Desarrollo	11
3.1. Planificación	11
3.2. REFIT Dataset	11
3.2.1. Pros y contras del dataset	11
3.2.2. Características principales	11
3.2.3. Recogida de datos	11
3.3. Parseado de REFIT a MYSQL	12
3.3.1. Motivación	12
3.3.2. Inconvenientes	12
3.3.3. Solución	12
3.4. Codificación GAF	12
3.4.1. Justificación del uso de OpenCL	12
3.4.2. Inconvenientes	12
3.4.3. Solución	12
3.5. Entrenamiento	12
3.5.1. ZLUDA y Darknet Framework	12
3.5.2. Monitorización	12
3.5.3. Inconvenientes	12

TABLA DE CONTENIDOS

4. Resultados y Análisis de Impacto	13
4.1. Resultados	13
4.2. Impacto	13
4.2.1. Personal	13
4.2.2. Empresarial	13
4.2.3. Económico	13
4.2.4. Medioambiental	13
5. Conclusiones y trabajo futuro	15
5.1. Conclusiones	15
5.2. Trabajo a Futuro	15
5.3. Alegato Final	15
Bibliografía	17
Anexos	19
A. Glosario	19
B. Fórmulas Matemáticas	21
B.1. Cadenas de Markov y Modelos Ocultos de Markov	21
B.2. Modelos Factoriales Ocultos de Markov	21
B.3. Sparse Coding	21
C. Repositorios	23
C.1. Código Fuente	23
C.2. Repositorios externos	23

Capítulo 1

Introducción

1.1. Motivación y Aplicaciones

Las técnicas de Monitorización No Intrusiva de Carga (en inglés NILM o Non Intrusive Load Monitoring) extraen el consumo de electrodomésticos a partir de una curva de consumo agregada. [NILMreview2017]. Dicho de otra forma. El monitoreo de carga no intrusivo es un sistema que infiere los consumos de diferentes dispositivos sobre una curva de consumo en la toma de corriente principal. Por ejemplo en la toma de corriente de una casa.

En Europa, un tercio de la energía eléctrica se ha generado mediante energías no renovables(12 % o 333TWh carbón, 17 % o 569TWh gas natural)[energyeurope][ember2024european]. Aún siendo un descenso récord, para poder continuar reduciendo el consumo de energías no renovables, debemos desarrollar herramientas que nos permitan hacer un uso más inteligente de la generación de energía. Herramientas como la monitorización de carga no intrusiva pueden ayudarnos a detectar patrones de uso[nilmstateoftheheart], para entonces poder desarrollar políticas que permitan optimizar la generación de energía.

Si, hipotéticamente, se redujese un 2 % la generación de electricidad a través de la combustión de carbón. Supondría una reducción de emisiones de 96.6 Millones de Toneladadas de CO₂¹.

Realizar un monitoreo de carga no intrusivo es un problema complejo. Actualmente el área de soluciones NILM se encuentra en la fase de investigación y desarrollo. Por lo tanto. Mi objetivo para este trabajo es entrenar y analizar el rendimiento de una inteligencia artificial especializada en la clasificación de imágenes, transformando los datos de entrenamiento de series temporales a imágenes. Detallado en el **Capítulo 3**. El trabajo se centrará en cómo modificar el enfoque al problema de disgregación de consumos afecta al rendimiento de la técnica en comparación con otras técnicas del estado del arte.

Este concepto fue inventado por George W. Hart, Ed Kern y Fred Schweppe en

¹ $333 * 10^3 \text{GWh} * 290 \frac{\text{tCO}_2}{\text{GWh}}$ [postcarboneyelectricidad]

Capítulo 1. Introducción

el Instituto Tecnológico de Massachussets, en los años ochenta [192069]. Fundados por el Electric Power Research Institute, podemos encontrar el proceso básico descrito en la patente estadounidense 4,858,141.

Una breve bibliografía de George W. Hart: Actualmente ejerce como escultor desde 2017. Tiene esculturas en exposición en Berkeley, Princeton, Cambridge, Duke, Brown... Cofundó el único Museo de las Matemáticas en Norteamérica y se ha retirado recientemente, trabajó en ciencia computacional, matemáticas, educación e investigación [HartBibliography].

La técnica, descrita por Hart, Kern y Schweppe en [192069] modela los consumos en lo que nombran como "Modelo Total de Carga". El modelo total de carga depende de que dispositivos estén encendidos en un momento dado. Por lo que definen un 'proceso de cambio'. Esto nos lleva a un problema computacionalmente imposible de resolver salvo mediante la fuerza bruta[192069] y poco margen de mejora en el modelado para simplificar su resolución. De manera resumida, el problema trata en encontrar el número de dispositivos y el consumo de cada uno. Este es un problema que se beneficia de hacer uso de modelos heurísticos.

Siguiendo este objetivo y con el asesoramiento de Esteban García Cuesta, se decidió en hacer uso de datasets clasificados para un entrenamiento supervisado; además de una arquitectura potente, compacta y computacionalmente baja en costes de entrenamiento.

extender motivación y aplicaciones, introducir con más detalle el problema y citar más motivaciones

1.2. Objetivos

Previamente al comienzo del desarrollo del proyecto, se establecieron una serie de objetivos a cumplir, dividiendo el objetivo final en pasos coherentes y concretos. En esta sección se detallan los objetivos de aprendizaje e investigación:

Este trabajo busca investigar sobre las técnicas más actuales en NILM². Dando al lector una base de las técnicas utilizadas en el estado del arte y las matemáticas detrás de estas técnicas. Los objetivos de aprendizaje son extensos, el trabajo busca además profundizar en la práctica detrás de el entrenamiento de una inteligencia artificial y de los retos que presenta el uso de la computación paralela, bases de datos, GPUs, lenguajes de bajo nivel (C++, OpenCL) y el uso de frameworks de creación, entrenamiento y análisis del rendimiento de la inteligencia artificial.

1.3. Alcance del proyecto

Al finalizar este proyecto, se busca tener un modelo capaz de realizar una monitorización de carga no intrusiva, además de una serie de progamas y herramien-

²ver Apéndice A

tas útiles a la hora de realizar trabajos futuros en el área.

1.4. Estructura de la Memoria

La memoria de esta investigación tiene los siguientes capítulos.

- **Introducción:** Se introduce el tema, sus potenciales aplicaciones, motivación, alcance...
- **Fundamentos y Panorama Actual:** Este capítulo sirve para dar al lector una base de la teoría detrás de la tecnología, algoritmos y conceptos sobre los que se desarrolla el trabajo de fin de grado. Además de informar sobre el estado del arte en el área.
- **Desarrollo:** En este capítulo se detallan los pasos tomados para el desarrollo del modelo y las herramientas que utiliza. Entrenamiento, codificación, evaluación del modelo, etc.
- **Resultados y Análisis de Impacto:** Este apartado recoge los resultados del modelo y se compara su rendimiento con otros modelos. Además recoge el análisis del impacto del modelo conforme a los Objetivos de Desarrollo Sostenible de la Agenda 2030.
- **Conclusiones y Trabajo Futuro:** En este capítulo se recogen las conclusiones del trabajo y el trabajo futuro que puede hacerse para mejorar el modelo y sus partes.
- **Bibliografía:** Se recogen todos los recursos bibliográficos citados durante el transcurso de la investigación.
- **Anexo:** Detalla código e información de interés que no tiene cabida en el cuerpo del texto.

Capítulo 2

Fundamentos y Estado del Arte

Capítulo dedicado a describir los fundamentos y el panorama actual del trabajo.

2.1. Conceptos generales

El concepto de NILM desde su introducción ha sido el método preferido por ingenieros e investigadores para la disgregación de consumo eléctrico desde su introducción, debido a sus ventajas económicas y prácticas.[Nalmpantis2019]

En cuanto al origen de este método, se buscó categorizar en [192069] los dispositivos dentro de tres grandes grupos, para poder estructurar los diferentes dispositivos y generalizar el modelado de dispositivos específicos. Estos grupos son:

1. **Señales de estados fijos** Siguen la arquitectura de una máquina de estados finita, las transiciones entre estados son consumos exactos, al igual que los estados
2. **Señales transitorias** Esto son señales que no se encuentran en un espacio discreto que permita modelar su comportamiento como una máquina de estados. Un ejemplo podría ser un calefactor automático, que regula su generación de calor en función de la diferencia de la temperatura ambiente y la temperatura objetivo.
3. **Otras** Señales que no pueden categorizarse en ninguno de los dos casos anteriores. Por ejemplo, los cambios de dirección del motor de una lavadora en el proceso de lavado.

Cabe mencionar que los propios investigadores reconocieron que en el futuro los dispositivos eléctricos que siguiesen un diseño de máquinas de estados perderían precedencia. Ya que habría más 'dispositivos inteligentes' [192069]. Actualmente, una calefacción programable o un aire acondicionado programable están al alcance de cualquiera. En 2021, en España el 50 % de las viviendas en alquiler constan de aire acondicionado[idealista2021].

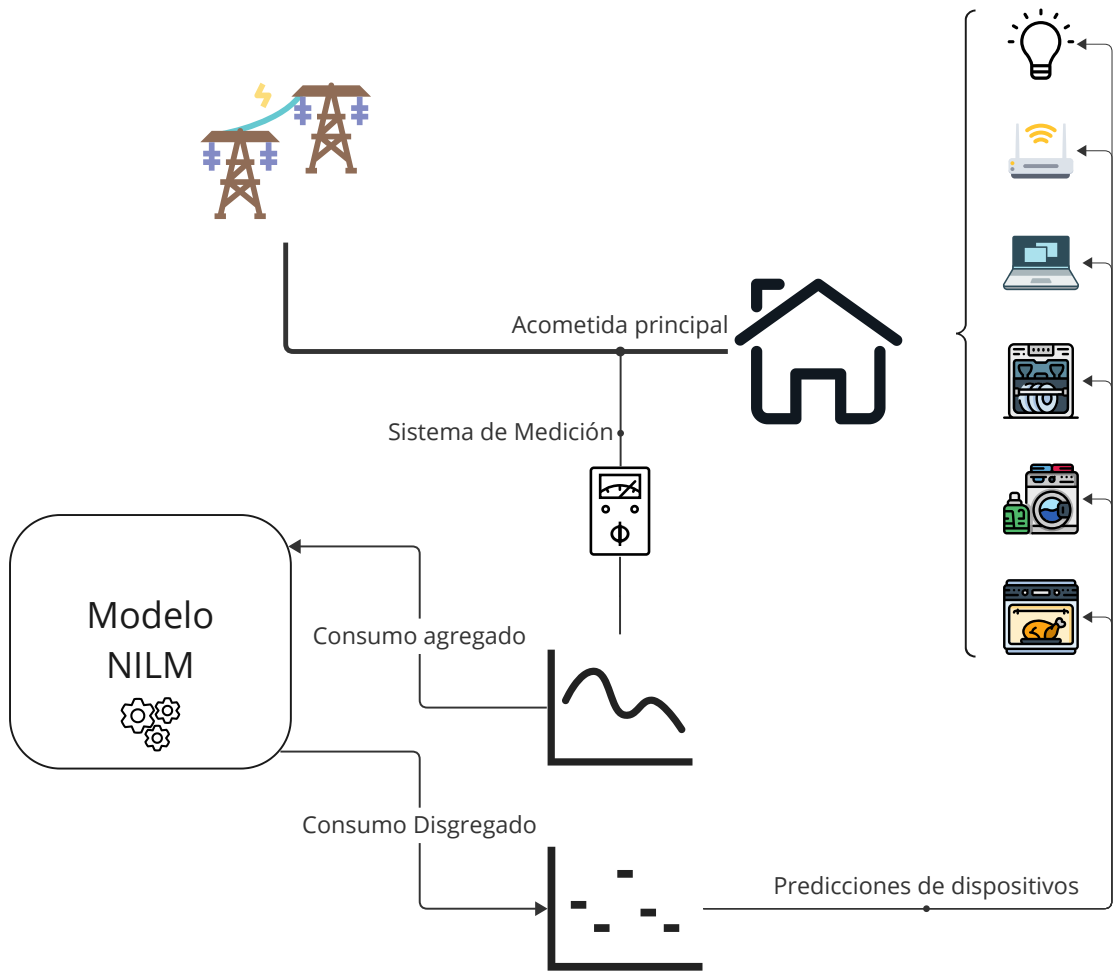


Figura 2.1: Un diagrama simple de un sistema NILM

Y esto es sólo un dispositivo. Habrá cientos de modelos de cada dispositivo, decenas de marcas, cada dispositivo con sus características específicas. Un modelado exhaustivo sería prácticamente imposible de crear y mantener, ya que sólo para el mercado doméstico hay una cantidad desorbitante de dispositivos, combinaciones de dispositivos, ect.

Un ejemplo de sistema NILM

Un sistema NILM consta de las siguientes partes: Una fuente de datos (acometida principal). Un dispositivo de recogida de datos (puede ser de Potencia Activa Y reactiva, Voltaje e Intensidad, ect). El modelo predictivo. Y el sistema de consulta de datos, que se ha omitido en la Figura 2.1.

Debido a estos argumentos, la investigación de la monitorización no intrusiva se inclinó a los modelos heurísticos y el aprendizaje automático.

2.2. Aprendizaje Automático

En la Historia de la Inteligencia Artificial, se han definido una serie de etapas de gran crecimiento y financiación como "veranos"(tres concretamente). Alternados por (dos) inviernos"; donde el interés, la financiación y el número de publicaciones decrecían significativamente [**briefAIHistory**]. En los principios de la década de los 2010, comenzó el tercer "verano" de la Inteligencia Artificial. En 2012 ocurre un avance técnico: unos años antes, Olga Russakovsky creó el reto ImageNet [**ImageNetRussakovsky**]; un repositorio de más de un millón de imágenes de objetos a lo largo de más de mil categorías diferentes. Este reto estimuló a la comunidad científica,

Resulta lógico que el número de artículos científicos proponiendo modelos que permitiesen resolver los retos que presenta NIML aumentase.

De los que voy a destacar los Modelos Ocultos de Markov, el Sparse Coding y finalmente las Redes Neuronales. Siendo estas últimas el foco principal del trabajo, y de este capítulo.

Si se sabe la teoría detrás de una cadena de markov, los modelos de Markov Ocultos resultan una dirección intuitiva a tomar, ya que estos representan la probabilidad de transición de un estado a otro. En esta subsección daremos una breve introducción a la teoría detrás de este modelo probabilístico.

La Cadena de Markov

Formalmente, una cadena de Markov se representa por los siguientes componentes:

- Un conjunto Q de N estados:

$$Q = q_1 q_2 \dots q_n$$

- Una matriz de probabilidad de transición A donde para cada a_{ij} representando la probabilidad de transicionar desde el estado j al estado i .

$$A = a_{11} a_{12} \dots a_{N1} \dots a_{NN}$$

Cumpliendo A la propiedad: $\sum_{j=1}^n a_{ij} \forall i$

- Una distribución de probabilidad inicial π sobre los estados. Siendo π_i la probabilidad de que la cadena de Markov comience en el estado i .

$$\pi = \pi_1, \pi_2, \dots, \pi_N$$

Si un estado j tuviera $\pi_j = 0$ no podría ser un estado inicial (teniendo probabilidad 0).

Además, π debe cumplir $\sum_{i=1}^N \pi_i = 1$

[**markovStandford**]

Identificando Datos Ocultos Con Cadenas de Markov

Una cadena de Markov resulta útil cuando tenemos que modelar un comportamiento en base a una secuencia de eventos observables, siendo L.E. Baum y Ted Petrie los autores de **[introduccionHMM]**, el primero de una serie de papers sobre modelos probabilísticos y estadística que introdujeron los HMM. Estos eventos observables son elementos conocidos, pero la secuencia de estados del modelo para esos eventos observables son desconocidos.

El problema, como en NILM, es que los estados no son directamente observables. Están *ocultos* y por tanto debemos ajustar la cadena para incluir una secuencia de posibles similitudes observadas **[markovStandford]**.

Continuando la explicación de la subsección anterior:

- Para un elemento i se expresa como $b_i(o)$ donde o es derivado de un conjunto de posibles valores observables V .

- Dado:

$$Q = q_1 q_2 \dots q_n$$

- Calculamos la probabilidad de una cadena de estados S de longitud T

$$S = s_1, s_2, \dots, s_T$$

- Dado un conjunto de valores observados

$$O = o_1, o_2, \dots, o_T$$

para T observaciones.

[markovStandford]

Resumiendo: dados una serie de eventos observables, podemos inferir la probabilidad de cada estado que causaron estos eventos observables: Calculando la probabilidad de una cadena de eventos observables dada la probabilidad de una cadena de estados del conjunto, para todas las posibles combinaciones de la cadena los estados del conjunto. Este algoritmo se conoce algoritmo de Viterbi. Siendo la fórmula:

$$\text{[markovStandford]} \arg \max_{O=o_1, o_2, \dots, o_T} P(O = o_1, o_2, \dots, o_T | S = s_1, s_2, \dots, s_T)$$

Este es el modelado que nos permite inferir de manera probabilística (fórmulas en detalle podrán encontrarse en el **Apéndice B**) Cabe añadir que las cadenas de markov tienen una serie de algoritmos para determinar la cadena de estados ocultos más probable para la cadena de eventos observables, se omite por brevedad, pero puede consultarse **[markovStandford]** para una excelente y detallada explicación sobre este área.

se añadirá el
anexo para la
entrega de la
memoria final

Modelos Ocultos de Markov

Explicadas brevemente las bases de los Modelos Ocultos de Markov, pasamos a la aplicación de este modelo estadístico en la disgregación de consumos.

Dado que un modelo de markov oculto infiere probabilísticamente una serie de estados, esto nos resulta útil para la disgregación de dispositivos. Para codificar diferentes estados de diferentes dispositivos se aumenta el número de cadenas del modelo. Teniendo una matriz de estados S_i^j donde S es un conjunto de estados; las observaciones dependen de todos los estados en un momento dado. Podemos observar esto en el siguiente diagrama:

Observando el diagrama b) no resulta muy complicado imaginar la utilidad de este modelo. Si cada cadena representa un dispositivo modelado en diferentes estados, resulta un diseño muy apropiado para NILM. Los Modelos de Markov Ocultos se han tratado en la literatura como efectivos en comparación con otras técnicas de monitorización cuando el entrenamiento supervisado presenta una baja tasa de muestreo en el eje temporal de la obtención de datos [frenchHMMNILM]. Otras ventajas que presenta es su bajo coste computacional en el entrenamiento. Pero el uso de este tipo de elementos lo vuelven altamente susceptible a máximos locales[NILMreview2017].

En el caso de los Modelos Factoriales de Markov Ocultos (con siglas FHMM en inglés), presentan ciertas ventajas en comparación con los HMM ¹: La complejidad y el coste computacional de los algoritmos de aprendizaje e inferencia son menores para los HMM. Sin embargo, tanto FHMM como HMM presentan limitaciones al volumen máximo de dispositivos simultáneos, incluso en casos como [afmap2012] donde se propone un nuevo algoritmo no supervisado. El algoritmo desarrollado (AFMAP) no presenta los problemas de alta complejidad y máximos locales, pero requiere de etiquetación manual después de la disgregación. Y presenta bajo rendimiento para dispositivos electrónicos y de cocina [NILMreview2017].

2.3. Redes Neuronales

En esta sección se tratan tres conceptos de las Redes Neuronales; se dará una introducción y se expondrán las aplicaciones pertinentes al tema del proyecto. Por motivos de brevedad, se asume que el lector tiene una cierta base en inteligencia artificial. En caso contrario, sugiero consultar las citas y el Apéndice A

2.3.1. Sparse Coding

Breve Introducción

Introducido por Olshausen et al., 1997 [OLSHAUSEN]. El Sparse Coding consiste en reducir la activación de la capa oculta de las neuronas, forzando a la red neuronal a realizar un aprendizaje [stanfordSparse]. Su objetivo es encontrar

¹Hidden Markov Models, o en español los Modelos Ocultos de Markov

Capítulo 2. Fundamentos y Estado del Arte

un conjunto de vectores ϕ_i tal que podamos representar un vector x como una combinación lineal de estos vectores:

$$x = \sum_{i=1}^k a_i \phi_i$$

Este es un concepto que está fuertemente relacionado con los autoencoders y el aprendizaje PCA.

Que tienen como objetivo es que su salida sea igual a la entrada, pero limitando la capa oculta para forzar un aprendizaje [**stanfordAutoEncoding**]. Por ejemplo, supongamos que buscásemos utilizar un autoencoder para comprimir una imagen para enviarla por internet. Si no implementásemos restricciones, dado que el objetivo de la red neuronal es reducir la pérdida de información a 0; la salida del autoencoder sería la imagen, completamente idéntica.

Para evitar esto en el caso de los autoencoders se introduce una capa oculta de menor tamaño en comparación a la capa de entrada como restricción para forzar un aprendizaje, al tener una capa oculta, la capa de salida (de igual tamaño a la de entrada) debe reconstruir la entrada [**stanfordAutoEncoding**].

Volviendo al Sparse Coding. Al tener un conjunto de vectores sobrecompleto, los coeficientes a_i ya no se determinan únicamente por el vector de entrada x . Luego se define una función de escasez². Definimos esta función de escasez como: tener los menos componentes (a_i) distintos de cero o el mínimo de componentes no cerca de cero [**stanfordSparse**].

Aplicaciones

Tanto los autoencoders como el sparse coding han sido usados en la literatura NILM. sparse coding:[**SparseCodingNILM**]

Pendiente continuar investigando

2.3.2. Clasificadores de imágenes

2.3.3. Redes Recurrentes

2.4.

2.4.1. CSPNet

2.4.2. Codificación GAF

2.4.3. Validación de Modelos: NILMTK

²De ahí el "Sparse"; escasez en Español

Capítulo 3

Desarrollo

TODOS LOS DIAGRAMAS SE HARAN CON MIRO, EXPORTADOS A SVG E INTEGRADOS

3.1. Planificación

Se realizó dos tandas de organización y planificación de tareas. La primera al principio del semestre y la segunda a mediados de abril. Las dificultades en la planificación fueron una combinación de motivos externos y una perspectiva algo ingenua del tiempo a invertir y el ritmo de progreso. Debido a la falta de conocimientos, se ha dedicado un tiempo considerable (unas 60-80 horas) a realizar investigación y estudio, para poder comprender las técnicas en el estado del arte y en lo

Los principales problemas fueron: la dificultad para programar un codificador en C++ y OpenCL, debido a la curva de dificultad que representa aprender un lenguaje para una nueva arquitectura y los retos que presenta la gran cantidad de datos usada para el entrenamiento de inteligencia artificial.

3.2. REFIT Dataset

3.2.1. Pros y contras del dataset

3.2.2. Características principales

3.2.3. Recogida de datos

Hablar de REFIT, sus características, los datos y los metadatos. Especial hincapié en cómo se recogieron

3.3. Parseado de REFIT a MYSQL

3.3.1. Motivación

3.3.2. Inconvenientes

3.3.3. Solución

Hablar de porqué lo parseamos a mysql, los inconvenientes (zona horaria e interrupciones en el parseo) y el diseño realizado para resolver estos problemas, además de los programas desarrollados (el primero que se rompía y el segundo que es traversecsv.cpp)

3.4. Codificación GAF

3.4.1. Justificación del uso de OpenCL

3.4.2. Inconvenientes

3.4.3. Solución

Hablar de porqué se decidió usar opencl, porqué se retrasó tanto esta parte del proyecto el diseño que era muy bueno pero muy complicado de implementar

3.5. Entrenamiento

3.5.1. ZLUDA y Darknet Framework

3.5.2. Monitorización

3.5.3. Inconvenientes

Apartado pendiente

Capítulo 4

Resultados y Análisis de Impacto

4.1. Resultados

Resultados del modelo, comparación con otros modelos.

4.2. Impacto

4.2.1. Personal

Mayor control de gasto, menos privacidad. Análisis de patrones de gastos y por tanto hábitos personales/de la vivienda

4.2.2. Empresarial

industria: mayor control y más barato. (explicar pq) del gasto energético

4.2.3. Económico

mejora de la gestión energética lleva a mayor eficiencia, optimización y a mejor aprovechamiento de las energías renovables

4.2.4. Medioambiental

pro: aprovechar el consumo/optimizar los patrones de gasto teniendo mayor granularidad y un modelo inferente como el que se ha desarrollado permite construir en un futuro herramientas predictivas de gastos que por tanto pueden aportar info valiosa para modelos más generales (usando estos y otros datos como los meteorológicos, mercantiles, mercados, etc) para estimar la demanda esperada y el volumen de energía a generar.

Se recomienda analizar también el potencial impacto respecto a los Objetivos de Desarrollo Sostenible (ODS), de la Agenda 2030, que sean relevantes para el trabajo realizado (ver enlace)

Capítulo 5

Conclusiones y trabajo futuro

5.1. Conclusiones

5.2. Trabajo a Futuro

5.3. Alegato Final

Anexos

Apéndice A

Glosario

Lista de términos y abreviaciones utilizados en el trabajo. Su sentido es facilitar y familiarizar al lector la comprensión de las abreviaciones.

- NILM: Non Intrusive Load Monitoring - Monitorización de Carga No Intrusiva.
- HMM: Hidden Markov Models - Modelos Ocultos de Markov.
- Sparse Coding: Codificación de escasez.
- CSPNet: Cross Stage Partial Network
- PCA: Principal Component Analysis - Análisis de Componentes Principales

Apéndice B

Fórmulas Matemáticas

B.1. Cadenas de Markov y Modelos Ocultos de Markov

Añadir todas las fórmulas relevantes para poder dejar constancia de la teoría matemática

B.2. Modelos Factoriales Ocultos de Markov

B.3. Sparse Coding

Apéndice C

Repositorios

C.1. Código Fuente

aquí irá mi repositorio, con el readme del repositorio

C.2. Repositorios externos

aquí irán los repositorios que he usado. Actualmente solo he usado el SafeMap