UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática

TFG del Grado en Ingeniería Informática

SmartBeds

Aplicación de técnicas de minería de datos para la detección de crisis epilépticas

Presentado por José Luis Garrido Labrador
en Universidad de Burgos — 15 de junio de 2019
Tutores: Álvar Arnaiz González y José Francisco Díez Pastor

UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática

Dr. D. Álvar Arnáiz González, profesor del departamento de Ingeniería Civil, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. José Luis Garrido Labrador, con DNI 71707244Y, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado "Smart-Beds - Aplicación de técnicas de minería de datos para la detección de crisis epilépticas".

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 15 de junio de 2019

Vº. Bº. del Tutor:	Vº. Bº. del co-tutor:
Dr. D. Álvar Arnaiz González	Dr. D. José Francisco Díez Pastor

Resumen

La epilepsia es una enfermedad que provoca crisis repentinas con convulsiones violentas y pérdidas del conocimiento. Ante estas situaciones es necesario suministrar unos primeros auxilios, sin embargo, durante las sesiones nocturnas es difícil detectar y tratar a tiempo estas crisis que pueden provocar daños graves a los pacientes que sufren esta enfermedad.

En este trabajo se investiga, a partir de datos reales provenientes de sensores en un colchón, métodos que puedan detectar situaciones de crisis en tiempo real de tal manera que se puedan suministrar las atenciones necesarias con la mayor celeridad posible.

Además de esto se ha creado una API REST y una página web con una gestión en tiempo real de camas con el estado actual del paciente y las probabilidades de una situación de crisis.

Descriptores

Crisis epilépticas, minería de datos, desequilibrados, monitorización, clasificador, API REST.

Abstract

Epilepsy is a disease whose symptoms are violent seizures and fainting. In that's situations is necessary to apply first aid, but, overnight is difficult to detect and treat in time these crisis. That can cause serious damage to the patients.

In this paper is researched, based on real data from sensors in a mattress, methods which can detect seizures in real time in such a way that caregivers can supply first aids as soon as possible.

In addition, it has been created a REST API and a web page with real time management of beds with the patient's current state and the seizure probability.

Keywords

Epileptic seizures, data mining, unbalanced, monitoring, clasificator, API REST.

Índice general

Índice de figuras

Índice de tablas

Introducción

Gracias al avance de las técnicas y algoritmos de minería de datos, disciplinas no directamente relacionadas con la computación se han ido beneficiando de las ciencias de datos, a modo particular, la medicina está desarrollándose hacia modelos más preventivos gracias a las predicciones que se pueden generar utilizando estos métodos.

Es por este motivo que podemos ver en la literatura científica de los últimos años como se relaciona medicina y ciencia de datos, por ejemplo en estudios de detección de caídas [27] o la motorización del sueño para la prevención de apneas [13]. En este trabajo de fin de grado, el objetivo es la detección de crisis epilépticas adscrito al proyecto homónimo vencedor del concurso universitario *Desafío Universidad Empresa* [4]

Aunque el análisis y detección de crisis epilépticas de manera automática ha sido ampliamente explorada por la comunidad científica esta se ha centrado o en el uso de pulseras inteligentes [20] o el uso de encefalogramas (*EEG*) [11, 14, 28] para la predicción de estos eventos. Por tanto, ante pacientes con diversos problemas de salud que impiden el uso de pulseras y de dispositivos de control de actividad cerebral se realiza este proyecto que enfoca la detección de las crisis epilépticas en el uso de sensores de presión y biométricos en la cama donde duerme el paciente.

1.1. Material adjunto

Junto a esta memoria se incluyen:

- **Cuaderno de investigación** con la evolución y pasos realizados durante la investigación junto con los resultados de cada experimento.

- **Anexos** donde se incluyen:
  - Plan de proyectos
  - Requisitos del sistema
  - Diseño del sistema
  - Manual para el programador
  - Manual para el usuario
- **API REST** en **Python-Flask**
- **Experimentos** en *Jupyter Notebook*

Además se puede acceder a través de internet a la página web en producción y al repositorio GitHub del proyecto.

Objetivos del proyecto

2.1. Objetivos generales

- Estudio del estado en materia de detección de crisis epilépticas, tanto en *hardware*, datos utilizados y técnicas y modelos ya probados. Con esto se busca explorar técnicas no antes utilizadas como optimizar esfuerzos por los métodos que ya han probado su utilidad.
- Exploración e interpretación de los datos, las formas por las cuales se pueden representar y como se distribuyen las distintas situaciones de crisis respecto a las situaciones normales. Aplicación de filtros, análisis estadísticos, proyecciones de n-variedad.
- Exploración de técnicas de balanceo de datos existentes y aprendizaje automático para conjuntos de datos desequilibrados. Buscar la mejor combinación de técnicas de balanceo y modelos de aprendizaje automático para optimizar la precisión.
- Búsqueda de un modelo que haga una clasificación lo más correcta posible centrando que la predicción sea acertada en situaciones de crisis mediante la optimización del valor del ratio de verdaderos positivos.
- Creación de una *API REST* con la que poder distribuir en tiempo real los datos de cama y sus predicciones.

2.2. Objetivos técnicos

- Hacer uso de las herramientas de minería de datos de *sci-kit learn* y *Weka*.

- Crear una serie de transformadores de datos para facilitar el preprocesado.
- Desarrollar una *API REST* sencilla y fácil de usar.
- Crear una interfaz web que permita ver los datos en tiempo real de los pacientes con los menores márgenes temporales posibles.

2.3. Objetivos personales

- Contribuir a la mejora de la calidad de vida de pacientes que sufran de epilepsia.
- Profundizar en el trabajo de investigador, sus metodologías y las fases de por las que pasa una investigación.
- Comprender más técnicas de minería de datos, nuevos modelos y nuevas formas de abordar análisis de datos.
- Completar mi formación académica con el desarrollo de una aplicación que englobe la mayor cantidad del conocimiento adquirido en el estudio del grado.

Conceptos teóricos

En este capítulo se explicarán superficialmente los conceptos por los cuales se han desarrollado este proyecto, una explicación más completa de los modelos que se han usado se encuentran en el capítulo 5.1.

Definiciones

Crisis epiléptica: se trata de un evento imprevisto de corta duración que comienza y termina de forma súbita. Se originan en el cerebro y pueden provocar convulsiones, rigidez, desvanecimiento o espasmos musculares según el foco de origen del mismo. Si la duración de la crisis fuese de más de cinco minutos se considera *status epilepticus* y puede provocar daños neuronales y es improbable que paren por si solas y se necesita atención médica inmediata para mitigar los efectos. [16]

Preprocesado: proceso por el cual se realizan operaciones sobre los datos con el fin de facilitar la interpretación de los mismos. Engloba procesos como la eliminación de instancias ruidosas, suavizado de señales mediante filtros, normalización de las características, eliminación de variables con una baja variabilidad y el análisis estadístico. [23]

Análisis de componentes principales: se trata de una técnica estadística que se utiliza para crear una descri31pción del conjunto de datos utilizando unas variables no correlacionadas. El objetivo es encontrar los ejes de máxima varianza y realizar proyecciones de menor dimensionalidad de los datos facilitando la interpretación de los datos. [36]

Proyección a 2-variedad: se define como variedad a un objeto geométrico que representa un espacio que se parece localmente, la idea intuitiva sobre este concepto es el de un mapa, que proyecta en dos dimensiones

un objeto que existe en tres dimensiones [32]. Por tanto, una proyección a 2-variedad es el proceso por el cual podemos bajar la dimensionalidad de un conjunto de datos a dos dimensiones y poder estudiarlo mejor [18]. De la misma manera que no existe una única forma de transformar una esfera a un plano tampoco existe una única forma de proyectar el conjunto de dimensiones a dos dimensiones, esto se explora en el capítulo 5 del *Cuaderno de investigación* adjuntado.

Detección de anomalías: también conocido como clasificación de clase única (*one-class clasification*), es un tipo de clasificador que se centra en acotar el espacio en el cual las instancias de una única clase existen de tal manera que detecte como anomalía cualquier instancia que no esté dentro de ese espacio. [35]

Ensemble: consiste en un método que se basa en la premisa que un conjunto de clasificadores débiles al combinarse genera un clasificador fuerte. Bajo esta definición se pueden crear diversas técnicas que permitan realizar crear estos clasificadores como son el crear subconjuntos aleatorios de las instancias para entrenar el mismo clasificador con diversos datos (*bagging*), entrenar a los clasificadores con pesos diferentes para las instancias o remuestreando los datos según este criterio (*boosting*), usar diferentes semillas para los clasificadores (comité aleatorio) o la creación de conjuntos de árboles según diferentes criterios (bosques). [22]

Balanceo de datos: este es el proceso por el cual de un conjunto de datos equilibrarlo de tal forma de limitar la diferente cantidad de instancias de las diversas clases, gracias a esto se evitan precisiones muy altas que simplemente desechen los datos de las clases menos balanceadas. Estos métodos pueden ser de sobremuestreo creando nuevas instancias de la clase minoritaria a partir de los datos existentes, submuestreo eliminando instancias innecesarias de la clase mayoritaria o el muestreo aleatorio creando un *ensemble* utilizando diversos conjuntos balanceados de manera aleatoria. [5, 6, 7]

Técnicas y herramientas

4.1. Investigación

El proyecto ha sido desarrollado aplicando diversas técnicas de minería de datos siguiendo el proceso KDD [23]. Aunque en esta memoria se explicará superficialmente, el desarrollo completo de la investigación se encuentra en el cuaderno de investigación adjuntado.

Para el proceso del clasificador final el flujo KDD ha sido el siguiente:

1. **Selección** Los datos han sido cedidos por la asociación abulense *PRONISA* que contiene jornadas nocturnas con identificadores de la cama, datos de presiones y datos vitales. Estos datos no están balanceados ya que existe una mayor cantidad de datos del paciente durmiendo que bajo una crisis epiléptica.

2. **Preprocesado** Los datos se han limpiado reduciendo ruidos mediante filtros de señal.

3. **Transformación** Se han generado datos estadísticos para series temporales que optimizaban el valor del área bajo la curva PCR [26] desde los datos preprocesados.

4. **Minería de datos** Se ha aplicado un sistema de clasificación doble, en primer lugar se utiliza un árbol de clasificación muy simple que divide entre las situaciones de despierto (bajas presiones en la cama) y acostado, en esta situación se aplica un clasificador *Random Forest* [3] para determinar si hay crisis o no.

5. **Interpretación** Se ha desplegado una aplicación web con datos en tiempo real para evaluar de manera constante la situación actual del paciente.

Este proceso de investigación se realizó sobre *Python* utilizando el ecosistema de librerías *SciPy* [12], en particular la librería de aprendizaje automático *scikit-learn* [18], de computación *NumPy* [17], de análisis de datos *Pandas* [15] y la de dibujado *MatPlotLib* [10]. Además, en algunas partes de la investigación se utilizó *Weka* [9] para estudiar métodos que no se encontraban en *scikit-learn* como el *Random Balance* [5] o *Rotation Forest* [21].

4.2. Servicio web

Backend

Las herramientas utilizadas para programar el servidor han sido las siguientes:

Flask [25] *Microframwork* de código abierto (BSD) que ofrece una capa de abstracción muy alta de un servicio web.

Jinja [24] Gestor de plantillas de código abierto (BSD) para Python.

Flask-SocketIO [8] Integración del servicio de *sockets*, *Socket.IO* [1], compatible con los *WebSockets*.

Gevent y Eventlet [2, 19] Librerías para el uso de tiempo real, asíncrono de hilos para el uso de *Socket.IO*.

Para la programación del sistema de hilos que distribuyen datos en tiempo real se siguieron los paradigmas de la programación orientada a objetos y de programación funcional. El sistema de rutas siguió las guías del *microframework Flask*.

Algunos patrones de diseño utilizados han sido el *Singleton* para la API así como un *Proxy* entre la interfaz web y la lógica de el API.

Frontend

Para el desarrollo de la parte visible de la aplicación se han usado otra serie de herramientas:

Bootstrap [30] *Framework* de *CSS* de código abierto (MIT) creado por *Twitter* para la creación de aplicaciones web redimensionables.

jQuery [31] *Framework* de *JavaScript* de código abierto (MIT) que simplifica el acceso al *HTML DOM* de la página.

Chart.js Librería de *JavaScript* de código abierto (MIT) para la creación de grafos en `canvas`.

4.3. Herramientas generales

Para el desarrollo general del proyecto se han utilizado las siguientes herramientas según el ámbito al que pertenecen:

Servidor

Nginx servidor web y de proxy reverso ligero de alto rendimiento [34] de código abierto (BSD simplificada).

MariaDB sistema de gestión de bases de datos derivado de *MySQL* [33] de código abierto (GPLv2). Este motor es extremadamente compatible con *MySQL* porque es creado como una bifuración de esto para garantizar la existencia de este motor bajo GPL.

Proxmox es un entorno de virtualización de servidores [37] de código abierto (AGPL). Su función principal es el despliegue y gestión de máquinas virtuales y contenedores.

Anarchy Arch sistema GNU/Linux derivado de *ArchLinux* [29] sobre el cual se ejecuta todo el servidor, está alojado en una máquina virtual del entorno *Proxmox*.

Miscelánea

- **Jupyter Notebooks**: IDE de programación de *Python* basado en *iPython* de código abierto (BSD).
- **PyCharm Professional**: IDE de programación para *Python* avanzado basado en *IntelliJ*.
- **Visual Studio Code**: Editor de código genérico de código abierto (MIT).
- **Postman**: IDE para la ejecución de request *HTTP*.
- **Selenium**: IDE de pruebas sobre web de código abierto (APACHE).
- **CertBot**: Sistema para la firma *SSL* sobre *HTTP* gratuito de *LetsEncrypt* de código abierto (MPL).
- **Overleaf**: Editor de LaTeX online para el trabajo colaborativo.
- **TeXStudio**: Editor de LaTeX de código abierto (GPLv2).
- **Dia**: Editor de diagramas genérico de código abierto (GPL).
- **StartUML**: Editor de diagramas UML.
- **Codesketch DB**: Traductor bidireccional de código-diagrama para bases de datos.
- **Filezilla**: Aplicación para la trasferencia de ficheros sobre *FTP* y *SFTP* de código abierto (GPLv2).
- **GitHub**: Servicio online de *hosting* para repositorios Git.
- **ZenHub**: Servicio online de integración de herramientas *SCRUM* sobre GitHub.

Aspectos relevantes del desarrollo del proyecto

5.1. Introducción

Trabajos relacionados

Conclusiones y Líneas de trabajo futuras

Bibliografía

[1] Damien Arrachequesne and Erik Little. Socket.io. https://socket.io/docs/, feb 2018.

[2] Karl J Aström. Event based control. In *Analysis and design of nonlinear control systems*, pages 127–147. Springer, 2008.

[3] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] Radio Amiga Burgos. Álvar Arnaiz González y José Francisco Díez Pastor – Premiados del Concurso Desafío Universidad Empresa, Marzo 2018.

[5] José F Díez-Pastor, Juan J Rodríguez, César García-Osorio, and Ludmila I Kuncheva. Random balance: ensembles of variable priors classifiers for imbalanced data. *Knowledge-Based Systems*, 85:96–111, 2015.

[6] José F Díez-Pastor, Juan J Rodríguez, César I García-Osorio, and Ludmila I Kuncheva. Diversity techniques improve the performance of the best imbalance learning ensembles. *Information Sciences*, 325:98–117, 2015.

[7] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484, 2012.

[8] Miguel Grinberg. Flask-socketio. https://flask-socketio.readthedocs.io/, jan 2019.

[9] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[10] John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.

[11] Jesper Jeppesen, Sándor Beniczky, A Fuglsang Frederiksen, Per Sidenius, and Peter Johansen. Modified automatic r-peak detection algorithm for patients with epilepsy using a portable electrocardiogram recorder. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4082–4085. IEEE, 2017.

[12] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. [Online; accessed 07/02/2019].

[13] Juha M Kortelainen, Mark Van Gils, and Juha Pärkkä. Multichannel bed pressure sensor for sleep monitoring. In *2012 Computing in Cardiology*, pages 313–316. IEEE, 2012.

[14] Yatindra Kumar, ML Dewal, and RS Anand. Epileptic seizures detection in eeg using dwt-based apen and artificial neural network. *Signal, Image and Video Processing*, 8(7):1323–1334, 2014.

[15] Wes McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010.

[16] National Institute of Neurological Disorders and Stroke NIH. Epilepsy information page. https://www.ninds.nih.gov/Disorders/All-Disorders/Epilepsy-Information-Page, may 2019.

[17] Travis E Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[19] Daniel Pope. Gevent: asynchronous i/o made easy. https://doi.org/10.5446/19958, 2014. [Online; accessed 16 May 2019].

[20] Sriram Ramgopal, Sigride Thome-Souza, Michele Jackson, Navah Ester Kadish, Iván Sánchez Fernández, Jacquelyn Klehm, William Bosl, Claus Reinsberger, Steven Schachter, and Tobias Loddenkemper. Seizure detection, seizure prediction, and closed-loop warning systems in epilepsy. *Epilepsy & behavior*, 37:291–307, 2014.

[21] Juan José Rodriguez, Ludmila I Kuncheva, and Carlos J Alonso. Rotation forest: A new classifier ensemble method. *IEEE transactions on pattern analysis and machine intelligence*, 28(10):1619–1630, 2006.

[22] Juan José Rodríguez. Clasificación bayesiana, basada en instancias y por combinación. Universidad de Burgos. [Online; accessed 15-Jun-2019].

[23] Juan José Rodríguez. Introducción a la minería de datos. Universidad de Burgos. [Online; accessed 15-Jun-2019].

[24] Armin Ronacher, David Lord, Adrian Mönnich, and Markus Unterwaditzer. Welcome to jinja2. http://jinja.pocoo.org/docs/2.10/, 2008.

[25] Armin Ronacher, David Lord, Adrian Mönnich, and Markus Unterwaditzer. Welcome to flask. http://flask.pocoo.org/docs/1.0/, 2010.

[26] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.

[27] Marie Tolkiehn, Louis Atallah, Benny Lo, and Guang-Zhong Yang. Direction sensitive fall detection using a triaxial accelerometer and a barometric pressure sensor. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 369–372. IEEE, 2011.

[28] Alexandros T Tzallas, Markos G Tsipouras, Dimitrios G Tsalikakis, Evaggelos C Karvounis, Loukas Astrakas, Spiros Konitsiotis, and Margaret Tzaphlidou. Automated epileptic seizure detection methods: a review study. In *Epilepsy-histological, electroencephalographic and psychological aspects*. IntechOpen, 2012.

[29] Wikipedia contributors. Arch linux — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Arch_Linux&oldid=896735398, 2019. [Online; accessed 16-May-2019].

[30] Wikipedia contributors. Bootstrap (front-end framework) — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Bootstrap_(front-end_framework)&oldid=895052706, 2019. [Online; accessed 16-May-2019].

[31] Wikipedia contributors. Jquery — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=JQuery&oldid=896325154, 2019. [Online; accessed 16-May-2019].

[32] Wikipedia contributors. Manifold — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Manifold&oldid=877548508, 2019. [Online; accessed 17-January-2019].

[33] Wikipedia contributors. Mariadb — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=MariaDB&oldid=897072367, 2019. [Online; accessed 16-May-2019].

[34] Wikipedia contributors. Nginx — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Nginx&oldid=896866646, 2019. [Online; accessed 16-May-2019].

[35] Wikipedia contributors. One-class classification — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=One-class_classification&oldid=897781715, 2019. [Online; accessed 15-June-2019].

[36] Wikipedia contributors. Principal component analysis — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Principal_component_analysis&oldid=878535871, 2019. [Online; accessed 17-January-2019].

[37] Wikipedia contributors. Proxmox virtual environment — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Proxmox_Virtual_Environment&oldid=896960348, 2019. [Online; accessed 16-May-2019].