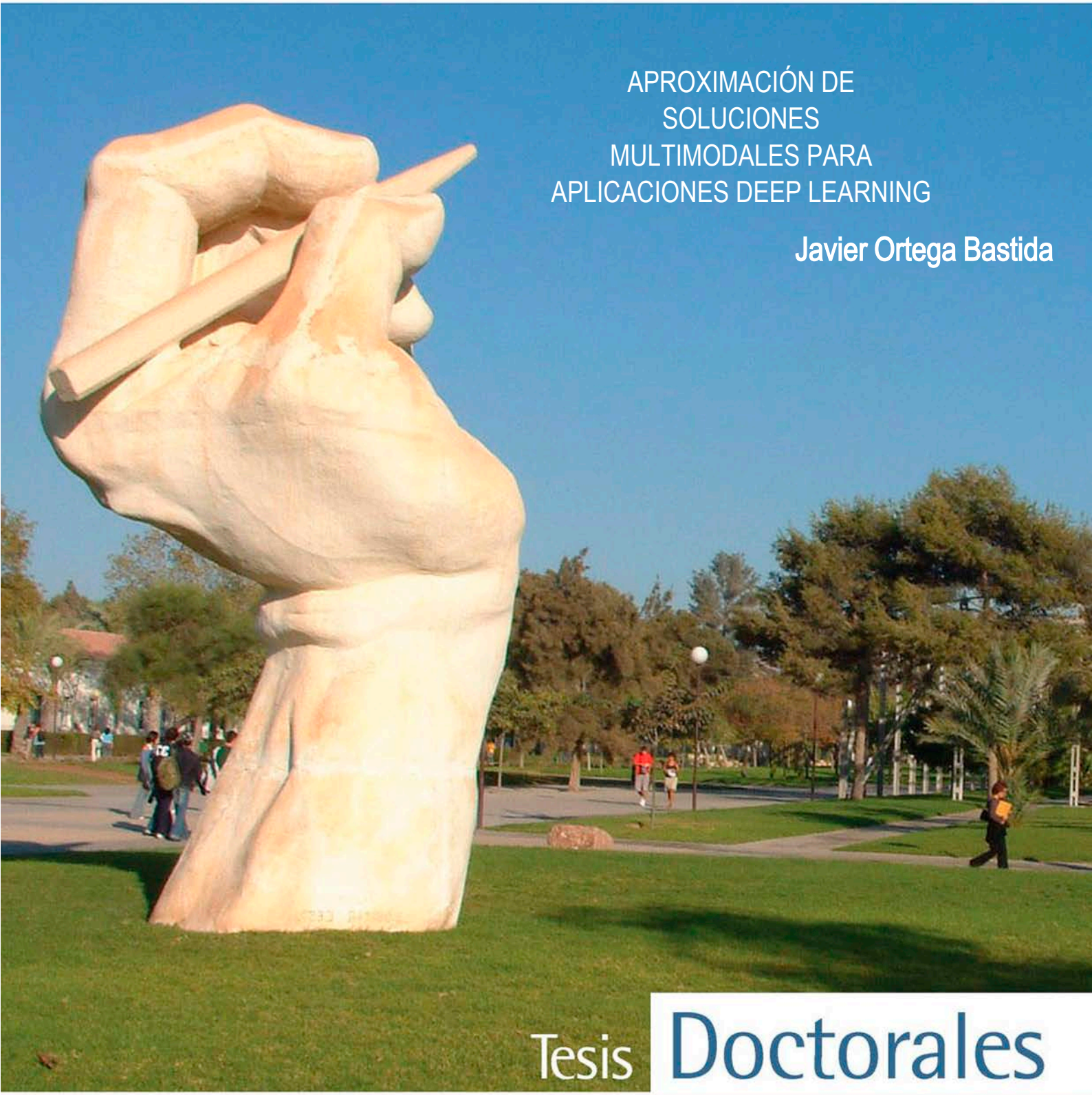
Universitat d'Alacant
Universidad de Alicante

APROXIMACIÓN DE SOLUCIONES MULTIMODALES PARA APLICACIONES DEEP LEARNING

Javier Ortega Bastida



Tesis Doctorales

UNIVERSIDAD de ALICANTE

Unitat de Digitalització UA
Unidad de Digitalización UA

Universitat d' Alacant
Universidad de Alicante

APROXIMACIÓN DE SOLUCIONES MULTIMODALES PARA APLICACIONES DEEP LEARNING

Universitat d'Alacant
Universidad de Alicante

Javier Ortega Bastida

Tesis doctoral

Alicante, julio 2021

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

DEPARTAMENTO DE LENGUAJES Y SISTEMAS INFORMÁTICOS

APROXIMACIÓN DE SOLUCIONES MULTIMODALES PARA APLICACIONES DEEP LEARNING

Javier Ortega Bastida

Universitat d'Alacant
Universidad de Alicante

DISERTACIÓN PRESENTADA PARA ASPIRAR AL GRADO DE DOCTOR POR LA UNIVERSIDAD DE ALICANTE (DOCTOR OF PHILOSOPHY) DOCTORADO EN INFORMÁTICA

Directores

Dr. Antonio Javier Gallego Sánchez

Dr. Juan Ramón Rico Juan

ALICANTE, 2021

Universitat d'Alacant
Universidad de Alicante

La combinación correcta de tecnología y humanos impulsará la prosperidad

Ulrich Spiesshofer.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Agradecimientos

Quiero dedicar un especial agradecimiento a mi director, Antonio Javier Gallego Sánchez, por su inestimable ayuda en el desarrollo del doctorado. Sin su apoyo, guía y exigencia de calidad, esta tesis no habría sido posible.

De igual forma, agradezco a mi codirector y tutor, Juan Ramón Rico Juan, sus sabios consejos. Su colaboración a lo largo del doctorado ha sido imprescindible para poder finalizarlo.

En tercer lugar, agradezco al profesor Pedro Albarrán su colaboración y asesoramiento en dos de los artículos que conforman esta disertación.

Por último, debo agradecer al Grupo de Reconocimiento de Formas e Inteligencia Artificial (GRFIA) del Departamento de Lenguajes y Sistemas Informáticos (DLSI) de la Universidad de Alicante por su apoyo económico a los trabajos realizados.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Resumen

Las oficinas de estadísticas de todo el mundo normalmente calculan y publican las estimaciones oficiales de indicadores macroeconómicos, como el Producto Interior Bruto (PIB), solo a nivel nacional y con una periodicidad anual. En algunos países se pueden encontrar también datos oficiales con frecuencias menores, como semestrales o trimestrales, y, en ocasiones, a niveles geográficos inferiores, aunque este último caso es menos común y cuando se publica suele realizarse únicamente con carácter anual. En países como España, el PIB nacional se comunica oficialmente con una regularidad trimestral, sin embargo, el PIB a nivel regional solamente se publica anualmente. Además de estos indicadores, no existen otras estimaciones oficiales, ya que, por ejemplo, no es posible conocer el PIB para zonas geográficas de niveles inferiores, como ciudades o provincias, o con periodicidades distintas.

Sin embargo, para la actividad política y financiera resulta de gran interés conocer la evolución económica en frecuencias menores a un año o trimestre, ya que esto resulta decisivo para la toma de decisiones. De igual forma, conocer esta información para unidades geográficas más específicas, permitiría a los actores económicos potenciar sus inversiones y evitar las zonas de mayor riesgo.

Esta disertación propone un enfoque multimodal basado en aprendizaje automático para la predicción del PIB a niveles geográficos distintos al nacional y con frecuencias temporales inferiores a la anual. La aproximación propuesta combina diferentes fuentes de información, como valores históricos y mensajes escritos en la red social Twitter. Para esto, se realiza un estudio exhaustivo donde distintas soluciones y metodologías son evaluadas para comparar su precisión frente a otras propuestas existentes, incluyendo técnicas de aprendizaje supervisado y no supervisado, de procesamiento del lenguaje natural y de análisis de secuencias.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Índice general

Índice de figuras xv

Índice de tablas xix

1. Introducción 1
- 1.1 Motivación . . . 1
- 1.2 Propuesta y objetivos . . . 4
- 1.3 Tecnologías . . . 6
- 1.4 Contribuciones . . . 9
  - 1.4.1 Publicaciones . . . 9
- 1.5 Estructura de la tesis . . . 11

2. Estado del arte 13
- 2.1 Aprendizaje Automático . . . 13
  - 2.1.1 Redes Neuronales . . . 16
- 2.2 Deep Learning . . . 21
  - 2.2.1 Redes Neuronales Convolucionales . . . 24
  - 2.2.2 Autoencoders . . . 29
  - 2.2.3 Redes Neuronales Recurrentes . . . 30
  - 2.2.4 Word embedding . . . 33
- 2.3 Aprendizaje Automático en la economía . . . 35
  - 2.3.1 Redes sociales como reflejo económico . . . 36
  - 2.3.2 Predicción del PIB . . . 38
- 2.4 Multimodalidad . . . 41
  - 2.4.1 Clasificación de la multimodalidad . . . 43
  - 2.4.2 Aplicaciones de la multimodalidad . . . 48

Universitat d'Alacant
Universidad de Alicante

3. Aplicaciones de la multimodalidad al reconocimiento de imágenes 53
3.1 Introducción 54
3.2 Conjunto de datos 56
3.2.1 Metadatos 57
3.3 Metodología 60
3.3.1 Preprocesado de las imágenes 61
3.3.2 Preprocesado de los metadatos 62
3.3.3 Paso de clasificación 66
3.4 Resultados obtenidos 67
3.4.1 Evaluación usando metadatos 67
3.4.2 Evaluación usando características visuales 70
3.4.3 Evaluación combinando metadatos y características visuales 72
3.5 Conclusiones y trabajo futuro 73

4. Estimación del PIB regional mediante información textual 75
4.1 Introducción 76
4.2 Metodología 77
4.2.1 Preprocesado de los tweets 78
4.2.2 Generación de word embedding 79
4.2.3 Autoencoder híbrido 81
4.2.4 Paso de predicción 83
4.3 Conjunto de datos 85
4.4 Experimentos 87
4.4.1 Resultados del autoencoder 87
4.4.2 Predicción del PIB 89
4.4.3 Comparación 91
4.5 Conclusiones y trabajo futuro 93

5. Aproximación multimodal para la predicción del PIB regional 95
5.1 Introducción 96
5.2 Metodología 99
5.2.1 Paso 1. Preprocesar tweets 100
5.2.2 Paso 2. Embedding 100
5.2.3 Paso 3. Codificación de los tweets 101
5.2.4 Paso 4. Filtrado de valores atípicos 102
5.2.5 Paso 5. Regresor 102

Universitat d'Alacant
Universidad de Alicante

5.2.6 Paso 6. Red recurrente multimodal 103
5.2.7 Etapas de entrenamiento e inferencia 105
5.3 Configuración para los experimentos 107
5.3.1 Conjunto de datos 107
5.3.2 Topologías de la red multimodal 110
5.4 Experimentos 113
5.4.1 Evaluación de los hiperparámetros 113
5.4.2 Evaluación del proceso de filtrado 114
5.4.3 Estudio de ablación 116
5.4.4 Evaluación final del modelo 119
5.4.5 Evaluación de la influencia de la entrada 122
5.4.6 Análisis de los tweets 124
5.4.7 Impacto económico del coronavirus 126
5.5 Conclusiones y trabajo futuro 128

6. Conclusiones **131**
6.1 Aplicación de la multimodalidad al reconocimiento de imágenes . 131
6.2 Estimación del PIB mediante información textual 133
6.3 Solución multimodal para la predicción del PIB 134
6.4 Conclusiones generales 135
6.5 Trabajo futuro 136

7. Bibliografía **137**

A. Lista de acrónimos **155**

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Índice de figuras

1.1 Captura del entorno de desarrollo online Google Colaboratory (Colaboratory, 2021). 8

2.1 Programación de IA tradicional frente al Aprendizaje Automático. 14
2.2 Aplicaciones del Machine Learning en la industria. 16
2.3 Representación de una neurona. Los impulsos van de izquierda a derecha. Imagen extraída de Hardy (2001). 17
2.4 Los dos principales tipos de sinapsis en el cerebro: excitadora (A) e inhibidora (B). Imagen extraída de Hardy (2001). 18
2.5 Ejemplo de una Red Neuronal, donde los círculos representan las neuronas artificiales que componen la red y las líneas las conexiones entre ellas. Imagen elaborada a partir de Bre *et al.* (2017). 19
2.6 Ejemplo de una neurona artificial o perceptrón. Recibe x entradas, pondera estas entradas por unos pesos w, suma el resultado y añade un umbral o bias b. Este resultado es modificado finalmente por una función de activación f. 20
2.7 Comparación entre las primeras redes neuronales con las redes neuronales profundas. Imagen adaptada de Waldrop (2019). . . 22
2.8 Esquema organizativo de la IA. 23
2.9 Esquema general de la arquitectura de red CNN. Imagen adaptada de Wikipedia Commons (2015). 25
2.10 Ejemplo de la arquitectura CNN denominada AlexNet propuesta por Krizhevsky *et al.* (2012). 27
2.11 Ejemplo de un bloque residual en la arquitectura ResNet. Imagen extraída de He *et al.* (2015). 28
2.12 Arquitectura ResNet con 34 capas propuesta por He *et al.* (2015). 29

Universitat d'Alacant
Universidad de Alicante

2.13 Esquema típico de un autoencoder. 30
2.14 Ejemplo de una neurona recurrente (izquierda) y su versión desplegada (derecha). Este tipo de neurona predice su salida para el instante h_t a partir de la entrada recibida en ese instante (x_t) y también a partir de la salida del estado anterior ($h_t - 1$). Imagen adaptada de Su y Li (2019). 31
2.15 Esquema detallado de una unidad tipo LSTM. Imagen extraída de Greff *et al.* (2017). 32
2.16 Muestra de correlación semántica entre palabras utilizando métodos de representación word embedding. 34
2.17 Mapa con los flujos de movilidad entre municipios de España basados en la actividad de Twitter. Imagen extraída de Llorente *et al.* (2015). 37
2.18 Desafío de la representación multimodal. (A) Representación conjunta. (B) Representación coordinada. 44
2.19 Aprendizaje a partir de la proyección lineal correlacionada entre las representaciones de varias modalidades propuesto por Andrew *et al.* (2013). 44
2.20 Ejemplo de cómo el modelo multimodal propuesto por Karpathy *et al.* (2014) aprende a asociar fragmentos de imágenes con oraciones para encontrar las relaciones. 45
2.21 Arquitecturas típicas de los métodos de fusión en modelos multimodales. 46
2.22 (A) Modelo de traducción basado en ejemplos. (B) Modelo de traducción generativo. Imagen elaborada a partir de Baltrušaitis *et al.* (2019). 47
2.23 Tipos de paralelismo de datos utilizados en el coaprendizaje: (A) En paralelo, (B) No paralelo, (C) Híbrido. Imagen elaborada a partir de Baltrušaitis *et al.* (2019). 48
2.24 Esquema de un autoencoder multimodal (*Bimodal Deep Autoencoder*) para la reconstrucción de audio y vídeo propuesto por Ngiam *et al.* (2011). 50
2.25 Arquitectura de la red multi-entrada denominada “MENET” propuesta por Do *et al.* (2017). 51

3.1 Arquitectura con la secuencia de procesamiento de la app Mirbot. 55

Universitat d'Alacant
Universidad de Alicante

3.2 Captura del panel de administración utilizado para la validación de las imágenes clasificadas por la aplicación Mirbot. En la parte izquierda se muestran ejemplos de la clase "plátano" y en la derecha de la clase "ratón", donde uno de los ejemplos se ha marcado como incorrecto ya que la imagen está desenfocada y probablemente no pertenece a dicha clase. 57
3.3 Arquitectura general del método propuesto. 60
3.4 Proyección de los word embeddings utilizando t-SNE (van der Maaten, 2008) para los códigos de país y los tipos de características. 71

4.1 Arquitectura general del método propuesto. 78
4.2 Resultados (MSE) obtenidos al variar el parámetro λ de la función de pérdida (ver Ecuación 4.2) para ponderar la contribución de las salidas de la red. $\lambda = 1$ significa que solo se usa la salida de los tweets, mientras que $\lambda = 0$ significa que solo se considera la salida del PIB. 88
4.3 Comparación entre el valor real del PIB (línea azul) y la predicción del PIB (línea roja) realizada por el método propuesto para todos los trimestres del periodo considerado. Además se ha marcado mediante líneas punteadas en color verde las diferencias entre los valores estacionales (solo para los ciclos ascendentes). 92

5.1 Esquema general del método propuesto. El orden de los pasos se marca en azul. 100
5.2 Análisis de la distribución de longitud de los tweets para cada uno de los conjuntos de datos considerados. 112
5.3 Resultados promedio (en términos de MSE más desviación estándar) obtenidos después de variar el parámetro λ de la función de pérdida en las cuatro CCAA consideradas. 114
5.4 Distribución de características aprendida por el espacio latente del autoencoder para dos trimestres aleatorios usando la representación obtenida con t-SNE (van der Maaten, 2008). 115
5.5 Comparación de la evolución del valor real del PIB (medido en tasa de crecimiento interanual) con la predicción realizada por el método propuesto para las cuatro comunidades autónomas consideradas. 122

Universitat d'Alacant
Universidad de Alicante

5.6 Resultado del método propuesto en presencia de diferentes niveles de ruido en los atributos (eje horizontal), introducido en cada una de las entradas por separado. 123
5.7 Tasa de error al predecir el crecimiento interanual del PIB mediante el método propuesto en función de la semana del trimestre en el que se realiza la previsión. El baseline estacional se incluye a modo de referencia. 124

Universitat d'Alacant
Universidad de Alicante

Índice de tablas

2.1 Tabla resumen de los desafíos asociados a cada área de aplicación de la multimodalidad. Tabla adaptada del trabajo de Baltrušaitis *et al.* (2019). 52

3.1 Metadatos obtenidos del *smartphone*. 58

3.2 Metadatos obtenidos de la codificación geográfica inversa (`Gisgraphy`) en el lado del servidor. 59

3.3 Metadatos filtrados utilizando diferentes métodos de selección de atributos combinados mediante un esquema de votación. El número de votos de cada metadato se indica entre paréntesis (de un total de 8 posibles votos). Todos los campos corresponden a valores numéricos a excepción de los marcados con un asterisco (*), los cuales se preprocesaron usando Doc2Vec. 68

3.4 Comparación de los mejores resultados (utilizando la métrica *top-1*) para cada clasificador utilizando los metadatos sin preprocesamiento y con preprocesamiento. 69

3.5 Resultados obtenidos (en términos de *top-1*) a partir de los NC por cada clasificador para los distintos niveles de etiquetado considerados. 72

3.6 Resultados obtenidos mediante los distintos clasificadores y niveles de etiquetado considerados al combinar los NC y los medatados del dataset. 73

Universitat d'Alacant
Universidad de Alicante

4.1 Topologías consideradas para el autoencoder híbrido. La última capa tiene dos salidas: $FC(n)$ (*Fully Connected* o capa de perceptrones totalmente conectada) para la reconstrucción del *embedding* n-dimensional de los tweets, y $FC(1)$ para la salida del PIB, la cual se conecta directamente al espacio latente de la red (esta conexión se representa con el símbolo $\rightarrow$). 83

4.2 Algunos ejemplos de las cuentas de Twitter utilizadas en el conjunto de datos. 86

4.3 Comparación del error de reconstrucción obtenido (en términos de MSE) con las diferentes topologías de autoencoder (AE) consideradas (ver Tabla 4.1). En negrita se muestran los mejores resultados obtenidos por topología, por método de *embedding* y en promedio. 89

4.4 Comparación del resultado obtenido (MAE) para la predicción del PIB utilizando el método propuesto y otros cuatro enfoques. 91

4.5 Comparación del método propuesto con otros tres métodos del estado del arte en términos de RMSE. 93

5.1 Cuentas de Twitter seleccionadas para la Comunidad Valenciana. 108

5.2 Cuentas de Twitter seleccionadas para Madrid. 109

5.3 Cuentas de Twitter seleccionadas para Andalucía. 109

5.4 Cuentas de Twitter seleccionadas para Murcia. 110

5.5 Listado de palabras clave utilizadas para filtrar los tweets. 111

5.6 Resumen de la composición de los conjuntos de datos utilizados en la experimentación. 112

5.7 Topologías consideradas para la red multimodal. La notación seguida es la misma que la empleada en la Tabla 4.1. La operación "$\oplus$ Concat" concatena el resultado de la capa #3 con el resultado obtenido en la primera etapa del método propuesto después de procesar los tweets (ver la Figura 5.1). 113

Universitat d'Alacant
Universidad de Alicante

5.8 Evaluación del resultado obtenido (en términos de MAE) por la primera parte del método propuesto (autoencoder + filtrado + regresor, ver Figura 5.1). Nuestra propuesta se compara con otros seis enfoques: tres baselines calculados a partir de datos históricos del PIB y otras tres propuestas calculadas a partir de tweets. Los mejores resultados se marcan en negrita y se muestran por separado para cada tipo de dato de entrada. 117
5.9 Evaluación del resultado obtenido (en términos de MAE) con el método propuesto al agregar la última parte de la red: la red multimodal que combina el resultado de los tweets con el PIB histórico (ver Figura 5.1). Para esta segunda red, también se evalúan diferentes tipos de capas y número de capas. Los mejores resultados están marcados en negrita. 118
5.10 Tasa de error al predecir el crecimiento interanual del PIB mediante el método propuesto en comparación con el baseline estacional y otros enfoques del estado del arte. 120
5.11 Resultados del test no paramétrico de Wilcoxon a partir de muestras pareadas (Demsar, 2006). El símbolo ✓establece que es test se supera significativamente (es decir, el esquema de la fila es mejor que el esquema de la columna). La significancia se ha establecido en $p < 0{,}05$. 121
5.12 Algunos ejemplos de los tweets más y menos influyentes detectados por el método propuesto para calcular el valor del PIB para el período 2012-2016. También se incluye la cuenta que escribió cada tweet. 127
5.13 Tasa de crecimiento del PIB real para los primeros trimestres de 2020 en comparación con la predicción realizada por el baseline estacional y el método propuesto. 128
5.14 Algunos ejemplos de los tweets más y menos influyentes detectados por el método propuesto para calcular el valor del PIB para el primer y segundo trimestre de 2020. 129

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 1

Introducción

Esta tesis es el resultado de los estudios desarrollados en el programa de Doctorado en Informática de la Universidad de Alicante. Los trabajos han sido realizados en el de Grupo de Reconocimiento de Formas e Inteligencia Artificial (GRFIA) del Departamento de Lenguajes y Sistemas Informáticos (DLSI) de esta misma universidad.

El capítulo se organiza en seis aparatados: en primer lugar se expone la motivación de esta tesis, a continuación, en la sección 1.2, se presenta la propuesta del trabajo y los objetivos que queremos alcanzar con ella. En la sección 1.3 hablaremos de las tecnologías empleadas y, seguidamente, en las secciones 1.4 y 1.4.1, se establecen las contribuciones de la tesis y los artículos publicados que sustentan el trabajo. Por último, la sección 1.5 detalla la estructura que se seguirá en el resto de capítulos.

Universitat d'Alacant
Universidad de Alicante

1.1. Motivación

La motivación de este trabajo se sustenta en la hipótesis de que el uso de información adicional y complementaria permitirá obtener mejores resultados en procesos que empleen algoritmos de Aprendizaje Automático (del inglés *Machine Learning* o ML) o, más recientemente, en métodos de Aprendizaje Profundo (*Deep Learning* o DL). A esta técnica de combinación de información se la conoce por el término *multimodalidad* y ha sido aplicada con éxito en multitud de trabajos Vidal y Casacuberta (2011). Por tanto, se espera que la

combinación de multimodalidad con técnicas de ML y DL pueda beneficiarse de la fusión de información y aprender características relevantes procedentes de las distintas fuentes de datos. Esto permitirá que los métodos encuentren información que no es visible en cada modalidad individual, sino que surja de la combinación multimodal y, en consecuencia, tomen decisiones más informadas.

Con este fin, se han evaluado y propuesto diferentes aproximaciones multimodales combinadas con técnicas de ML y DL. En concreto, nos centraremos en el estudio de un sistema para el reconocimiento de objetos en una primera toma de contacto de estas técnicas, el cual combinará las características extraídas de las imágenes con una serie de metadatos, y un método para la predicción del Producto Interno Bruto (PIB) regional utilizando diferentes fuentes de información, como datos numéricos e información textual.

La primera aproximación de esta disertación se centra en realizar un estudio de las técnicas del estado del arte. Tradicionalmente, las tareas de identificación de objetos aplicadas en la visión por computador trabajan con imágenes como única fuente de información para realizar la clasificación. Un ejemplo de estos sistemas es la aplicación Mirbot (Pertusa y Gallego, 2018) implementada en Pertusa *et al.* (2018). En esta primera aproximación se propone una extensión de este trabajo para combinar los metadatos recopilados por el dispositivo móvil que tomó la imagen con la información visual de la fotografía capturada.

La segunda aproximación multimodal consistirá, como ya se ha adelantado, en un método para la predicción del PIB a nivel regional y con una periodicidad trimestral. Actualmente, el análisis y solución a problemas económicos se sigue abordando desde la perspectiva de las aproximaciones de resolución tradicionales, como son los métodos estadísticos. Sin embargo, dada la evolución tecnológica, este tipo de tareas podrían ser mejoradas mediante propuestas más avanzadas, como por ejemplo mediante el uso de técnicas de ML y DL y la combinación multimodal de información. En el estado del arte encontramos soluciones que emplean Inteligencia Artificial (IA) para efectuar predicciones más precisas que las aproximaciones estadísticas que se usaban habitualmente en el ámbito financiero. Un ejemplo de ello, son los algoritmos de robots asesores (o *robo-advisors*) (Waliszewski y Zieba-Szklarska, 2020) para la gestión y el

asesoramiento automatizado de valores financieros, como pueden ser las carteras de inversión.

Dentro de este campo, el Mercado de Valores ha sido el objetivo principal de los avances en ML y DL. Sin embargo, el mundo económico cuenta con múltiples vertientes en donde el uso de este tipo de técnicas no ha sido estudiado en profundidad, por lo que se considera de gran interés su aplicación. Una de las vertientes con más interés son las estadísticas oficiales de agregados macroeconómicos relevantes, como el PIB o la tasa de desempleo, las cuales se centran principalmente en la evolución anual de estos indicadores para todo el país. Sin embargo, la formulación de políticas y toma de decisiones a menudo requieren información con una mayor frecuencia (trimestral o mensual) y también para unidades geográficas más específicas (regiones o ciudades).

Por este motivo, las Oficinas Nacionales de Estadística calculan algunos de estos indicadores económicos mediante técnicas de interpolación u otro tipo de métodos que no siempre son públicos o claros, y que además no cubren todos los casos posibles. En España, el PIB nacional está disponible al público sobre una base anual y trimestral, pero el PIB a nivel regional, para las diferentes administraciones territoriales, se calcula solo anualmente y no se proporciona una cifra oficial para provincias o ciudades.

Como se verá en el siguiente capítulo, en la literatura encontramos solo unos pocos trabajos donde se empleen técnicas de ML para la predicción del PIB (y normalmente a nivel nacional), cuyos resultados dejan espacio al desarrollo de aproximaciones más precisas y generalizables. Esto hace que sea de gran interés la búsqueda de nuevos enfoques que mejoren tanto los resultados del estado del arte como que permitan su aplicación para la predicción a distintos niveles geográficos y con periodicidades distintas.

Por estos motivos, la segunda aproximación multimodal propuesta consistirá en un método para la predicción del PIB a nivel regional, utilizando para esto la información económica extraída y filtrada de la red social Twitter y una serie de valores históricos referentes a la evolución de la economía. El sistema propuesto para esta tarea permitirá a los actores del mundo económico (ya sean políticos, accionistas, inversores, empresarios, etc.), conocer la evolución

Universitat d'Alacant
Universidad de Alicante

económica de una región en diferentes intervalos de tiempo para, de esta forma, disponer de más información a la hora de tomar decisiones. Debido al perfil con el que instituciones, periódicos o economistas utilizan Twitter, y al límite de caracteres establecido que fuerza a sintetizar en los mensajes las ideas más importantes, se eligió esta red social frente a otras alternativas. Además, el uso de esta fuente de información aportará una ventaja añadida, como es el estudio de las opiniones más y menos influyentes a la hora de realizar una predicción.

Para la evaluación de esta metodología se considerarán distintas Comunidades Autónomas (CCAA), incluyendo casos uniprovinciales y pluriprovinciales, se examinarán los resultados para frecuencias temporales trimestrales y semanales y se comparará con trabajos del estado del arte. Por último, y dadas las circunstancias en las que este estudio se realizó, el método propuesto también se evaluará en un entorno real condicionado por la pandemia de COVID-19. Muchos trabajos han puesto en práctica aplicaciones de IA en un entorno marcado por el COVID-19 (Bullock *et al.*, 2020), por lo que en este trabajo también se pretende aplicar la propuesta presentada para conocer si es capaz de capturar los cambios ocasionados en la actividad económica por los efectos de la pandemia. Esto permitirá analizar la capacidad del método en una situación extrema, en la que el desplome del PIB ha sido el mayor en casi un siglo, y que, como veremos, no se podría predecir sin la combinación de diferentes fuentes de información.

Universitat d'Alacant
Universidad de Alicante

1.2. Propuesta y objetivos

El objetivo principal de este trabajo es el estudio de los beneficios que la multimodalidad puede aportar a las técnicas de ML y DL. Para ello, se considerarán dos casos de estudio, un trabajo inicial sobre la combinación de imagen y metadatos, y un segundo trabajo, como tema central de la tesis, para el estudio de la aplicación de la multimodalidad en la economía, en concreto, para la predicción del PIB a partir de mensajes escritos en Twitter y secuencias de valores históricos del PIB. Este segundo trabajo pretende además, dado su interés en el campo de la economía, calcular previsiones con mayor frecuencia que las estadísticas oficiales (publicadas solo anualmente a nivel regional en

España) y que las predicciones trimestrales no oficiales existentes (que se basan en indicadores económicos que solo están disponibles con un retraso de varios meses).

Por lo tanto, los principales objetivos a alcanzar con las propuestas presentadas son:

1. Estudiar y evaluar técnicas de multimodalidad en tareas de visión por computador en un entorno como la aplicación Mirbot para comprender las fortalezas y debilidades a tener en cuenta al trabajar con este tipo de metodologías.

2. Diseñar un sistema multimodal para la clasificación de imágenes. Como caso de uso, usaremos la base de datos del sistema Mirbot Pertusa *et al.* (2018), la cual almacena tanto imágenes como metadatos del dispositivo móvil que tomó la captura.

3. Estudiar el preprocesado de datos secuenciales (como las cadenas de texto) para su uso en redes neuronales: desde el filtrado, pasando por la codificación usando distintas aproximaciones para su transformación a vectores numéricos (*embeddings*), hasta el uso de las características intermedias de las redes neuronales que sinteticen la información.

4. Construir un autoencoder que capture en su espacio latente las características más representativas de dos modalidades de información, como será el caso de la información económica contenida en textos y su relación con el valor del PIB.

5. Implementar un sistema predictivo para la estimación del PIB que combine varias modalidades de información que le permita tanto seguir la tendencia o evolución del valor del PIB como detectar posibles fluctuaciones o cambios bruscos en esta tendencia.

6. Elaborar una base de datos representativa de la realidad económica a partir de las cuentas de Twitter más influyentes en el mundo económico de varias CCAA de España donde se vaya a predecir el PIB.

Universitat d'Alacant
Universidad de Alicante

7. Evaluar la metodología propuesta en un entorno real para predecir el impacto económico regional ocasionado por la pandemia del COVID-19.

1.3. Tecnologías

Establecidos los objetivos, pasamos a presentar las tecnologías que emplearemos para alcanzarlos.

Para la programación usaremos el lenguaje Python (Python, 2020), el cual se ha consolidado como uno de los lenguajes preferidos para la informática científica (Dubois, 2007; Millman y Aivazis, 2011) y la implementación de algoritmos de IA. Para el desarrollo de técnicas de ML y DL, Python nos ofrece una gran cantidad de librerías que facilita el trabajo gracias a su lenguaje conciso y sencillo, lo que permite programar de forma más productiva.

El trabajo lo realizaremos sobre la plataforma que nos ofrece la tecnología TensorFlow desarrollada por Google Brain (Abadi *et al.*, 2015, 2016). TensforFlow se define como "un ecosistema integral y flexible de herramientas, bibliotecas y recursos de la comunidad que permite a los investigadores impulsar un aprendizaje automático innovador y, a los desarrolladores, compilar e implementar con facilidad aplicaciones con tecnologías de ML" (TensorFlow, 2021).

Para el desarrollo de las metodologías basadas en DL, además de TensforFlow, se utilizará la librería Keras (Keras, 2021). Esta librería implementa una API de alto nivel escrita en Python para el desarrollo de redes neuronales que abstrae al usuario del funcionamiento interno de su *framework*. Gracias a esta abstracción podemos construir modelos complejos escribiendo pocas líneas de código, ya que su modulación en funciones agrupa toda la lógica de la librería, y solo es necesario bajar a bajo nivel para la implementación de funcionalidades propias. Si bien Keras se ejecutaba sobre TensorFlow, en 2020 Keras terminó por formar parte de su API, pasando a llamarse *tf.Keras* (TensorFlow Core, 2021).

De forma complementaria a TensorFlow y Keras, también se ha utilizado la librería *Scikit-learn* (Pedregosa *et al.*, 2011). Esta librería ofrece un conjunto

de algoritmos de ML que incluye métodos para la clasificación, el etiquetado de datos, la regresión o el cálculo de métricas. Si bien la librería está escrita principalmente en Python, incorpora las bibliotecas C++ LibSVM (Chang y Lin, 2011) y LibLinear (Fan *et al.*, 2008), que proporcionan las implementaciones de determinados modelos, como *Suport Vector Machine* (SVM).

Durante el proceso de experimentación, además del servidor de procesamiento proporcionado por el grupo GRFIA (con las especificaciones: CPU Intel(R) Core(TM) i7-6700@ 3.40GHz con 16 GB de RAM DDR4 y una GPU Nvidia GeForce GTX 1070), se ha utilizado el entorno de desarrollo online de Google, llamado Google Colaboratory (o Google Colab) (Colaboratory, 2021). Esta herramienta (ver Figura 1.1) está integrada entre las muchas opciones que nos ofrece la plataforma de Google Drive para trabajar desde el navegador, lo que además nos permite mantener en la nube nuestro código y poder compartirlo con otros desarrolladores. Se trata de una web dinámica en la que podemos trabajar con bloques de código de forma separada (llamados *Jupyter Notebooks*). Estos cuadernos interpretan el lenguaje Python y, por su puesto, las librerías de TensorFlow, Keras y Scikit-learn. Sin embargo, la gran ventaja de Google Colab reside en el acceso a GPUs y TPUs de forma gratuita. De esta forma, al ejecutar un código, se conecta a un servidor de Google que nos brinda el acceso a estas unidades de procesamiento. Esto nos ha permitido acelerar los tiempos de entrenamiento y ejecución gracias a su paralelización.

Para la codificación de cadenas de texto se consideraron distintas metodologías propuestas en el estado del arte (como revisaremos en los siguientes capítulos). Para la evaluación de los métodos pre-entrenados utilizamos varias librerías de *text* y *word embedding*, como son: *Spacy* (spaCy, 2020), *LASER (Language-Agnostic SEntence Representations)* (LASER, 2020), *TF-HUB* (TensorFlowHub, 2020) y *UKPLab (Ubiquitous Knowledge Processing Lab)* (UKPLab, 2020). En concreto, una de las que mejor resultado obtuvo fue *UKPLab*, ya que nos ofrece modelos para más de 50 idiomas pre-entrenados con enormes conjuntos de datos obtenidos de emails, Reddit, noticias, artículos de Wikipedia e incluso mensajes de Twitter, lo que hace a esta librería la idónea para trabajar con tweets.

Para recopilar las bases de datos de tweets inicialmente se consideró la API facilitada por Twitter. Sin embargo, esta interfaz está limitada a un número de

Figura 1.1.: *Captura del entorno de desarrollo online Google Colaboratory (Colaboratory, 2021).*

Universitat d'Alacant
Universidad de Alicante

descargas. Para solucionar este inconveniente utilizamos la librería de código abierto *GetOldTweets* (GetOldTweets3, 2020), con la que podemos descargar tweets sin límite, indicando como entrada el nombre de la cuenta de la que se quieren descargar los mensajes, el rango temporal en el que mirar e incluso una lista de palabras clave para filtrar solo los tweets que nos interesen.

Para la visualización de los datos codificados (ya sean los *embeddings* resultantes de la codificación de textos o la representación interna aprendida por las distintas arquitecturas de red que consideraremos), usaremos el método *t-Distributed Stochastic Neighbor Embedding* (t-SNE) (van der Maaten, 2008) y la librería MatPlotLib (Barrett *et al.*, 2005). Además, para la generación de gráficas nos basaremos en Gnuplot (Williams *et al.*, 2020).

1.4. Contribuciones

Las principales contribuciones de esta tesis pueden sintetizarse en los siguientes puntos:

1. Un sistema multimodal para la clasificación de imágenes combinando la información visual aprendida por una red neuronal con metadatos extraídos de dispositivos móviles.

2. Un método multimodal para la predicción del PIB regional con capacidad para detectar eventos que tengan un impacto significativo en la economía (como fue la pandemia del COVID-19).

3. La evaluación de modelos pre-entrenados para la codificación de tweets y su síntesis a través de una arquitectura de red tipo autoencoder para la extracción de las características más representativas que combinan varias fuentes de información.

4. Un sistema que permite determinar las opiniones más y menos influyentes sobre una tendencia económica.

5. El empleo de diferentes técnicas de ML en el filtrado de la información codificada para mejorar la precisión de los algoritmos de DL empleados y su combinación con aproximaciones multimodales.

Universitat d'Alacant
Universidad de Alicante

1.4.1. Publicaciones

Esta tesis es el resultado de las investigaciones realizadas por el candidato doctoral en el programa de Doctorado en Informática, las cuales se sustentan en las siguientes publicaciones:

Artículo de revista:

- **Autores:** Ortega-Bastida, J., Gallego, A.J., Rico-Juan, J.R., & Albarrán, P.
  Título: A Multimodal Approach for Regional GDP Prediction using Social Media Activity and Historical Information.
  Revista: Applied Soft Computing.
  ISSN: 1568-4946
  Año de publicación: 2021
  Indicios de Calidad: Indexada en el Journal Citation Reports (JCR) en el cuartil Q1. Factor de impacto 6,725.

Artículos de congreso:

- **Autores:** Ortega-Bastida, J., Gallego, A.J., Rico-Juan, J.R., & Albarrán, P.
  Título: Regional gross domestic product prediction using Twitter deep learning representations.
  Congreso: 17th IADIS International Conference on Applied Computing (IADIS AC).
  ISBN: 978-989-8704-24-5
  Páginas: 89–96 **Lugar de celebración:** Lisboa, Portugal
  Año de publicación: 2020
  Indicios de Calidad: Calificado como CORE C en el CORE Alpha Rank.

- **Autores:** Ortega-Bastida, J., Gallego, A.J., & Pertusa, A. (2019).
  Título: Multimodal object recognition using deep learning representations extracted from images and smartphone sensors.
  Congreso: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. Iberoamerican Congress on Pattern Recognition (CIARP)
  ISBN: 978-3-030-13469-3
  Páginas: 521–529
  Lugar de celebración: Madrid, España
  Año de publicación: 2018
  Indicios de Calidad: Calificado como CORE C en el CORE Alpha Rank.

Universitat d'Alacant
Universidad de Alicante

1.5. Estructura de la tesis

El resto de contenidos de esta tesis se estructuran en los siguientes capítulos:

1. El capítulo 2 revisa el estado del arte con los trabajos previos que han servido de inspiración y punto de partida para las soluciones propuestas.

2. El capítulo 3 realiza un estudio inicial sobre aproximaciones multimodales para la clasificación de objetos a partir de imágenes y metadatos.

3. El capítulo 4 detalla la primera etapa del método propuesto para la predicción del PIB para la Comunidad Valenciana empleando únicamente la información almacenada en los mensajes de Twitter.

4. Y el capítulo 5 trata la propuesta multimodal para la predicción del PIB regional a partir de información textual y secuencias de valores históricos de cuatro CCAA.

5. Finalmente, el capítulo 6 presenta las conclusiones generales extraídas de esta tesis y las posibles líneas de trabajo futuro.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 2

Estado del arte

Una vez expuesta la motivación que nos ha llevado al desarrollo de esta tesis, procedemos a revisar los trabajos de la literatura que han servido de inspiración para sentar las bases de nuestro estudio. Empezaremos realizando una introducción al aprendizaje automático y sus orígenes para, posteriormente, centrarnos en el Deep Learning, la multimodalidad y la aplicación de estas técnicas en la economía.

2.1. Aprendizaje Automático

Universitat d'Alacant
Universidad de Alicante

Si bien los orígenes históricos de los principios teóricos del desarrollo de la Inteligencia Artificial (IA) se remontan a la década de los años treinta, es en 1950, con la publicación del artículo *Computing machinery and intelligence* (Turing, 1950) de Alan Turing, cuando se considera el inicio de esta ciencia. Con la simple pregunta "¿pueden las máquinas pensar?" propuso un método, que hoy denominamos "Test de Turing", cuya superación por una máquina se basaba en mantener una conversación con una persona real sin que esta supiera distinguir si hablaba con otra persona o con una IA. Este test sigue considerándose en la actualidad en diferentes estudios e investigaciones. Si bien Alan Turing es considerado el padre de la IA, es en 1956 que esta rama del conocimiento pasa a considerarse una disciplina, cuando John McCarty, Marvin Misky, Nathaniel Rochester y Claude Shannon realizaron su presentación en la Conferencia de Darmouth (McCarthy *et al.*, 2006).

El desarrollo de la IA se ha basado desde sus orígenes en la simulación del comportamiento del ser humano, imitando tanto actividades mentales como físicas. El cerebro procesa la información basándose en nuestra experiencia, adquiriendo conocimiento del entorno con el que interactuamos y aprendiendo con dicha información recibida. Con el objetivo de imitar este comportamiento, es cuando nace el llamado Aprendizaje Automático (*Machine Learning* o ML en inglés) como un subconjunto de técnicas para el desarrollo de IA.

Hasta su puesta en escena, la IA consistía en especificar características y reglas en las que el modelo debía centrarse, teniendo que rediseñar toda la arquitectura frente a cambios de estrategia. La Figura 2.1 representa la distinción entre estas dos aproximaciones. En las técnicas de IA tradicionales se definen manualmente las reglas con las que clasificar la información, mientras que con el Aprendizaje Automático esta perspectiva cambia. El programa recibe como entrada los datos y la respuesta que esperamos obtener (centrándonos de momento en un caso de aprendizaje supervisado, como detallaremos más adelante) y realiza un proceso de entrenamiento para de forma automática extraer las reglas que mejor se adecuen a la clasificación del conjunto de datos considerado.

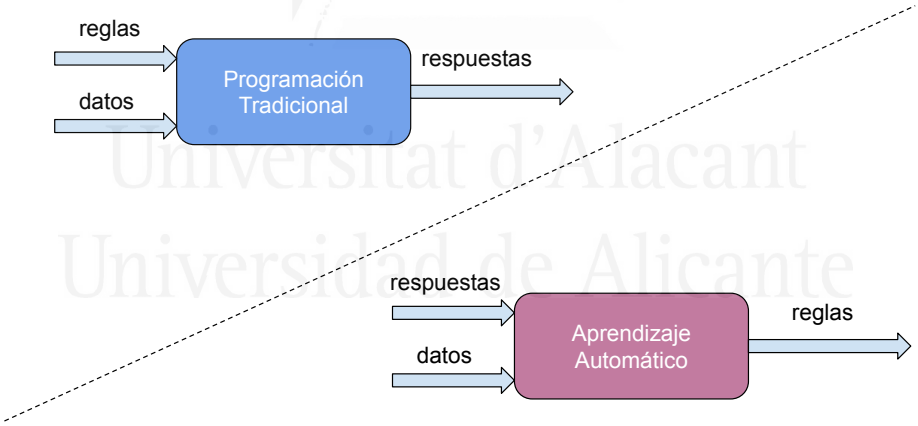


Figura 2.1.: *Programación de IA tradicional frente al Aprendizaje Automático.*

El Aprendizaje Automático comprende por tanto al grupo de técnicas que aprenden de forma automática a clasificar, predecir o procesar un determinado conjunto de datos. Gracias a esta disciplina, se hace posible la construcción de modelos capaces de aprender patrones a partir de un conjunto de datos de entrenamiento que posteriormente permiten clasificar o efectuar predicciones ante nueva información con la que no habían sido entrenados.

Desde su aparición, el Aprendizaje Automático ha avanzado para ofrecer una gran variedad de alternativas que permiten abordar la resolución de problemas de muy diferentes ámbitos. Entre estas técnicas podemos encontrar algoritmos tales como *Random Forest* (RF) (Breiman, 2001), *Support Vector Machine* (SVM) (Vapnik, 1998) o *k-nearest neighbors* (kNN) (Altman, 1992). También encontramos uno de los algoritmos que más ha aportado al desarrollo del aprendizaje automático: las *redes neuronales*, las cuales se explicarán en la siguiente sección debido a su relevancia en esta tesis doctoral.

Como resume la Figura 2.2, esta rama de la IA ha sido empleada en múltiples campos: detección de anomalías en procesos de clasificación, recomendaciones de búsqueda en los navegadores web o en comercios electrónicos, publicidad personalizada, clasificación automática de imágenes, predicción de valores numéricos (problemas de regresión), etc. Una de las áreas donde más popularidad ha tenido es en el ámbito conocido como *Big Data* (L'Heureux *et al.*, 2017). Una empresa con millones de datos (no necesariamente homogéneos ni correctamente estructurados) puede entrenar algoritmos para predecir comportamientos futuros, como predecir el volumen de venta de un determinado producto o conocer el riesgo de concesión de un crédito bancario en función de unos parámetros del usuario, entre otras aplicaciones.

Como ya hemos avanzado, para la generación de un modelo de ML, se realiza un proceso de entrenamiento utilizando un conjunto de datos. En dependencia de si los datos utilizados están etiquetados o no con la respuesta esperada y de la metodología utilizada durante el entrenamiento, podemos clasificar este proceso como:

1. **Aprendizaje supervisado**: Este entrenamiento se produce cuando los datos están etiquetados, es decir, para cada entrada se especifica al sistema la salida o resultado que debe generar. De esta forma, durante el proceso de entrenamiento, se puede comparar la salida generada por el modelo con la salida esperada y ajustar su configuración para reducir el error cometido.

2. **Aprendizaje no supervisado**: En este tipo de entrenamiento los datos no están etiquetados, por lo que no es posible generar modelos para su

Figura 2.2.: *Aplicaciones del Machine Learning en la industria.*

clasificación. En su lugar estas técnicas se centran en buscar patrones comunes para agrupar los datos (*clustering*), buscar similitudes entre ellos o simplificar su estructura para mantener solo las características más relevantes.

3. **Aprendizaje por refuerzo**: Este método de entrenamiento consiste en una estrategia de "recompensa", en la que el sistema aprende a resolver la tarea asignada modificando su conducta a través de prueba y error, sin seguir ninguna forma específica de hacerlo, sino simplemente recompensando las acciones o predicciones correctas.

2.1.1. Redes Neuronales

El principio de pretender imitar las capacidades cognitivas de los seres humanos nos llevó al siguiente paso (Hardy, 2001): ya no se trataba de preguntar

si las máquinas podían aprender, sino de cómo funciona la mente humana; ¿cuáles son los fundamentos neurobiológicos que utiliza nuestro cerebro cuando pensamos? La Figura 2.3 sintetiza el funcionamiento del sistema nervioso de una neurona de nuestro cerebro. Este tipo de células son estimuladas mediante una serie de impulsos nerviosos que envían señales por medio de las dendritas. Estas señales se procesan en el cuerpo de la neurona y, posteriormente, se envía una nueva señal a las siguientes neuronas a través del axón. Las neuronas se conectan en una región llamada sinapsis, en donde una célula (presináptica) se pone en contacto con otra (postsináptica).

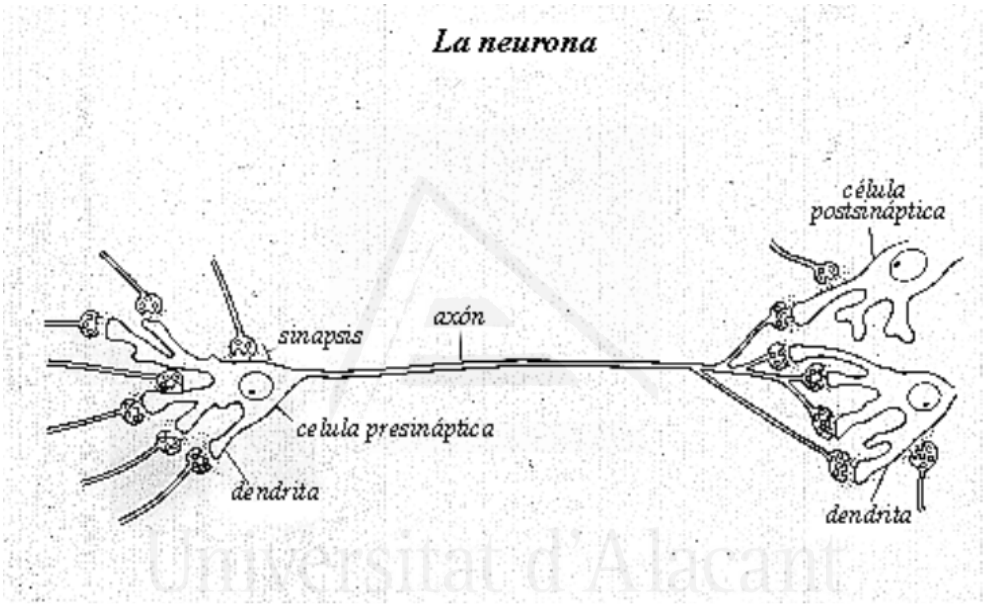


Figura 2.3.: *Representación de una neurona. Los impulsos van de izquierda a derecha. Imagen extraída de Hardy (2001).*

La Figura 2.4 específica el funcionamiento de la sinapsis. Como explica Hardy (2001): "la transmisión eléctrica que proviene del axón pone en marcha la emisión de mediadores químicos en la abertura sináptica, que provoca a su vez una transmisión eléctrica del lado de la dendrita. La célula recibe por medio de sus dendritas un conjunto de señales, que son combinadas en su cuerpo para ajustar su tasa de emisión de señales neuronales en dirección a células vecinas. Las neuronas reciben a la vez señales excitadoras, que incrementan su nivel de *activación*, y señales inhibidoras que reducen ese mismo nivel. Si bien cada neurona recibe tantas señales excitadoras como inhibidoras, emite sin embargo un sólo tipo de señal a la vez".

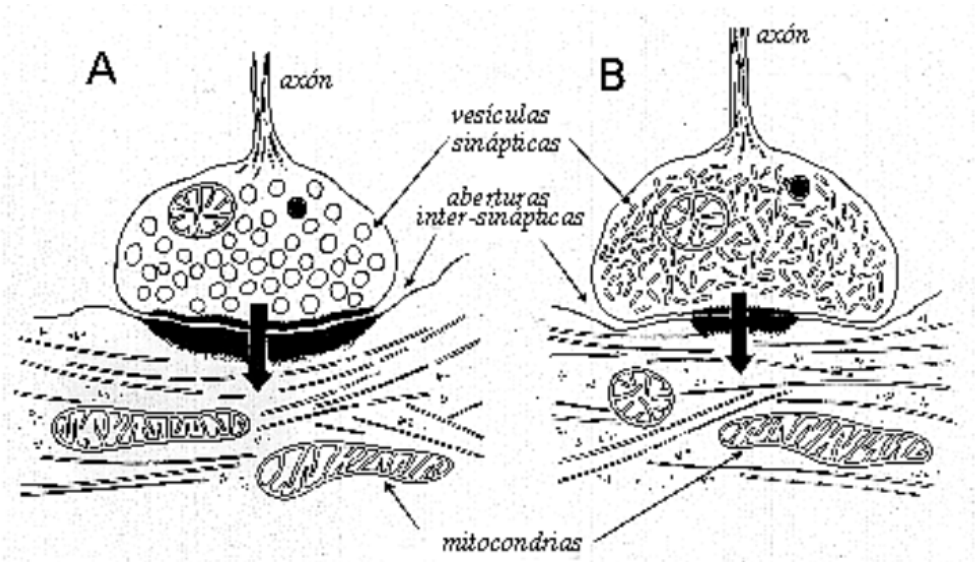


Figura 2.4.: *Los dos principales tipos de sinapsis en el cerebro: excitadora (A) e inhibidora (B). Imagen extraída de Hardy (2001).*

Con el fin de imitar este funcionamiento neurobiológico, se propuso una serie de algoritmos y métodos matemáticos. El primer modelo de red neuronal teórico fue propuesto por McCulloch y Pitts (1943) en términos de un proceso computacional de actividad nerviosa. Posteriormente, Rosenblatt (1958) propuso el primer modelo de neurona artificial, más conocido como *perceptrón*. Y en 1985, Rumelhart *et al.* propusieron el perceptrón multicapa (*Multilayer Perceptron* o MLP).

Un perceptrón multicapa está formado por una serie de capas donde cada una está compuesta por unidades lógicas (neuronas artificiales o perceptrones) que son activadas por las de la capa anterior. El resultado matemático de cada unidad será el encargado de activar las de la siguiente capa y así procesar la información hasta la salida como lo haría un circuito cerebral. Estas arquitecturas MLP, posteriormente denominadas simplemente como Redes Neuronales (*Neural Networks* o NN) se componen de tres partes básicas (como refleja la Figura 2.5: la capa de entrada, una o más capas ocultas intermedias y la capa de salida. Cada neurona está conectada con todas las neuronas de la capa anterior y los enlaces entre las neuronas pueden modificar el estado de activación de las neuronas con las que se conecta. Las capas que conforman un MLP también se

las conoce como "capas totalmente conectadas" (del inglés *Fully Connected* o FC).

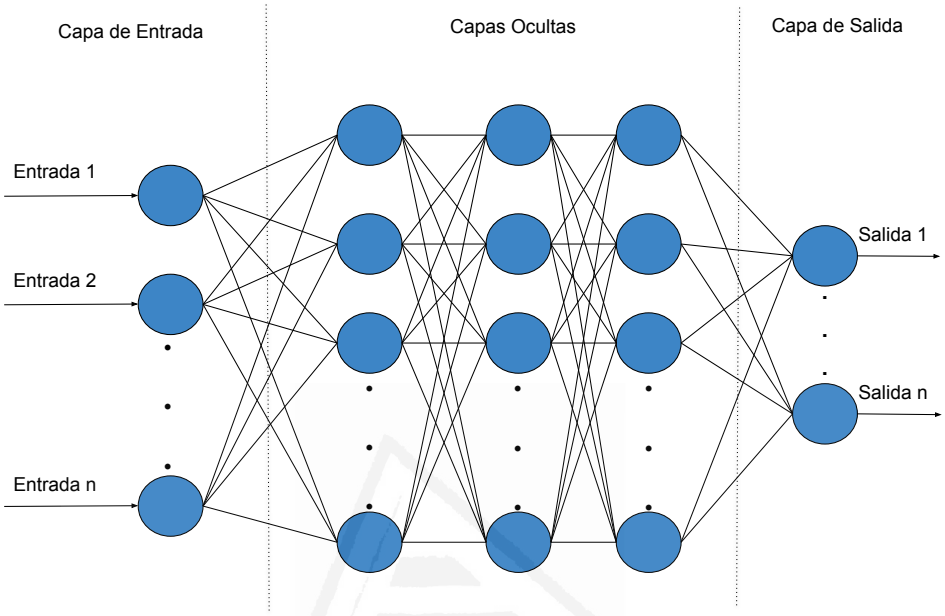


Figura 2.5.: *Ejemplo de una Red Neuronal, donde los círculos representan las neuronas artificiales que componen la red y las líneas las conexiones entre ellas. Imagen elaborada a partir de Bre et al. (2017).*

Universitat d'Alacant
Universidad de Alicante

Como hemos indicado, en estas arquitecturas de red neuronal, cada capa está formada por una o más neuronas artificiales. La Figura 2.6 muestra la estructura de una de estas neuronas. Como se puede ver, la neurona recibe un conjunto de entradas x, correspondientes a los valores recibidos de la capa anterior (o de la propia entrada en el caso de la capa de entrada), pondera los valores recibidos usando los pesos w, suma el resultado y añade un umbral o bias b. La salida de cada neurona se transforma aplicando una función de activación f para que, de esta forma, podamos modificar dicha salida, aplicar funciones lineales o no lineales para restringirla a un rango dado (entre 0 y 1, o entre -1 y 1) o a valores discretos (0 o 1). Hay una gran variedad de funciones de activación, pero las más utilizadas son la función Softmax, la Sigmoidal y la *Rectified Linear Unit* (ReLU) (Glorot *et al.*, 2011).

Para que estas redes aprendan a clasificar un conjunto de datos se realiza un

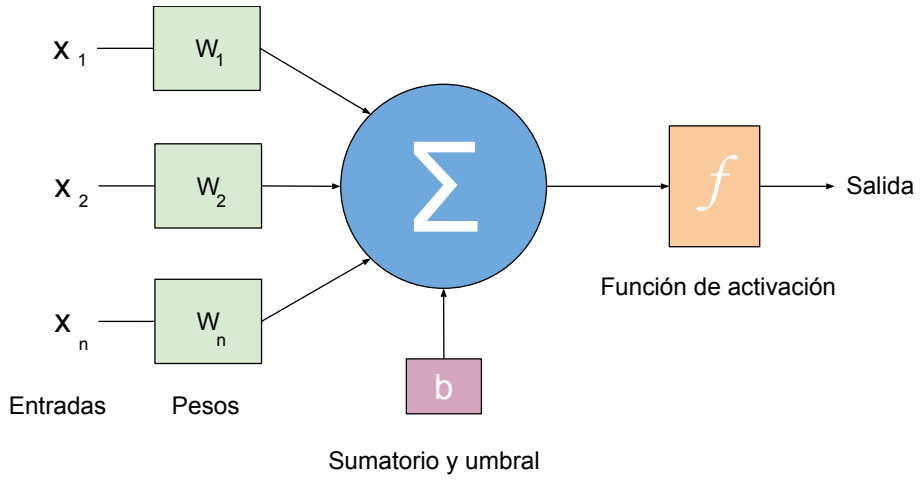


Figura 2.6.: *Ejemplo de una neurona artificial o perceptrón. Recibe x entradas, pondera estas entradas por unos pesos w, suma el resultado y añade un umbral o bias b. Este resultado es modificado finalmente por una función de activación f.*

proceso de entrenamiento. Este proceso básicamente consiste en ir ajustando los parámetros de la red (los pesos w y bias b de los perceptrones) con el objetivo de reducir el error de predicción del modelo para los datos considerados. En los años setenta Werbos (1975) propuso el algoritmo de "propagación hacia atrás de errores" o "retropropagación" (*backpropagation*) utilizado para realizar este proceso de entrenamiento. En poco tiempo, este método se convirtió en el preferente para el entrenamiento de redes neuronales, como pudo constatar el propio Werbos (Werbos, 1990). Este algoritmo consta de dos fases:

1. Paso *forward* o hacia adelante: se suministran las muestras de entrenamiento a la red, se obtiene la predicción y se compara esta salida con la salida deseada. Con esto se puede calcular el error (usando una función de error o *loss*) para cada una de las salidas.

2. Paso *backward* o hacia atrás: El error calculado se propaga hacia atrás desde la capa de salida. Cada neurona recibirá la fracción de error cometido por la misma. Este proceso se realiza utilizando el método de optimización de descenso por gradiente, el cual se basa en calcular la pendiente formada por el error cometido en función de la configuración de parámetros de la red. Esta pendiente se calcula para la configuración y se actualizan los pesos en función del parámetro de aprendizaje o *learning rate*, moviéndonos

Universitat d'Alacant
Universidad de Alicante

en la dirección opuesta al gradiente para ir progresivamente reduciendo el error cometido por el modelo.

Un problema inherente al proceso de entrenamiento de las redes neuronales es el denominado *overfitting* o sobreajuste. El cual consiste en que los parámetros aprendidos durante el entrenamiento se especializan en la detección de patrones muy específicos del conjunto de datos utilizado, en lugar de buscar características generales que puedan aplicarse también ante nuevas muestras no vistas durante el entrenamiento (Burnham y Anderson, 2002). Para solucionar este problema existen diferentes técnicas, aunque la solución más sencilla es considerar un conjunto de datos de entrenamiento variado, con muestras representativas de todos los posibles casos a considerar. De no haber suficientes muestras etiquetadas o no ser lo suficientemente representativas del dominio del problema, se podrían buscar nuevos datos a añadir. Una técnica muy utilizada para incrementar la variabilidad de este conjunto de manera automática es el aumentado de datos (Krizhevsky *et al.*, 2012), que básicamente consiste en generar nuevas muestras aplicando diferentes tipos de transformaciones (por ejemplo, en el caso de imágenes podríamos aplicar rotaciones, traslaciones, cambios de color, etc. sobre las imágenes originales). Desde el punto de vista de la arquitectura de red utilizada también se pueden aplicar distintas técnicas, como por ejemplo reducir el número de capas o neuronas por capa del modelo, para de esta forma reducir el número de parámetros a aprender y forzar a que tenga que generalizar. O también se pueden añadir capas de regularización, como el *dropout* (Hinton *et al.*, 2012), que consiste en desconectar de manera aleatoria un porcentaje de neuronas durante el entrenamiento.

2.2. Deep Learning

Las redes neuronales han ido evolucionando progresivamente durante años gracias a la aparición de nuevas técnicas, como mejoras en el proceso de entrenamiento, nuevos tipos de capas y funciones de activación, etc. Esto, unido al incremento de la potencia de cálculo de los ordenadores y la disponibilidad de grandes cantidades de datos etiquetados con los que entrenar, ha permitido que estas arquitecturas cada vez obtengan mejores resultados.

El *Deep Learning* ha supuesto un gran avance en el desarrollo y evolución de la IA moderna, alcanzando a un punto en el que se obtienen resultados comparables y, en algunos casos, incluso superando el desempeño de los expertos humanos. Ya no solo se han aplicado con éxito en tareas en las que se puede determinar un conjunto formal de reglas (como el ajedrez), sino también en otras más complejas o abstractas, como por ejemplo la visión artificial, el reconocimiento de voz, el procesamiento del lenguaje natural, la traducción automática, la bioinformática, el diseño de fármacos, el análisis de imágenes médicas, la inspección de materiales o los videojuegos, entre otros.

A raíz de una serie de publicaciones realizadas en 2006 (Hinton *et al.*, 2006; Bengio *et al.*, 2006; Ranzato *et al.*, 2007) se demostró cómo era posible entrenar de forma efectiva arquitecturas neuronales con muchas capas. Esto permitió pasar de emular el comportamiento individual de las neuronas biológicas a imitar el sistema de aprendizaje del cerebro, creando una jerarquía de características o conceptos (ver Figura 2.7), de más simples a más complejos, que se aprenden automáticamente en cada una de las capas de estas redes.

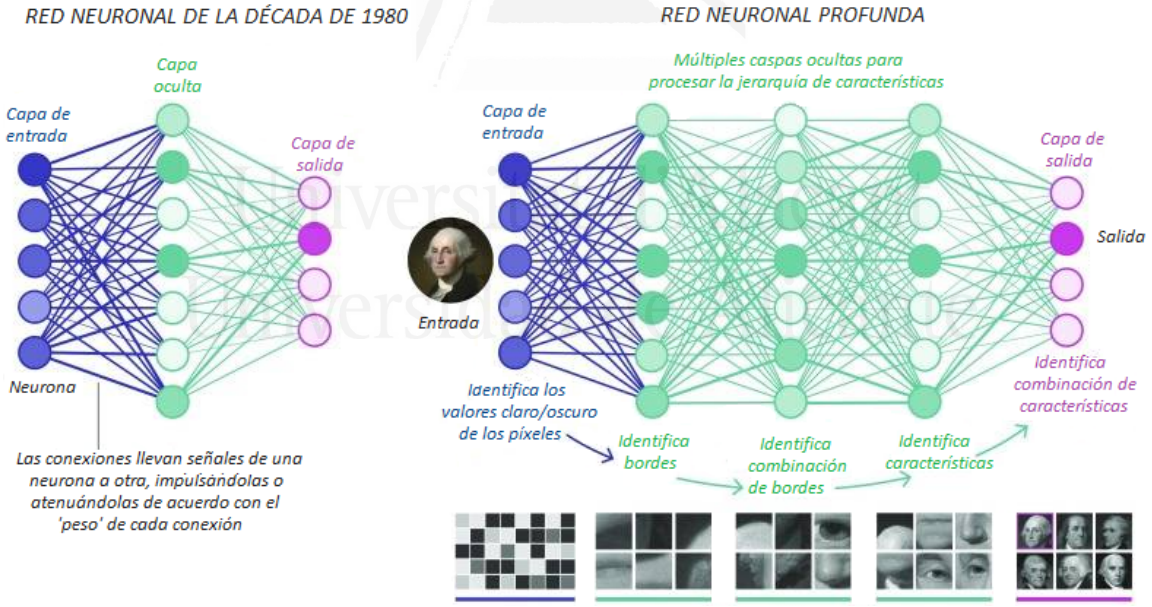
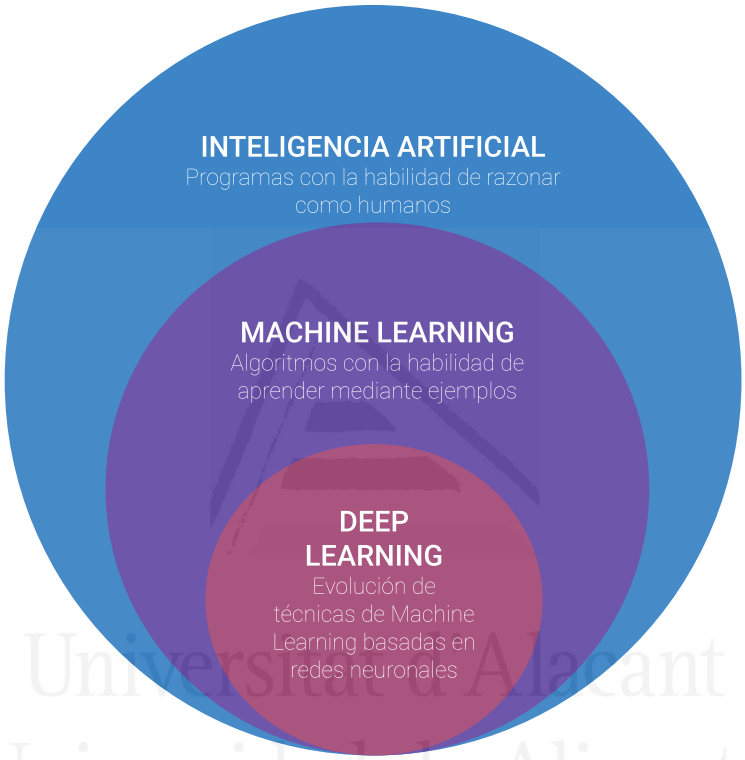


Figura 2.7.: *Comparación entre las primeras redes neuronales con las redes neuronales profundas. Imagen adaptada de Waldrop (2019).*

La Figura 2.8 muestra un esquema organizativo de las distintas técnicas y categorías que podemos encontrar dentro de la IA. Como se puede ver, la IA es el

campo de investigación que engloba al conjunto de métodos capaces de razonar (aparentemente) como humanos. Dentro de este conjunto diferenciamos ML, que sería el subconjunto de los algoritmos con la habilidad de aprender mediante ejemplos. Y a su vez, dentro de este último paradigma, encontraríamos el subconjunto de DL, que comprendería a las técnicas basadas en redes neuronales profundas.



Universitat d'Alacant
Universidad de Alicante

Figura 2.8.: *Esquema organizativo de la IA.*

Para su entrenamiento, al igual que las redes neuronales que explicamos en la sección anterior, se emplea el método de *backpropagation* para ajustar la configuración de parámetros de las capas y de esta forma reducir progresivamente el error cometido por el modelo. Durante el proceso de aprendizaje, este tipo de arquitecturas extraen de manera automática el conjunto de características más apropiado para la tarea que tengan asignada. A este proceso se le conoce como "aprendizaje de características" (o *feature learning*) y en muchas ocasiones permite descubrir características que incluso los expertos no hubiesen seleccionado como relevantes (Goodfellow *et al.*, 2016).

Dentro de DL podemos encontrar diferentes tipos de arquitecturas y metodologías según la tarea que se pretenda resolver, como clasificación o regresión, entre otras. Aplicándose además todos los conceptos que hemos visto anteriormente, puesto que también se pueden entrenar de manera supervisada, no supervisada o mediante aprendizaje por refuerzo. Entre las arquitecturas de red más utilizadas, además de las redes neuronales profundas (*Deep Neural Networks* o DNN) basadas en los MLP vistos previamente, destacar las Redes Neuronales Convolucionales (*Convolutional Neural Network* o CNN) aplicadas principalmente para el procesamiento de imágenes (LeCun *et al.*, 2015), las Redes Neuronales Recurrentes (*Recurrent Neural Networks* o RNN) (Jozefowicz *et al.*, 2015), para el procesamiento de datos secuenciales, o, dentro del paradigma del aprendizaje no supervisado, las arquitecturas tipo autoencoder (Goodfellow *et al.*, 2016). En las siguientes secciones revisaremos con más detalle algunas de estas arquitecturas.

2.2.1. Redes Neuronales Convolucionales

Las Redes Neuronales Convolucionales (*Convolutional Neural Network* o CNN) son un tipo de red neuronal donde las neuronas imitan los campos receptivos de las neuronas de la corteza visual primaria de un cerebro biológico (LeCun *et al.*, 2015). Son una variación de los MLP explicados previamente, sin embargo, debido a que su aplicación se realiza sobre matrices bidimensionales, funcionan muy bien en tareas de visión artificial, como la clasificación, la detección de objetos o segmentación de imágenes.

En general, las redes CNN se componen de una o más capas convolucionales seguidas de una o más capas tipo FC (ver Figura 2.9). Después de cada capa convolucional se suele disminuir su dimensionalidad (aplicando un proceso de *subsampling*) para que las capas más profundas puedan aprender características cada vez más complejas (combinando las características extraídas previamente). De esta forma, en las primeras capas de la red se extraen características de más bajo nivel (bordes, colores, gradientes, etc.) y, progresivamente, se van agrupando y combinando mediante el *subsampling*. Este proceso crea por tanto una jerarquía de características, en la que, las características de alto nivel

extraídas en las últimas capas de convolución permiten discriminar la clase de la que se trata la imagen de entrada usando para esto las últimas capas FC de la red (Goodfellow *et al.*, 2016).

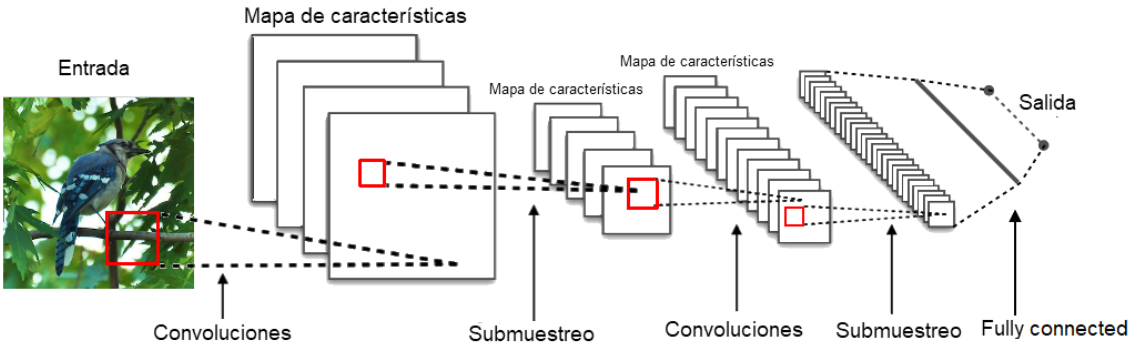


Figura 2.9.: *Esquema general de la arquitectura de red CNN. Imagen adaptada de Wikipedia Commons (2015).*

Para efectuar este proceso, cada capa de convolución de la red aplica una o más funciones de convolución (convolución discreta) sobre los datos que recibe como entrada. En este caso, la entrada será un array 2D de datos (una imagen) y se aplica sobre ella un filtro o *kernel* que devuelve un mapa con las características extraídas de la imagen original. En concreto, cada convolución realiza la siguiente función:

Universitat d'Alacant
Universidad de Alicante

$$y = f\left(b + \sum K \otimes x_n\right) \tag{2.1}$$

donde la salida y es una matriz que se calcula por medio de la combinación lineal de las entradas x_n recibidas de las neuronas en la capa anterior, operadas con el núcleo o *kernel* de convolución K correspondiente, se le añade el bias b y por último se pasa por la función de activación f. Como se puede observar, esta operación aparentemente resulta similar a la que vimos para los perceptrones, pero en lugar de aprender un conjunto de pesos w para ponderar las entradas, en este caso se aprende los pesos del kernel K para realizar la operación de convolución sobre las matrices de entrada y, de esta forma, extraer ciertas características, que estarán determinadas por el kernel utilizado.

Como se ha indicado previamente, después de cada capa (o conjunto de

capas) de convolución se suele aplicar una operación de *subsampling* o *pooling*, cuyo objetivo es reducir el tamaño de las matrices y a la vez quedarse con las características más relevantes de las mismas. Las técnicas más utilizadas para esto son *Max-Pooling* y *Average Pooling*. En ambos casos se aplica un filtro (de dimensiones $w \times h$, habitualmente de 2×2) sobre la imagen de entrada y se guarda el máximo (o la media en el caso del *Average Pooling*) de cada región en la matriz de salida.

Según las dimensiones de los datos de entrada y de los kernels utilizados nos podremos encontrar con redes CNN para procesar tanto datos 2D, como son el caso de las imágenes en el que nos estamos centrando, pero también se pueden utilizar para procesar series temporales o señales de audio utilizando convoluciones 1D (Kiranyaz *et al.*, 2021), así como para la clasificación de datos volumétricos usando convoluciones 3D.

Una de las primeras CNN en aplicarse con éxito en imágenes fue LeNet LeCun *et al.* (1989), utilizada para el reconocimiento de dígitos manuscritos. Dado sus buenos resultados y el potencial mostrado, a partir de este momento casi todos los esfuerzos de investigación en este campo se centraron en este tipo de arquitecturas, dando lugar a la aparición de multitud de publicaciones. A continuación vamos a revisar algunas de las más importantes.

Universitat d'Alacant
Universidad de Alicante

2.2.1.1. AlexNet

En 2012, Krizhevsky, Sutskever y Hinton, publicaron la red conocida como AlexNet (Krizhevsky *et al.*, 2012), ganando el concurso de clasificación de imágenes ImageNet (ImageNet, 2021). Esto supuso un hito para el DL, ya que redujo en un 10 % el error cometido por el resto de métodos tradicionales, lo que suponía una diferencia enorme considerando que el resto de métodos competían por reducir el error en diferencias de como mucho un 1 %.

Para esto emplearon una arquitectura de red compuesta por cinco capas convolucionales y tres capas FC (ver Figura 2.10). Además combinaron varias técnicas que les permitieron entrenar con éxito una de las redes con más capas utilizadas hasta el momento. Con el fin de evitar el sobreajuste del modelo

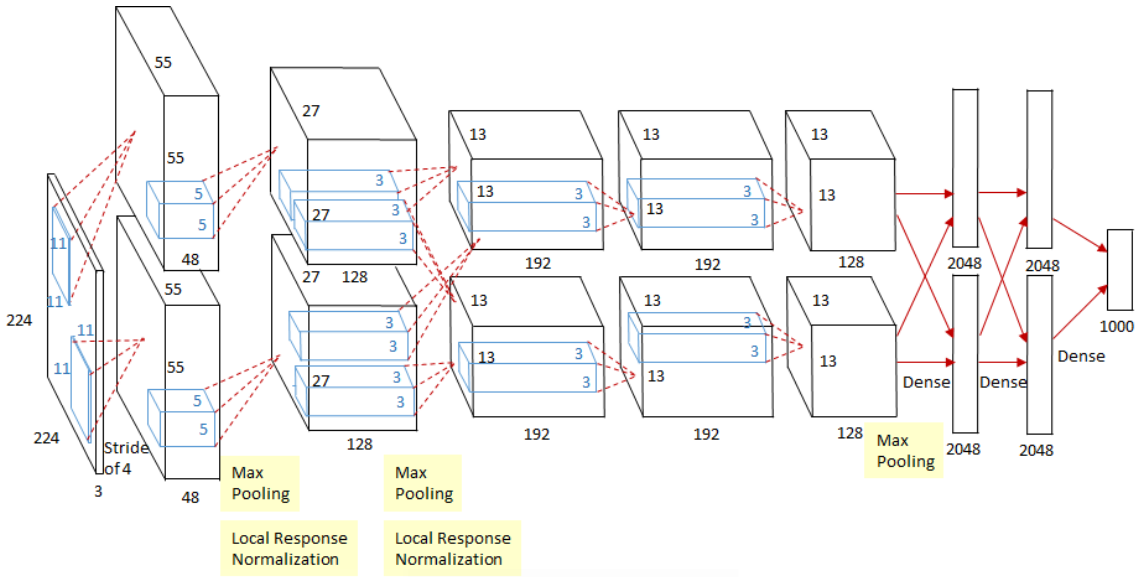


Figura 2.10.: *Ejemplo de la arquitectura CNN denominada AlexNet propuesta por Krizhevsky et al. (2012).*

(*overfitting*), aplicaron técnicas de aumentado de datos y de *dropout* (Hinton *et al.*, 2012). Además, también utilizaron la función de activación ReLU en lugar de la tangente hiperbólica, que era la más común hasta el momento. Y, por último, para conseguir entrenar en un tiempo razonable (ImageNet contiene más de un millón de imágenes y la arquitectura tenía más de 60 millones de parámetros) se utilizaron dos GPU para paralelizar el entrenamiento.

Universitat d'Alacant
Universidad de Alicante

2.2.1.2. VGG-16/19

Las arquitecturas VGG-16 y VGG-19 fueron propuestas por Chatfield *et al.* (2014), como una evolución de AlexNet al añadir más capas. La versión VGG-16 consta de dieciséis capas, trece de convolución con la función de activación ReLU, intercalando algunas capas de *pooling*, más tres capas finales tipo FC con función de activación softmax para la capa de salida. La versión VGG-19 consta de diecinueve capas, dieciséis capas convoluciones, también intercalando capas de *pooling*, más tres FC finales. En todos los casos la arquitectura emplea kernels de tamaño 3×3 y *max pooling* de tamaño 2×2.

2.2.1.3. ResNet

Uno de los inconvenientes de las redes neuronales profundas reside en que durante el proceso de entrenamiento el valor del gradiente tiende a desaparecer o a "explotar" (*vanishing and exploding gradients*). En la etapa de *backpropagation*, cuando se pretende minimizar el error cometido ajustando los pesos de cada capa a partir del cálculo de los gradientes, si estos tienen un valor muy pequeño, la multiplicación de estos hará que progresivamente el gradiente tienda a cero; en cambio, si este valor es grande, sucederá lo contrario y tomará valores que crecerán de manera exponencial. Con el fin de solucionar este problema y poder construir modelos de mayor profundidad, He *et al.* (2015) propuso la arquitectura ResNet (*Residual Network*). Esta red está formada por múltiples *bloques residuales* (ver Figura 2.11) que realizan *conexiones residuales* que saltan varias capas para combinar sus valores. Estas conexiones se repiten a lo largo de toda la red, gracias a lo cual se controla el valor del gradiente y se evita que crezca a valores infinitos o que tienda a cero.

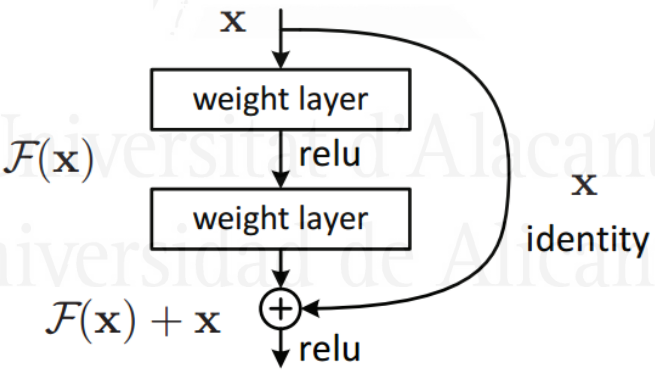


Figura 2.11.: *Ejemplo de un bloque residual en la arquitectura ResNet. Imagen extraída de He et al. (2015).*

La Figura 2.12 muestra la arquitectura completa de una red ResNet para la versión con 34 capas con múltiples conexiones residuales como las que se ilustran en la Figura 2.11. En este trabajo y gracias a este tipo de conexiones, se consiguió entrenar con éxito (y mejora el resultado del estado del arte para ImageNet) versiones de ResNet con hasta 152 capas.

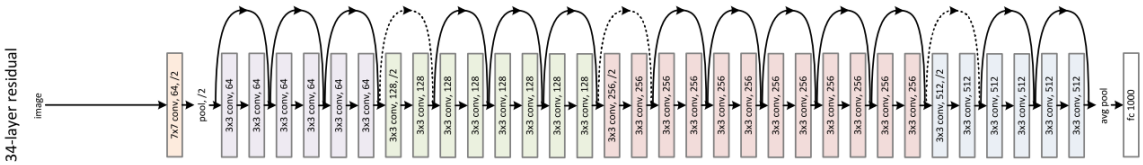


Figura 2.12.: *Arquitectura ResNet con 34 capas propuesta por He et al. (2015).*

2.2.1.4. Inception

Como hemos visto, al aumentar la profundidad de las redes se incrementa el número de parámetros y aparecen problemas para entrenarlas de manera efectiva. Con el propósito de seguir mejorando el resultado de este tipo de redes sin la necesidad de únicamente añadir más y más capas, el equipo de Google propuso la arquitectura Inception (anteriormente llamada GoogleLeNet) (Szegedy *et al.*, 2014). En comparación con las redes CNN convencionales, el módulo Inception procesa el mismo canal mediante convoluciones con diferente tamaño de kernel (1×1, 3×3 y 5×5), cuyos resultados se concatenan y envían a la siguiente capa. Esto permite extraer diferentes características para una misma capa, lo que reduce tanto el número de parámetros como la profundidad de la propia red, resultando en una arquitectura más eficiente.

Universitat d'Alacant
Universidad de Alicante

2.2.2. Autoencoders

Un tipo de arquitectura de red muy específico que hemos utilizado para nuestro trabajo ha sido el autoencoder. Los autoencoders fueron propuestos hace décadas por Hinton y Zemel (1994) y, desde entonces, se han investigado activamente (Baldi, 2012). Consisten en redes neuronales entrenadas para reconstruir su entrada. Por lo general, se dividen en dos etapas: la primera parte (llamada *encoder*) que recibe la entrada y crea una representación intermedia o latente de la misma, y la segunda parte (el *decoder*) que toma esta representación intermedia e intenta reconstruir la entrada. Además, se suele considerar una representación intermedia con una dimensionalidad inferior a la de la entrada. Esto fuerza a que se tenga que aprender una representación reducida con las características esenciales o más relevantes que permitan reconstruir la entrada

con la menor pérdida posible. La Figura 2.13 resume la arquitectura típica de un autoencoder, aunque existe una gran variedad de formas con las que se puede definir para todo tipo de tareas, como compresión de vídeos, imágenes o texto (Bank *et al.*, 2020), como veremos más adelante.

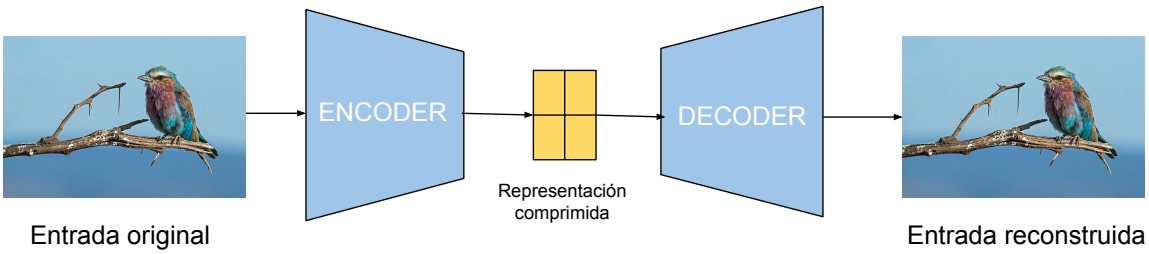


Figura 2.13.: *Esquema típico de un autoencoder.*

2.2.3. Redes Neuronales Recurrentes

Como hemos visto, las redes convolucionales han supuesto un gran avance en tareas como la clasificación de imágenes. Pero existen otro tipo de problemas cuya resolución no puede basarse en usar únicamente la información del instante actual. Por ejemplo, qué está sucediendo en una película viendo solamente una imagen, el rol o significado de una palabra sin consultar su contexto en un texto o, más sencillo, el siguiente valor de una serie basándonos solamente en el valor anterior. La redes neuronales tradicionales no pueden solucionar este tipo de problemas, ya que solo condicionan su salida con la entrada actual o, dicho de otro modo, para una una misma entrada, siempre darán la misma salida.

En estos casos, cuando se requiere trabajar con valores secuenciales o series temporales, disponemos de las Redes Neuronales Recurrentes (RNN) (Jordan, 1986). Estas redes están formadas por *neuronas recurrentes* (Figura 2.14, parte izquierda), las cuales son muy similares a las neuronas artificiales normales, pero con el añadido de un bucle recurrente que les permite mantener un *estado*. Este estado interno de cada neurona irá cambiando en función que lo que ha visto hasta el momento, transmitiendo la información que hace falta para el problema de aprendizaje en cuestión. Es decir, la neurona predice su salida h para el instante de tiempo t (denotado por h_t en la figura) a partir de la entrada x en

ese mismo instante (x_t) y también a partir de la salida del estado anterior (h_{t-1}). También podemos pensar en las neuronas recurrentes de forma "desplegada" (ver Figura 2.14, parte derecha), en la que tendríamos tantas neuronas como elementos de la secuencia, y donde cada una pasa su estado a la siguiente.

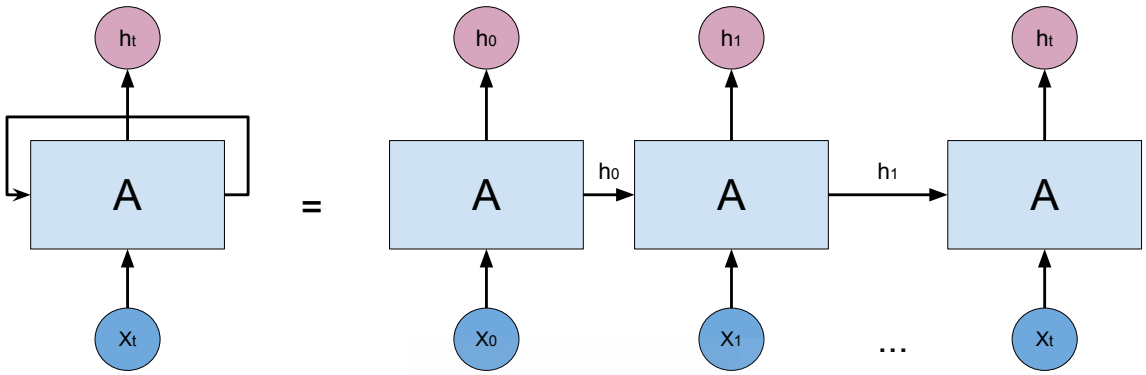


Figura 2.14.: *Ejemplo de una neurona recurrente (izquierda) y su versión desplegada (derecha). Este tipo de neurona predice su salida para el instante h_t a partir de la entrada recibida en ese instante (x_t) y también a partir de la salida del estado anterior ($h_t - 1$). Imagen adaptada de Su y Li (2019).*

En algunas tareas solo es necesario utilizar la información más reciente para calcular la siguiente predicción. Sin embargo, en otras ocasiones se hace necesario consultar más contexto. Por ejemplo, para predecir la siguiente palabra en la frase "Nací en Francia pero a los 12 años me vine a España, [...] por eso es que hablo tan bien el [?]", podría parecer (si usamos el contexto reciente) que la siguiente palabra es un idioma, pero realmente, para saber el idioma correcto, tenemos que utilizar la información del principio de la frase. Este es un problema para las RNN, dado que según aumenta la distancia hasta la información a utilizar, se vuelve más difícil aprender a conectar la información. Cuando la información hay que propagarla muy atrás en el tiempo, el gradiente de las funciones de activación tiende a 0. Esto se conoce con el nombre de *vanishing gradient*, como ya hemos mencionado previamente, y constituye uno de los problemas típicos de las redes neuronales.

Para lidiar con esta problemática surgieron una serie de alternativas más modernas, como GRU (*Gated Recurrent Unit*) (Chung *et al.*, 2014) o LSTM (*Long Short Term Memory*) (Hochreiter y Schmidhuber, 1997; Shi *et al.*, 2015). Ambas propuestas resuelven el problema de forma distinta. Las neuronas tipo

LSTM incluyen las *gates units* (o puertas) que permiten mantener, modificar u olvidar los valores de entrada. La Figura 2.15 muestra el ejemplo de una LSTM, en la que se marcan los distintos tipos de puerta que utiliza: puerta de entrada (*input gate*), puerta de olvido (*forget gate*) y puerta de salida (*output gate*), las cuales ayudan a la red a mantener o eliminar su estado interno. Las neuronas tipo GRU realizan un proceso similar para retener la información, pero simplifican el esquema y, en vez de tres tipos de puerta, utilizan únicamente dos: puerta de actualización (*update gate*) y puerta de reinicio (*reset gate*).

Existe además otra variante de las arquitecturas de red basadas en LSTM, las llamadas BiLSTM (*Bidirectional LSTM* (Huang *et al.*, 2015; Plank *et al.*, 2016)), las cuales hacen un doble recorrido de los valores de entrada (procesándolos en ambos sentidos) y consiguiendo de esta forma mejores resultados cuando se trabaja con información textual.

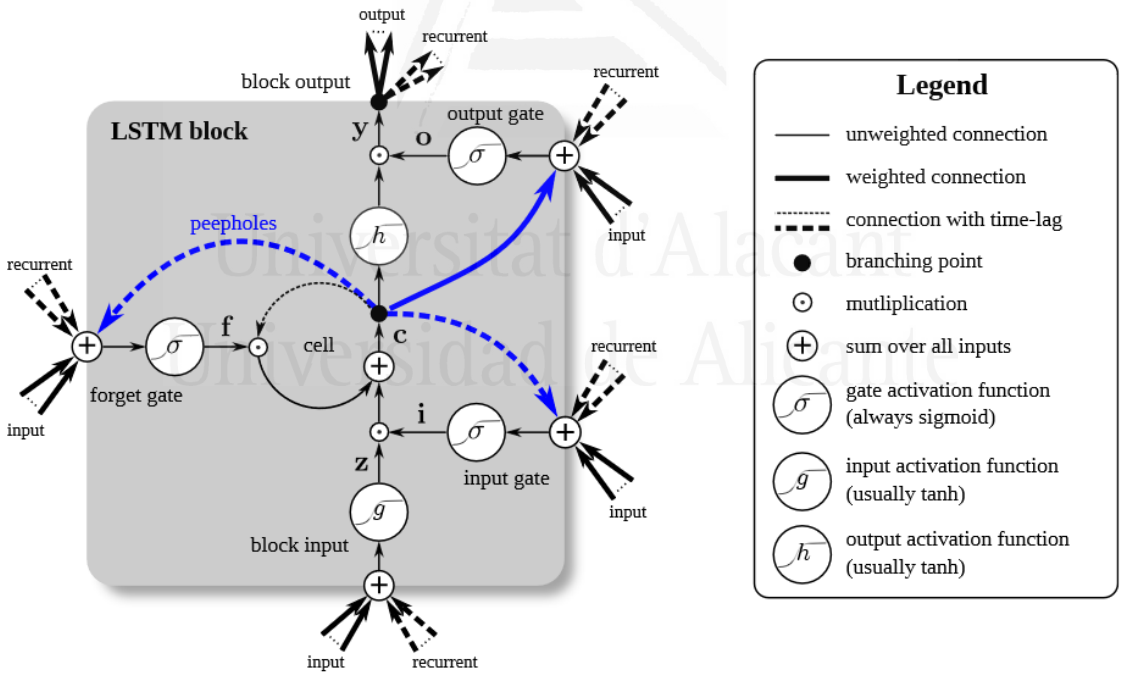


Figura 2.15.: *Esquema detallado de una unidad tipo LSTM. Imagen extraída de Greff et al. (2017).*

2.2.4. Word embedding

A la hora de procesar un texto mediante un algoritmo de ML, se hace necesario codificar las palabras en formato numérico, de manera que el método en cuestión pueda trabajar con ese tipo de información. La opción más simple es la asignación de identificadores únicos a cada palabra, creando para esto un diccionario que permite mapear cada palabra con su correspondiente valor. Sin embargo, utilizar esta solución en métodos de ML puede ocasionar que el método aprenda relaciones entre las palabras a partir del valor de su identificador (lo cual no tiene porque tener sentido o ser directamente erróneo).

Una alternativa bastante utilizada es la codificación *one-hot*, en el que los identificadores son vectores binarios donde todos sus valores valen 0 a excepción de la posición asignada al identificador, que valdrá 1. De esta manera, la distancia siempre es 1 si son diferentes y 0 si coinciden. Sin embargo, esta solución, al igual que la anterior, también conllevaba la pérdida del significado semántico de las oraciones, ya que no existe relación alguna entre las palabras.

Para abordar este problema, existen codificaciones que nos permiten transformar la representación de cada palabra a un nuevo espacio vectorial que sí que mantiene este significado y además nos permite encontrar relaciones entre las palabras. Estas nuevas representaciones son vectores numéricos comúnmente conocidos como *word embedding* o simplemente *embeddings*. En la Figura 2.16 se muestra un ejemplo de esta representación, en las que los *embeddings* que representan palabras como “hombre” y “mujer”, “rey” y “reina”, o “tío” y “tía”, aparecerían representados uno cerca del otro. Además, al preservarse el significado sintáctico y semántico en esta representación numérica, es posible realizar operaciones como, por ejemplo, sumar los vectores numéricos que representan “mujer” y “rey” y obtener un resultado muy cercano a la representación vectorial de “reina”, o encontrar distancias y relaciones similares entre “hombre” y “mujer” que entre “tío” y “tía”.

Existen múltiples propuestas que nos permiten obtener estas representaciones vectoriales, siendo Word2Vec (Mikolov *et al.*, 2013) uno de los más conocidos. El funcionamiento básico de Word2Vec consiste en entrenar un modelo de red neuronal empleando una ventana de tamaño w (*windows size*) para mirar las

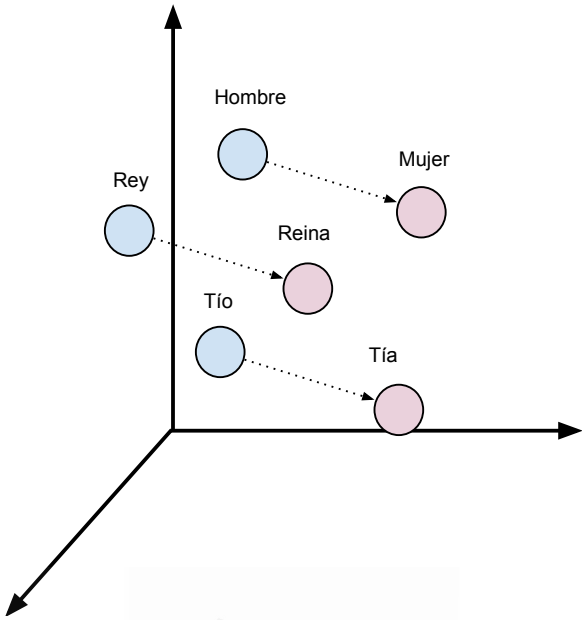


Figura 2.16.: *Muestra de correlación semántica entre palabras utilizando métodos de representación word embedding.*

w palabras anteriores o posteriores en la frase con respecto a la palabra de entrada. Los pesos generados en su capa oculta (llamados *word vectors*) serán los que representen las palabras del corpus.

La aparición de Word2Vec supuso un gran avance en las técnicas de procesamiento del lenguaje natural (*Natural Language Processing* o NLP), ya que permitió mejorar significativamente los resultados obtenidos. Sin embargo, la codificación Word2Vec tiene el inconveniente de perder el significado semántico dentro de la oración en concreto que se esté procesando, ya que las palabras pueden tener un significado u otro en función del contexto de la frase. Con el fin de resolver este problema surgieron nuevas propuestas, como Doc2Vec (Le y Mikolov, 2014) o Sense2Vec (Trask *et al.*, 2015). Tanto Doc2Vec como Sense2Vec son extensiones de Word2Vec y su funcionamiento es muy similar, con el añadido de que ambos métodos extraen vectores de características que representan los párrafos y oraciones considerando el contexto de las palabras. De esta forma, el significado de las palabras ya no solo dependerá de los *word vectors* que las representan, sino de un vector adicional que identifica el párrafo y la palabra en su contexto. Por este motivo, Doc2Vec y Sense2Vec son preferibles a Word2Vec cuando se pretende trabajar con textos u oraciones.

Cabe destacar que trabajos como el de Kumar y Sebastian (2012) nos indican que es preferible que las cadenas textuales, antes de ser codificadas, sean preprocesadas siguiendo una serie de pasos tales como: eliminar caracteres especiales, *stop words* y enlaces, convertir el texto a minúsculas, etc. Para codificar valores textuales, salvo que dispongamos de una gran cantidad de datos para entrenar nuestros propios modelos, es preferible el uso de modelos pre-entrenados, ya preparados para su uso general en distintos casos.

2.3. Aprendizaje Automático en la economía

Los avances en las técnicas de ML y DL han permitido el desarrollo de nuevas y mejores herramientas con las que resolver de manera automática multitud de problemas en diferentes campos, incluido el económico, en el que se centrará parte de esta tesis. Por un lado, y como hemos revisado previamente, estas técnicas pueden procesar datos secuenciales, como cadenas de texto o series temporales (LeCun *et al.*, 2015), lo que resulta muy adecuado para el tipo de datos con el que se suele trabajar en economía. Por otro lado, y como veremos a continuación, en la literatura podemos encontrar muchas propuestas basadas en aprendizaje que mejoran los resultados de los modelos estadísticos que tradicionalmente se han aplicado en esta área (Mullainathan y Spiess, 2017).

En el mundo financiero y económico, la evolución de los datos dependen de una gran cantidad de factores relacionados entre sí y que además se encuentran en un cambio constante. Por lo que, predecir y analizar estas fuentes de información, constituye una problemática no lineal dependiente en el tiempo de múltiples variables con elementos comunes entre ellos; una relación que el DL nos puede ayudar a encontrar (Yu y Yan, 2020).

En el estado del arte podemos encontrar varios trabajos que utilizan ML y DL para predecir magnitudes macroeconómicas, como el Producto Interno Bruto (PIB) o la tasa de desempleo, utilizando para esto indicadores de la economía basados en el conocimiento (*knowledge economy*) (Junoh, 2004; Richardson *et al.*, 2018). Como ya indicamos en la introducción, la vasta mayoría de los estudios de ML o DL en el mundo económico se han centrado en la predicción de

la variación de precios en el Mercado de Valores (Nelson *et al.*, 2017; Roondiwala *et al.*, 2017; Moghar y Hamiche, 2020). En estos trabajos puede apreciarse cómo el empleo de redes recurrentes tipo LSTM es el predilecto dado el carácter secuencial de los datos. Aunque también encontramos propuestas interesantes mediante el uso de otras arquitecturas, como en Sezer y Ozbayoglu (2018), en la que se transforman los datos 1D del análisis técnico financiero en imágenes 2D etiquetadas con las acciones "vender", "comprar" o "mantener" para su posterior clasificación mediante una CNN.

Todas estas propuestas nos pueden servir de ayuda a la hora de comprender o detectar de manera automática patrones en actividades financieras. Por ejemplo, para predecir de esta forma los puntos críticos que determinen los cambios en una tendencia (Shynkevich *et al.*, 2017; Feng *et al.*, 2018) o las dinámicas latentes en los datos (Selvin *et al.*, 2017). Además de todas estas propuestas, principalmente centradas en el Mercado de Valores, las diferentes técnicas de DL disponibles también nos permiten enfocarnos en muchas otras áreas de la economía en las que el uso de estas herramientas ha sido poco estudiado.

2.3.1. Redes sociales como reflejo económico

Estudios recientes, como los trabajos de Llorente *et al.* (2015) y Bokányi *et al.* (2017), han demostrado que el clima económico puede verse reflejado en la actividad de los usuarios en las redes sociales. Estos dos trabajos analizan la incidencia del desempleo en regiones de España y Estados Unidos a partir de los flujos de movilidad (Figura 2.17) y la actividad en las redes sociales. En particular, plataformas como Twitter proporcionan enormes bases de datos de información con la actividad constante de sus usuarios. La aplicación de técnicas de NLP, como el análisis de sentimientos (Joshi y Deshpande, 2018; Hassan *et al.*, 2020), permite capturar información relevante contenida en las opiniones compartidas por los usuarios. Xing *et al.* (2019) demuestra que es posible extraer los sentimientos de estas opiniones para determinar el estado de ánimo de la sociedad en relación con las perspectivas del Mercado de Valores y la variabilidad de los precios. Xing *et al.* (2018) realiza un estudio similar pero para la previsión financiera.

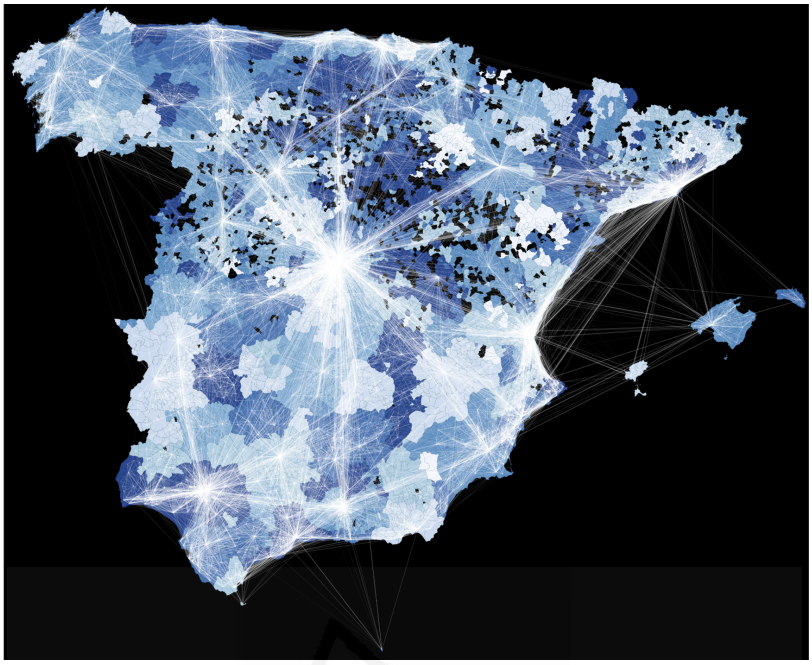


Figura 2.17.: *Mapa con los flujos de movilidad entre municipios de España basados en la actividad de Twitter. Imagen extraída de Llorente et al. (2015).*

El DL también se puede utilizar para encontrar correlaciones entre la información contenida en los tweets, el índice del Mercado de Valores y los cambios en los precios de las acciones de una empresa (Pagolu *et al.*, 2016). En definitiva, el uso de las redes sociales combinado con técnicas de DL puede aportar información muy valiosa a tener en cuenta, ya no solo por mejorar los resultados de los enfoques tradicionales, sino también por proporcionar información casi en tiempo real sobre el estado de la economía.

Dado que los mensajes en Twitter se pueden encontrar en diferentes idiomas incluso para una misma región, se hace necesario traducirlos a un idioma común o bien aplicar técnicas que soporten los diferentes idiomas considerados. Estudios recientes, como Wehrmann *et al.* (2017), han propuesto soluciones que calculan representaciones o *embeddings* genéricos para múltiples idiomas. Por otro lado, algunos otros trabajos (Zhao *et al.*, 2017; Gencoglu, 2018; Gu *et al.*, 2019) han demostrado también que las arquitecturas tipo autoencoders son una buena opción para generar representaciones comunes a partir de entradas en formato texto y de esta forma agrupar (crear *clusters*) con la información contenida.

2.3.2. Predicción del PIB

Una de las vertientes con más interés para aplicar este tipo de métodos son las estadísticas oficiales de agregados macroeconómicos relevantes, como el PIB o la tasa de desempleo, ya que resultan cruciales para la formulación de políticas y la toma de decisiones. Los enfoques clásicos para la predicción del PIB y otras variables económicas se han centrado principalmente en el uso de modelos autorregresivos (AR) (Mills y Mills, 1990), como los modelos AR integrados de media móvil (*Autoregressive Integrated Moving Average* o ARIMA), los modelos tipo *Threshold* AR (TAR) (Hansen, 2011) o las ampliaciones posteriores para incluir regresores exógenos (*Nonlinear Autoregressive with exogenous variables model*, NARX) (Billings, 2013). Para efectuar los pronósticos se han utilizado muchas versiones de estos modelos, que están cubiertos por el término genérico de modelos Vectoriales AR (VAR). El Modelo de Factor Dinámico (*Dynamic Factor Models*, DFM), descrito en Stock y Watson (2016), es una variante particularmente exitosa. Los bancos centrales de todo el mundo utilizan habitualmente este marco para pronosticar el PIB nacional.

Sin embargo, y hasta donde sabemos, el trabajo de Cuevas *et al.* (2015) es la única propuesta orientada a predecir el PIB a nivel regional y con una frecuencia trimestral. Este procedimiento es implementado por la Autoridad Independiente de Responsabilidad Fiscal (AIReF) en España. La primera y principal etapa de este método implica la estimación de un DFM utilizando varios indicadores a nivel regional. Esto permite construir un *rastreador* del PIB que combina la evolución cronológica del PIB regional y la de los indicadores utilizados con la información a corto plazo que estos mismos indicadores transmiten. En la siguiente etapa del método, este rastreador se ajusta para garantizar la coherencia tanto con el PIB regional anual como con el PIB trimestral nacional.

Como explican en Jung *et al.* (2018), la mayoría de los modelos de pronóstico tradicional "se basan en ajustar los datos a una relación previamente especificada entre las variables de entrada y salida, asumiendo así un proceso funcional y estocástico específico subyacente a ese proceso". Los algoritmos de ML y DL nos permiten la detección automática de estas relaciones, encontrando patrones y características a partir de un conjunto de datos de entrenamiento, lo que ofrece un enfoque distinto, basado en el aprendizaje, y que nos permite utilizar otras

fuentes de datos (como ya veremos más adelante) para la resolución de este tipo de problemas.

Distintas propuestas del estado del arte han demostrado que los métodos de ML y DL pueden mejorar los pronósticos de PIB en comparación con los enfoques estadísticos tradicionales. Por ejemplo, Junoh (2004) propuso un método para predecir el PIB de Malasia utilizando cuatro indicadores económicos (usuarios de teléfonos móviles, usuarios de Internet, número de ordenadores y ordenadores personales por cada mil habitantes). En este trabajo se entrenó una red neural tipo MLP con estos indicadores para predecir el PIB. Del mismo modo, Richardson *et al.* (2018) empleó técnicas de ML para predecir el PIB de Nueva Zelanda utilizando datos macroeconómicos, como encuestas comerciales, el índice de precios de consumo, estadísticas comerciales nacionales e indicadores generales de la actividad nacional (como por ejemplo la producción de hormigón, los gastos de tarjetas de crédito y débito, etc.). Este trabajo compara varios métodos de ML, incluidos los modelos AR, SVM, MLP y DFM. Otro trabajo similar basado en indicadores macroeconómicos es el propuesto por Cicceri *et al.* (2020) para predecir el PIB de Italia. Utiliza registros de 1995 en adelante obtenidos de diferentes fuentes oficiales (Eurostat, Banco de Italia, etc.), y variables como la tasa de inflación, producción industrial, tasa de desempleo o la deuda nacional. También compara diferentes enfoques de ML, incluidos modelos AR lineales y no lineales, NARX, *Ordinary Least Squares regression Support Vector Regression* (SVR), kNN y *Boosted Trees*. Destacar también el trabajo de Jung *et al.* (2018), el cual predijo el PIB de 7 países (Alemania, México, Filipinas, España, Reino Unido, Estados Unidos y Vietnam) entrenando varios algoritmos de ML con indicadores económicos proporcionados por el Fondo Monetario Internacional (FMI) y la Guía Internacional de Riesgo País.

Sin embargo, todos estos métodos se enfocan en predecir el PIB solo a nivel nacional y basándose en indicadores macroeconómicos que no están disponibles a nivel regional o en periodos trimestrales. Además, también es importante considerar la periodicidad de publicación de estos indicadores, ya que se publican meses después del final del trimestre, lo que implica que para utilizarlos y poder realizar una predicción es necesario esperar meses hasta obtener este valor. Esto es importante destacarlo, ya que uno de los objetivos de esta tesis es la de proponer un método capaz de predecir el PIB a nivel regional y a frecuencias

trimestrales (o incluso a frecuencias inferiores), por lo que no será posible basarse en los indicadores económicos habitualmente utilizados para esto.

También podemos encontrar propuestas basadas únicamente en datos históricos que intentan solucionar este problema. Por ejemplo, Jaromír (2016) predice el PIB de los países de la zona euro utilizando series temporales de valores del PIB proporcionados por el Banco Mundial. Este trabajo compara MLP y las redes de base radial (*Radial Basis Function networks* o RBF), además generando varias topologías aleatorias para cada una para determinar la mejor configuración. La precisión de este tipo de aproximaciones es, sin embargo, baja, ya que los datos históricos solo permiten la estimación de la tendencia, pero no captan las posibles fluctuaciones en el precio que puedan ocurrir en el trimestre.

Como ya se abordó previamente, Twitter pone a nuestra disposición una enorme base de datos que almacena la actividad constante de sus usuarios. Algunos estudios han demostrado que este hecho hace que esta red social resulte perfecta para descubrir, casi en tiempo real, la actividad económica de un país, o de cualquier otra zona. Por ejemplo, el trabajo de Indaco (2020) estima el PIB de diferentes países del mundo utilizando el volumen de mensajes e imágenes publicados. En este caso se obtiene una baja precisión, ya que esta información es insuficiente por sí misma al no incorporar la tendencia de este valor en los últimos meses, y no puede, por tanto, capturar su valor real.

Por su relación con parte de la experimentación realizada en esta tesis, destacar también el trabajo de Carrillo-Larco y Castillo-Cara (2020), recopilado en Bullock *et al.* (2020), donde se estudian aplicaciones IA en entornos marcados por el COVID-19. Si bien este trabajo no predice cómo ha impactado la pandemia en el crecimiento económico, sí utilizan varios indicadores (siendo el PIB per cápita uno de ellos) de 155 países para hacer *clusters* en dónde englobar a las naciones según el efecto del virus. Más recientemente también podemos encontrar la propuesta presentada por Kim *et al.* (2021), la cual mide el futuro impacto económico del COVID-19 entrenando para esto una arquitectura de red tipo LSTM con datos de pandemias previas y con la tendencia económica actual para predecir de esta forma la tasa de inflación.

Universitat d'Alacant
Universidad de Alicante

2.4. Multimodalidad

Como hemos visto, existe un amplio abanico de técnicas y soluciones aplicables al ML y al DL, principalmente enfocadas en permitir su uso con diferentes tipos de datos y en maximizar su potencial para mejorar los resultados obtenidos. Una de estas técnicas es la multimodalidad (Ngiam *et al.*, 2011), la cual, dado su particular interés para esta tesis, vamos a revisar en esta sección.

La multimodalidad consiste en combinar la información almacenada en varias fuentes de datos para, de esta forma, intentar mejorar el resultado obtenido. Frente a la estrategia clásica de usar un único tipo de dato (unimodal) y entrenar el algoritmo con él para realizar la acción (por ejemplo, entrenar una red neuronal con imágenes para clasificar en categorías), es de esperar que, si contamos con información adicional almacenada en otras fuentes de datos (por ejemplo, y siguiendo el ejemplo de clasificación de imágenes, los metadatos textuales o numéricos de la cámara que tomó la fotografía), la combinación de estos ofrezca más información y por tanto un mejor resultado.

Al analizar la literatura relacionada con el campo de la multimodalidad podremos observar que el conjunto de técnicas y aproximaciones utilizadas ha ido evolucionando con el paso de los años. En concreto, podríamos dividir su evolución histórica en cuatro etapas (Baltrušaitis *et al.*, 2019):

1. *La era del comportamiento (1970-1984)*: El inicio del estudio de la multimodalidad se centró en el ámbito de los comportamientos psicológicos cuando Lazarus (1976) presentó un enfoque psicoterapéutico multimodal para la evaluación de la personalidad de un paciente empleando siete modalidades: comportamiento, afecto, sensación, imágenes, cognición, relaciones interpersonales y drogas. Durante los primeros años de la década de los ochenta se presentaron sistemas de integración multimodales para la detección de signos (Mulligan y Shaw, 1980), la detección de relaciones entre la rigidez de los objetos con el tipo de sonido que emiten (Bahrick, 1983) y una evaluación multimodal de los déficits cognitivos y conductuales en niños en edad preescolar maltratados o abandonados (Hoffman-Plotkin y Twentyman, 1984).

2. *La era computacional (1984-2000)*: Motivados por el efecto McGurk (un efecto que produce que nuestro cerebro sea capaz de inferir un sonido en función del movimiento de los labios), múltiples estudios se enfocaron en el reconocimiento de voz mediante vídeos (*Audio-Visual Speech Recognition, AVSR*). El primer sistema AVSR fue desarrollado por Petajan (1984), aunque en los años siguientes se produjo una rápida mejora en el procesamiento de contenidos multimedia (Yao Wang *et al.*, 2000; Potamianos *et al.*, 2003). Durante estos años también se popularizó los sistemas multimodales HCI (*Human-Computer Interaction*), donde se buscaba diseñar nuevos sistemas informáticos donde los humanos interactuaban a través de múltiples modalidades de entrada y salida (Fels y Hinton, 1993).

3. *La era de la interacción (2000-2010)*: En esta etapa los sistemas multimodales se centraron en modelar los comportamientos interactivos propios de los humanos, como los proyectos CALO (*Cognitive Assistant that Learns and Organizes*) (Mark y Perrault, 2005) o SSP (*Social Signal Processing*) (Ochoa *et al.*, 2006), para la creación de asistentes personalizados que aprendían a través de nuestra interacción con ellos. Así como otras tareas de modelado multimodal para la segmentación de datos en contenidos multimedia (Bengio, 2004) o para la detección de gestos a partir de secuencias de vídeos (Morency *et al.*, 2007).

4. *La era del Deep Learning (2010-actualidad)*: Como ya comentamos, la aparición del DL supuso un gran avance en el desarrollo de la IA y, por su puesto, esto también influyó en el desarrollo de los sistemas mutimodales. Esta era se encuentra en auge gracias a los avances tecnológicos, como el desarrollo de unidades GPU más eficientes para soportar una mayor carga computacional, y a una mayor disponibilidad de información. Esto ha permitido la creación de redes neuronales más complejas para el desempeño de múltiples tareas de manera más eficiente y con mucho mejores resultados (Raina *et al.*, 2009). El primer trabajo que presentó un sistema DL multimodal fue el propuesto por Ngiam *et al.* (2011) para la reconstrucción de vídeo y audio. Posteriormente han surgido multitud de trabajos que han aprovechado todo este potencial para el desarrollo de nuevas soluciones en todo tipo de tareas (Srivastava y Salakhutdinov, 2012; Ma *et al.*, 2015; Xu *et al.*, 2016a).

Universitat d'Alacant
Universidad de Alicante

En esta tesis doctoral nos centraremos en esta última etapa de la multimodalidad, estudiando sus ventajas y debilidades al integrarse con diferentes técnicas de DL para la resolución de problemas en los que se pueda aprovechar la combinación de varias fuentes de información. A continuación vamos a analizar los diferentes tipos de clasificación que podemos hacer de la multimodalidad en función de la estrategia utilizada para su aplicación.

2.4.1. Clasificación de la multimodalidad

La multimodalidad en DL puede clasificarse en cinco grupos principales según los desafíos a tener en cuenta a la hora de su uso (Baltrušaitis *et al.*, 2019):

1. *Representación*: Un punto fundamental a la hora de emplear diferentes fuentes de información es su representación. Cuando queremos usar varias modalidades, podemos optar por una representación conjunta de todas ellas que sea capaz de englobar a todos sus elementos (Figura 2.18 (A)). Pero esta opción plantea la dificultad de cómo representar modalidades distintas que se almacenan en formatos distintos (por ejemplo, el lenguaje se representa con símbolos alfanuméricos, mientras que el sonido o el audio con señales). Una alternativa más conveniente cuando trabajamos con DL es hacer uso de un medio que represente cada modalidad de forma más precisa y encontrar una representación coordinada entre ambas (Figura 2.18 (B)). Para este caso, Andrew *et al.* (2013) propuso una aproximación matemática para aprender las proyecciones lineales correlacionadas entre las representaciones (Figura 2.19).

2. *Alineación*: Otro desafío que presenta la multimodalidad es cómo identificar las relaciones entre los elementos que representan las modalidades para medir sus similitudes y dependencias. Existen dos maneras de abordarlo: de una forma explícita o de una forma implícita. La alineación explicita tiene como objetivo encontrar directamente correspondencias entre elementos de las modalidades, mientras que la implícita emplea una alineación interna que consigue encontrar estas relaciones. Un ejemplo de ello es el trabajo realizado por Karpathy *et al.* (2014), el cual propone un *embedding*

Universitat d'Alacant
Universidad de Alicante

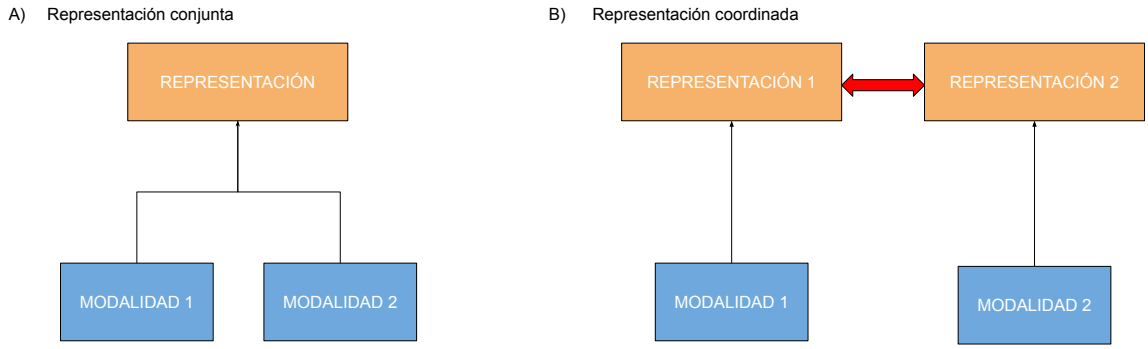


Figura 2.18.: *Desafío de la representación multimodal. (A) Representación conjunta. (B) Representación coordinada.*

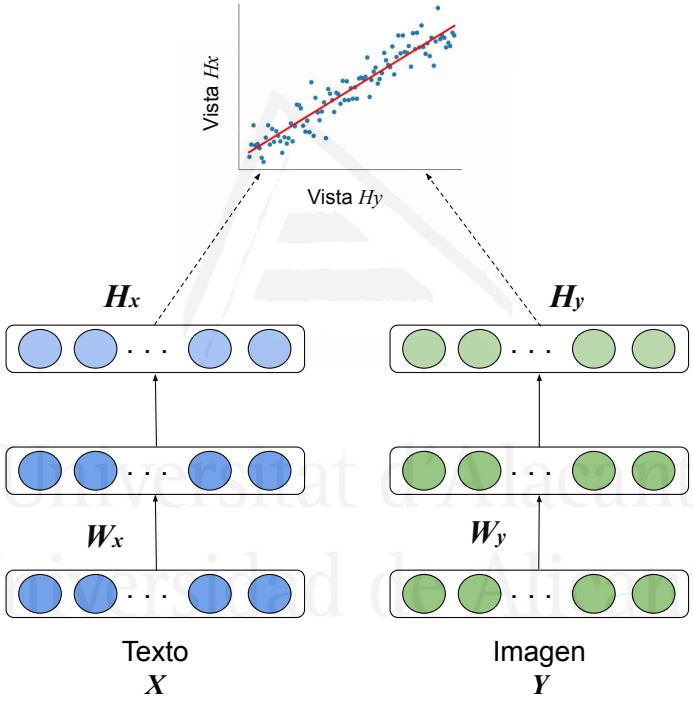


Figura 2.19.: *Aprendizaje a partir de la proyección lineal correlacionada entre las representaciones de varias modalidades propuesto por Andrew et al. (2013).*

multimodal para encontrar las correlaciones entre imágenes y oraciones extrayendo fragmentos de cada modalidad para relacionarlos en un espacio común (ver Figura 2.20).

3. *Fusión*: Uno de los elementos más importantes a la hora de diseñar

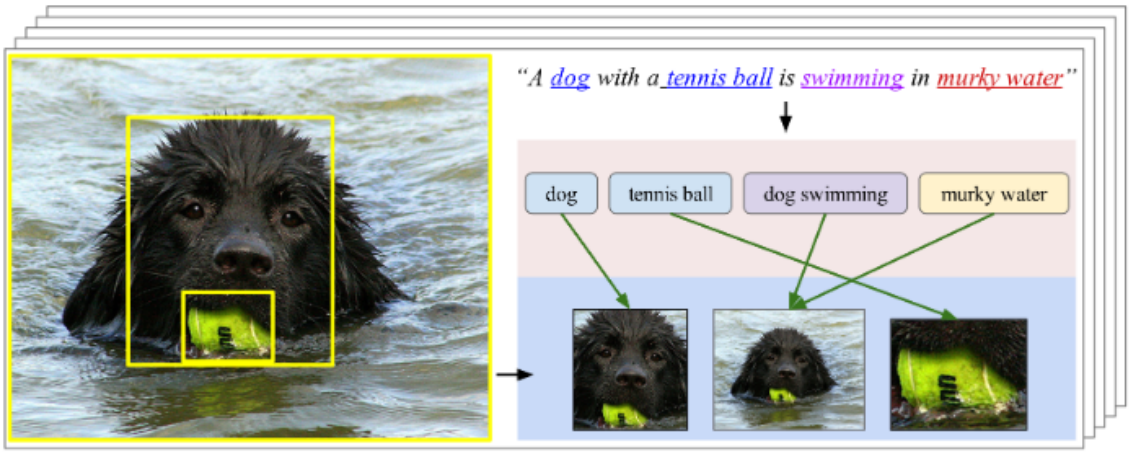
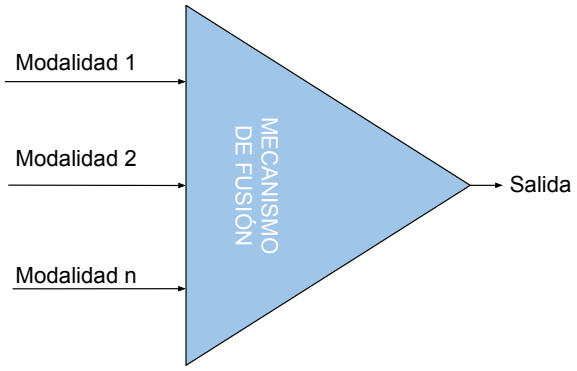


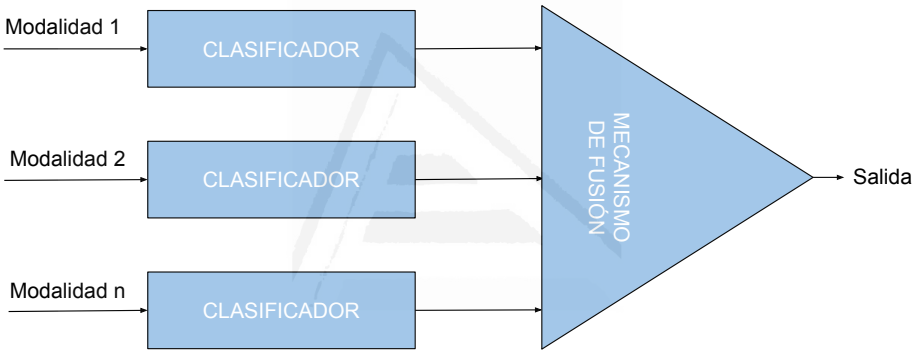
Figura 2.20.: *Ejemplo de cómo el modelo multimodal propuesto por Karpathy et al. (2014) aprende a asociar fragmentos de imágenes con oraciones para encontrar las relaciones.*

arquitecturas multimodales es decidir cómo se combinará la información de las distintas modalidades empleadas. La fusión de la información plantea varios desafíos, tales como: el ruido generado en la fusión, el desbalanceo de datos entre las fuentes, diferentes frecuencias de muestreo o una mayor calidad de información de una modalidad frente a otra/s. Para afrontar la fusión se distingue principalmente entre dos tipos de estrategias: las basadas en modelos (como las NN o los *Multiple Kernel Learning*, MKL (Gönen y Alpaydin, 2011)), que realizarían el aprendizaje de las distintas fuentes de forma integrada en el propio modelo, y los enfoques agnósticos (D'mello y Kory, 2015) o independientes del modelo, en los que se procesan las fuentes de información por separado y se aplican mecanismos de fusión para combinar los resultados. Los enfoques agnósticos, como representa la Figura 2.21, se dividen a su vez en tres arquitecturas (Atrey *et al.*, 2010): los que realizan primero una combinación de la información de las distintas modalidades con las que entrenar, y el resultado de dicha fusión se utiliza después para realizar la clasificación/predicción (denominada fusión temprana o *early fusion*); o las que en primer lugar efectúan la clasificación/predicción con la información individual de cada modalidad y acto seguido realizan una fusión de los resultados obtenidos (fusión tardía o *late fusion*). También existe una tercera opción híbrida en la que se

combinan las dos opciones anteriores, como las propuestas por Wu *et al.* (2005) y Lan *et al.* (2014).



(a) *Early fusion*



(b) *Late fusion*

Figura 2.21.: *Arquitecturas típicas de los métodos de fusión en modelos multimodales.*

4. *Traducción*: Este desafío hace referencia al proceso de intercambio de datos de una modalidad a otra. Si dos modalidades tienen relación entre sí (y usando de ejemplo el objetivo de hallar relaciones entre imágenes y oraciones de Andrew *et al.* (2013)), a partir de una imagen deberíamos ser capaces de mapear su representación equivalente en la modalidad textual y viceversa. Existen dos formas de lograrlo: fundamentar la traducción en algoritmos basados en un diccionario de ejemplos (opción limitada a los datos del entrenamiento) o los modelos generativos basados en un modelo. Esta segunda opción consiste en generar un modelo entrenándolo con los datos de la modalidad que deseamos traducir y usando como salida la modalidad destino a la que se desea traducir (ver Figura 2.22). Un ejemplo que representa de forma fehaciente este tipo de traducción es el propuesto

Universitat d'Alacant
Universidad de Alicante

por Marsella *et al.* (2013). En este trabajo se genera la interpretación de un personaje virtual en 3D a partir de una señal de audio mediante las propiedades acústicas y semánticas del enunciado. El sistema es entrenado con texto para determinar el contenido semántico y, a partir de él, realizar un análisis del tono o estado de agitación para generar expresiones faciales en el personaje virtual que represente dichos sentimientos.

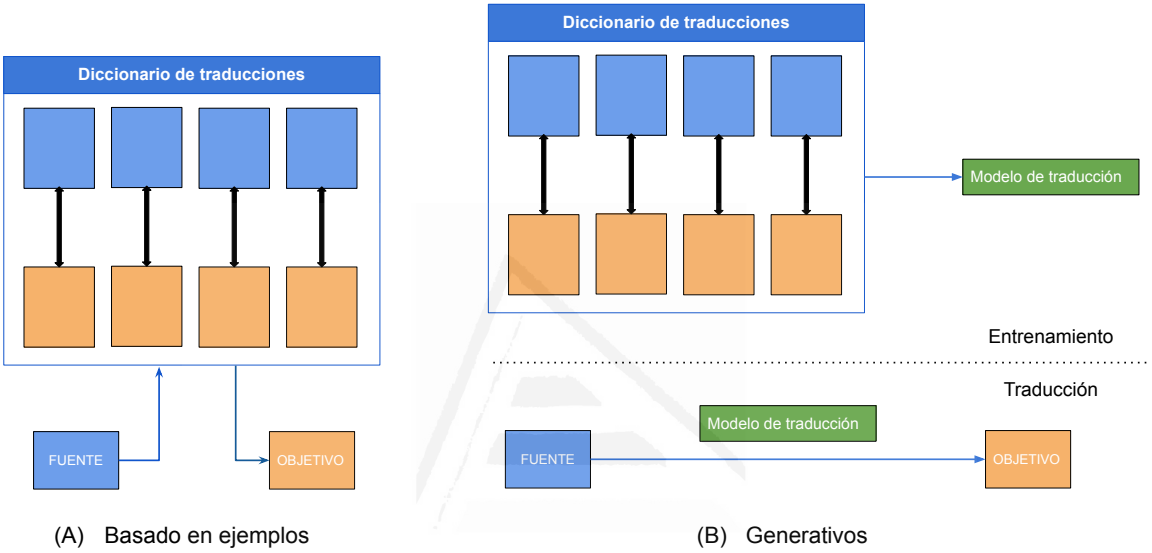


Figura 2.22.: *(A) Modelo de traducción basado en ejemplos. (B) Modelo de traducción generativo. Imagen elaborada a partir de Baltrušaitis et al. (2019).*

5. *Coaprendizaje*: Este último punto representa un elemento clave en cualquier sistema multimodal, ya que hace referencia a la capacidad que debe tener el modelo de transferir el conocimiento entre modalidades, incluidas sus representaciones y modelos predictivos. De esta forma, lo que se pretende es que una modalidad limitada en recursos (con, por ejemplo, pocos datos etiquetados o con una considerable cantidad de ruido en ellos) se pueda ayudar de otra/s modalidad/es. Existen tres tipos de enfoques de coaprendizaje: paralelo, no paralelo e híbrido (ver Figura 2.23). Los enfoques de datos en paralelo requieren que las observaciones de las distintas modalidades estén directamente vinculadas entre sí (por ejemplo, un conjunto de datos audiovisuales donde las muestras de vídeo y voz están sincronizadas). Los enfoques no paralelos no requieren esta vinculación

directa entre modalidades, sino que se basan en buscar conceptos o categorías compartidas (lo que permite una mayor comprensión semántica de estos). En el enfoque híbrido se dispone de dos modalidades que en principio no se encuentran en paralelo, pero que se pueden relacionar (o sincronizar) gracias a una tercera modalidad común. La propuesta que mejor refleja este enfoque es la NN Correlacional presentada por Rajendran *et al.* (2016) para aprender representaciones multimodales de datos no paralelos.

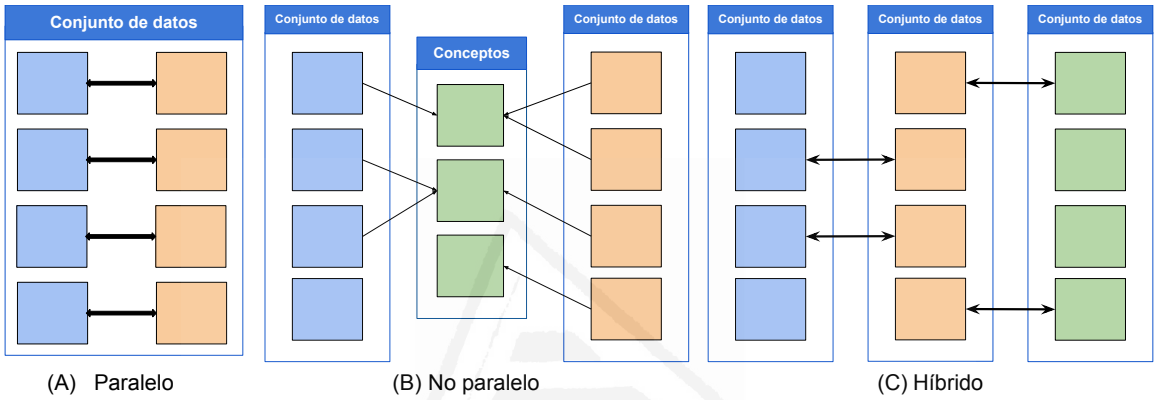


Figura 2.23.: *Tipos de paralelismo de datos utilizados en el coaprendizaje: (A) En paralelo, (B) No paralelo, (C) Híbrido. Imagen elaborada a partir de Baltrušaitis et al. (2019).*

Universitat d'Alacant
Universidad de Alicante

2.4.2. Aplicaciones de la multimodalidad

Como hemos visto, y como sustenta la literatura previa a nuestro estudio, la multimodalidad ha sido empleada en múltiples tareas que abarcaban todo tipo de aplicaciones y materias. La motivación principal del uso de esta técnica viene apoyada por el trabajo de Lahat *et al.* (2015). Desde una perspectiva teórica, se nos presenta un marco matemático que demuestra los beneficios del uso de varias fuentes de información provenientes de distintos tipos de datos, así como varios ejemplos prácticos que prueban la hipótesis.

En Baltrušaitis *et al.* (2019) se nos presenta una taxonomía común de los avances recientes en el aprendizaje automático multimodal, en la que concentra una recopilación de trabajos que demuestran las ventajas de su empleo y las

estrategias futuras para extraer el máximo partido a esta técnica. Encontramos una gran variedad de ejemplos de trabajos que la utilizan para mejorar los resultados obtenidos por las soluciones tradicionales, las cuales se limitaban a una única fuente de datos. Disciplinas tales como la creación de sistemas para la detección de gestos (Neverova *et al.*, 2014), el reconocimiento de emociones (Liu *et al.*, 2016), para responder preguntas (Fukui *et al.*, 2016) o para la reconstrucción de vídeo y audio (Ngiam *et al.*, 2011) mediante el aprendizaje de características compartidas entre estas modalidades (ver Figura 2.24). En una materia tan específica como la transcripción de notas musicales, también podemos encontrar ejemplos de los beneficios de la multimodalidad: en Krumhansl (2001) se nos presenta un método que combina distinta información de los instrumentos, como la tonalidad y el timbre de una determinada pieza, lo que permite identificar las notas que aparecen con más probabilidad; o Cambouropoulos (2010), que combina tres tipos de descripciones, como son los acordes, la instrumentación y el ritmo, para complementar su información y de esta forma restringir el espacio de búsqueda. La multimodalidad en el DL ha sido empleada incluso en los diagnósticos clínicos de pacientes para detectar enfermedades, como en (Xu *et al.*, 2016b), donde combina la información visual procesada con una red tipo CNN con otras fuentes de datos para mejorar la detección de la Displasia Cervical.

En el caso de la clasificación de imágenes, algunos métodos combinan la información visual con otras fuentes de datos para intentar mejorar de este modo los resultados obtenidos. En algunos casos, se incluyen datos como la posición GPS o información de los sensores de teléfonos móviles, como en Pertusa *et al.* (2018). Algunos enfoques anteriores combinaron con éxito descriptores visuales con información textual (Dinakaran y Kumar, 2010), y también con características como los metadatos de la cámara (Boutell y Luo, 2005). También se ha estudiado la multimodalidad en el DL para la creación de redes que puedan detectar las características más relevantes de las diferentes fuentes de datos. Un ejemplo es la red neuronal convolucional multimodal propuesta en Ma *et al.* (2015) para hacer coincidir imágenes y oraciones, o el aprendizaje de representaciones multimodal de imagen y texto mediante entrenamiento adversario realizado por Park e Im (2016).

En el campo de los estudios económicos y financieros, también podemos

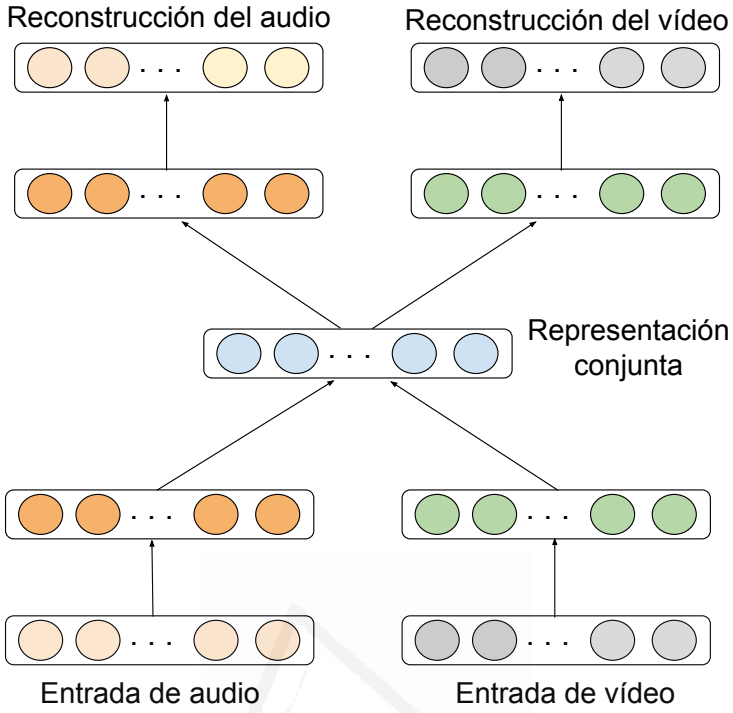


Figura 2.24.: *Esquema de un autoencoder multimodal (Bimodal Deep Autoencoder) para la reconstrucción de audio y vídeo propuesto por Ngiam et al. (2011).*

encontrar algunos trabajos (Lee y Yoo, 2020) que utilizan la multimodalidad para descubrir la relación entre características, comparando los diferentes enfoques empleados para llevar a cabo la fusión entre modalidades. En Zhao *et al.* (2018) se utiliza una representación multimodal del Mercado (precio-volumen, rendimiento histórico, indicadores técnicos, indicadores fundamentales, etc.) para perfilar las dinámicas temporales de su evolución y, de esta forma, poder efectuar predicciones futuras. En Cheng *et al.* (2019) se presenta una interesante propuesta multimodal para la predicción del éxito de un proyecto de *crowdfunding*, mostrando resultados prometedores. La combinación de diferentes fuentes de datos permite utilizar técnicas de NLP para predecir la evolución de los precios de las acciones y su volatilidad (Sawhney *et al.*, 2020; Sardelich y Manandhar, 2018). Por otro lado, también podemos encontrar trabajos (Do *et al.*, 2017) que proponen arquitecturas de redes neuronales (ver Figura 2.25) para combinar la información de las redes sociales, como es Twitter, con otras características y metadatos. Este trabajo también nos enseña la utilizad de emplear

la normalización ℓ_2 (Nie *et al.*, 2010) para evitar el sobreajuste (*overfitting*) y mejorar de este modo el resultado del proceso de entrenamiento.

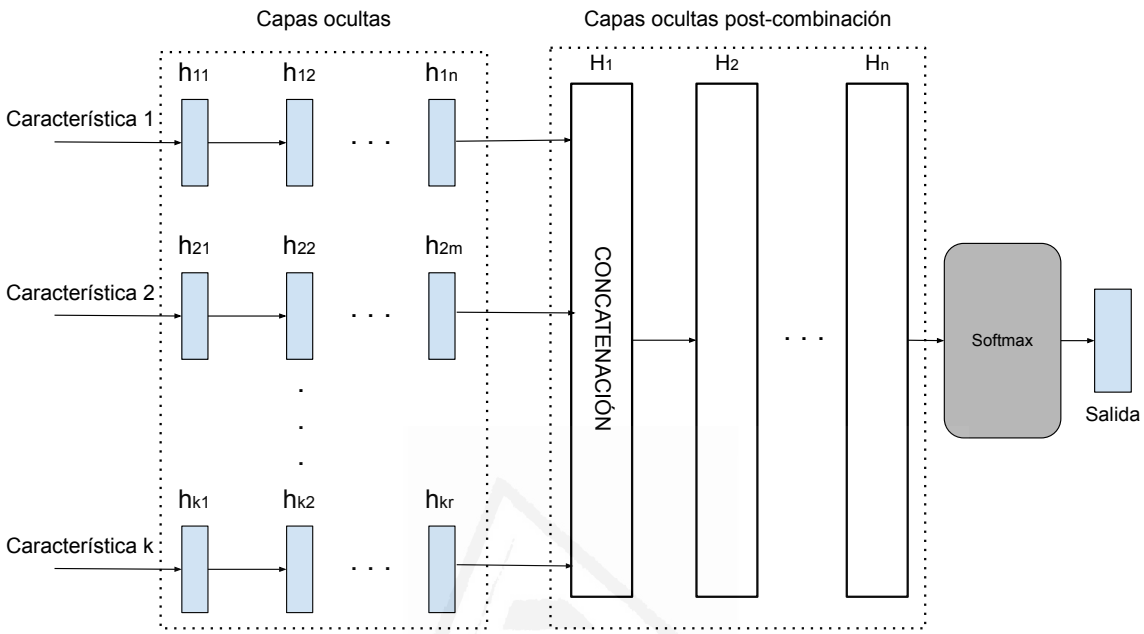


Figura 2.25.: *Arquitectura de la red multi-entrada denominada "MENET" propuesta por Do et al. (2017).*

Cada área de aplicación de la multimodalidad tiene asociado sus propios desafíos, que previamente hemos clasificado en el apartado 2.4.1. En la Tabla 2.1 se relaciona esta clasificación con las distintas aplicaciones de la multimodalidad que hemos revisado. La primera cuestión a destacar es que el desafío de la representación aparece en todos los tipos de aplicaciones, mientras que el resto dependen del tipo o la naturaleza de la información utilizada.

Como se ha podido apreciar en muchos de los trabajos revisados, los autoencoders constituyen una arquitectura de red idónea para su aplicación en tareas de multimodalidad. Como se explica en Wang *et al.* (2015) con su *Deep Canonically Correlated Autoencoder* (DCCAE), esta arquitectura permite comprimir en su espacio latente las características más representativas de las distintas modalidades y hallar en ellas las correlaciones entre las fuentes de información. Este elemento característico de los autoencoders es el que permite resolver el desafío de la *representación* al condensar la información de las modalidades en una representación coordinada (ver Figura 2.24). Esto resulta de especial

Tabla 2.1.: *Tabla resumen de los desafíos asociados a cada área de aplicación de la multimodalidad. Tabla adaptada del trabajo de Baltrušaitis et al. (2019).*

APLICACIONES	DESAFÍOS				
	Representación	Traducción	Alineación	Fusión	Coaprendizaje
Reconocimiento de voz					
Reconocimiento de voz audiovisual	✓		✓	✓	✓
Detección de eventos					
Clasificación de acciones	✓			✓	✓
Detección de eventos multimedia	✓			✓	✓
Emoción y afecto					
Reconocimiento	✓		✓	✓	✓
Síntesis	✓	✓			
Descripción de contenidos multimedia					
Descripción de imagen	✓	✓	✓		✓
Descripción de vídeo	✓	✓	✓	✓	✓
Respuesta visual a preguntas	✓		✓	✓	✓
Resumen	✓	✓		✓	
Recuperación de contenidos multimedia					
Recuperación multimodal	✓	✓	✓		✓
Hashing multimodal	✓				✓
Generación multimedia					
Síntesis (visual) de voz y sonido	✓	✓			
Generación de imágenes y escenas	✓	✓			

relevancia para el desafío de "fusión" cuando la combinación de la información se realiza de forma temprana (*early fusion*).

Dado los buenos resultados obtenidos por este tipo de técnica, se ha decidido incluirla en nuestra propuesta con el fin de combinar información de tipo visual, con información textual, así como con valores numéricos y con series temporales de datos. La arquitectura de red multimodal propuesta por Do *et al.* (2017) para la predicción del PIB, así como las distintas arquitecturas tipo autoencoder analizadas, han supuesto una base de referencia para la construcción de los modelos propuestos. Además, durante el desarrollo de esta tesis se considerarán tanto aproximaciones agnósticas, optando por la fusión de la información tipo *early* (con las ventajas que esto aporta, como veremos en los siguientes capítulos) como soluciones dependientes o integradas en el modelo.

Finalizada la revisión de los trabajos del estado del arte que han constituido las bases de nuestra investigación, pasamos a la presentación de los trabajos realizados que conforman el cuerpo de esta tesis.

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 3

Aplicaciones de la multimodalidad al reconocimiento de imágenes

En este capítulo se estudia un enfoque multimodal para realizar el reconocimiento de objetos a partir de fotografías capturadas mediante *smartphones*. El método propuesto procesa inicialmente la imagen de entrada utilizando una red tipo CNN, a partir de esta se extraen las características aprendidas (denominadas códigos neuronales, *Neural Codes* o NC) y se combinan con una serie de metadatos recopilados de los sensores del dispositivo. Estos metadatos complementan los contenidos visuales y pueden proporcionar información adicional para determinar la clase objetivo. Los metadatos también son procesados usando métodos de selección de características y codificando los campos textuales (como el tipo de lugar donde se tomó la fotografía) para mantener su significado semántico. Por último, para la combinación de las representaciones obtenidas a partir de las imágenes y los metadatos, se estudian diferentes aproximaciones multimodales para clasificar las muestras. Los resultados obtenidos muestran que el preprocesamiento de los metadatos es beneficioso, y que la combinación de NC y metadatos solo mejora los resultados cuando las imágenes se clasifican en categorías generales.

Universitat d'Alacant
Universidad de Alicante

3.1. Introducción

El reconocimiento de objetos es un campo de la visión por computador que tiene como objetivo identificar los elementos que aparecen en imágenes o vídeos. En los trabajos más recientes que podemos encontrar en el estado del arte (ver sección 2.2), este tipo de tarea se lleva a cabo principalmente utilizando redes tipo CNN (LeCun *et al.*, 2015). Entre la multitud de aplicaciones que tienen este tipo de arquitecturas, resulta especialmente interesante su integración en dispositivos móviles (Howard *et al.*, 2017). Un sistema de este tipo, para el reconocimiento y la clasificación de las imágenes que se estén visualizando, permite aprovechar la información obtenida para facilitar la interacción con el usuario, abriendo la puerta al desarrollo de multitud de aplicaciones y utilidades. Además, su integración en un dispositivo también permite la combinación de diferentes fuentes de información, como podrían ser los datos obtenidos por los sensores.

Una aplicación de este tipo es MirBot (Pertusa y Gallego, 2018), un sistema colaborativo que permite a los usuarios entrenar una IA para el reconocimiento de objetos. Para esto, el usuario tiene que capturar fotografías y seleccionar la región en la que se encuentra un objeto (también denominada región de interés, *Region Of Interest* o ROI). La imagen, las coordenadas del ROI y los metadatos asociados se envían a un servidor que realiza una búsqueda de similitudes y devuelve una predicción (utilizando para esto los *synsets* de WordNet (Fellbaum, 1998)) a partir de la imagen más similar encontrada en la base de conocimiento del sistema. Los usuarios de la aplicación pueden validar la respuesta indicando si es correcta o no, y, en caso de que no lo fuese, seleccionando la respuesta correcta. Gracias a esta retroalimentación, el sistema va aprendiendo y mejorando sus resultados. Además, esto también permite ir incrementando continuamente la base de conocimientos con imágenes etiquetadas y metadatos validados por los usuarios. El flujo de este funcionamiento se detalla en la Figura 3.1.

Los metadatos del conjunto de datos de Mirbot se extraen de los sensores del *smartphone* (como podría ser el ángulo con respecto a la horizontal, giroscopio, flash, GPS, etc.), además se calcula también información de geocodificación inversa (tipo de lugar, país, puntos de interés más cercanos, etc.) y todo

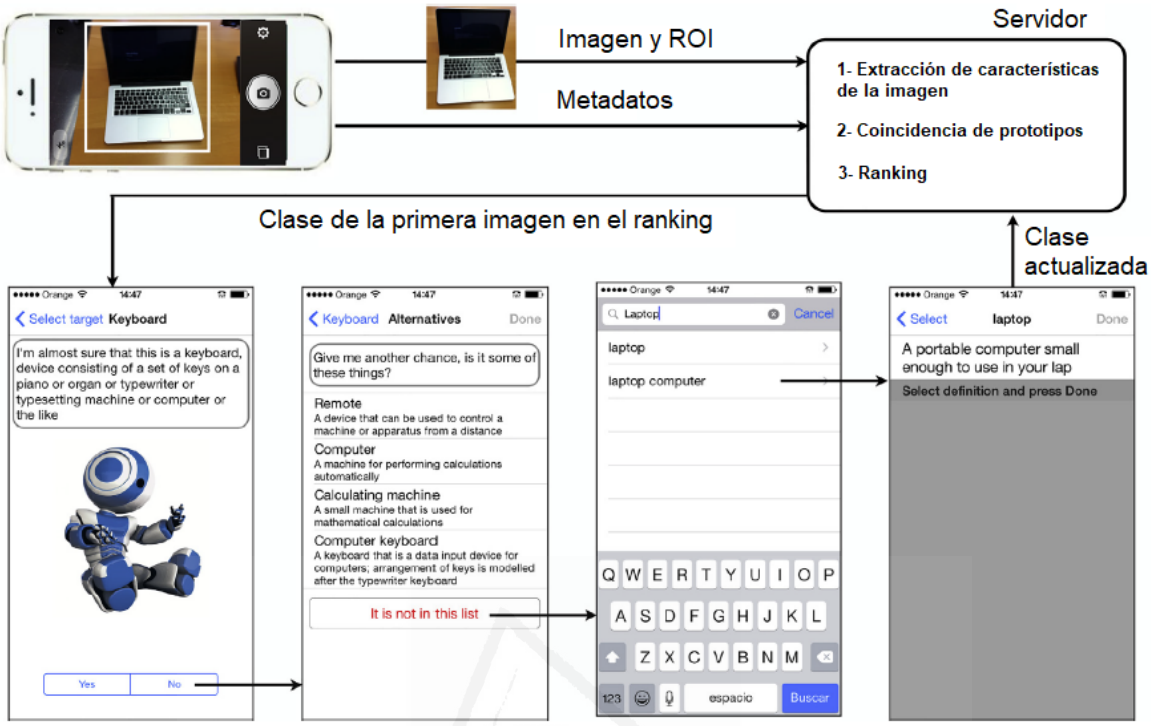


Figura 3.1.: *Arquitectura con la secuencia de procesamiento de la app Mirbot.*

esto se almacena junto a los datos EXIF (*Exchangeable Image File Format*) de la imagen capturada (apertura, brillo, ISO, etc.). Todos estos metadatos recopilados pueden ser de gran utilidad para reducir el espacio de búsqueda. Por ejemplo, si un usuario toma una fotografía de un elefante, es más probable que sea en un zoológico que en una playa, que el ángulo con respecto a la horizontal sea cercano a los 90 grados y que el flash esté apagado.

En este capítulo se realiza un estudio para incorporar el uso de la multimodalidad al método de clasificación utilizado en la aplicación Mirbot Pertusa *et al.* (2013, 2018), ya que actualmente, aunque recopile todos estos metadatos, se basa únicamente en la información visual para realizar las predicciones. Para esto se estudiarán distintas aproximaciones para realizar la fusión de la información, incluyendo algunas de las detalladas previamente en la sección 2.4.1, como son la fusión temprana, tardía o híbrida.

El resto de este capitulo está organizado de la siguiente manera. La sección 3.2 describe el conjunto de datos y la sección 3.3 la metodología utilizada para

la clasificación multimodal. Los resultados de la evaluación se detallan en la sección 3.4. Finalmente, la sección 3.5 aborda nuestras conclusiones y el trabajo futuro para seguir mejorando la propuesta.

3.2. Conjunto de datos

Para realizar este estudio, vamos a utilizar el conjunto de datos recopilado mediante la aplicación Mirbot. Como se ha indicado en la introducción, esta base de datos está continuamente creciendo gracias a las aportaciones de los usuarios. Por este motivo, con la intención de obtener resultados comparables a los publicados en Pertusa *et al.* (2018), nos basaremos en la versión del 23 de octubre de 2016. A esta fecha, 3.431 usuarios habían agregado 25.292 imágenes distribuidas en 1.808 clases. Algunos objetos aparecen con más frecuencia que otros y las clases están, por tanto, muy desbalanceadas. La mayoría de las imágenes se clasifican como objetos (18.685), seguidas de animales (4.928), alimentos y bebidas (1.113) y plantas (546). En la Figura 3.2 se pueden ver algunos ejemplos de imágenes de este dataset. Esta vista corresponde al panel de administración implementado en el servidor para validar las imágenes, ya que, además de la validación que realizan los usuarios, las nuevas imágenes se vuelven a validar para revisar que su etiquetado sea el correcto antes de añadirlas a la base de datos.

Para el etiquetado de las imágenes se utilizan los *synsets* definidos en la base de datos léxica WordNet (Fellbaum, 1998). Estos *synsets* son identificadores únicos de conceptos semánticos significativos. Cada uno de ellos está vinculado a una definición, aunque pueden estar relacionados con diferentes lemas. La principal ventaja que aporta este etiquetado es su estructura semántica, ya que gracias a la organización jerárquica se previenen las ambigüedades en el etiquetado. Por ejemplo, las dos clases del ejemplo anterior tendrían la siguiente

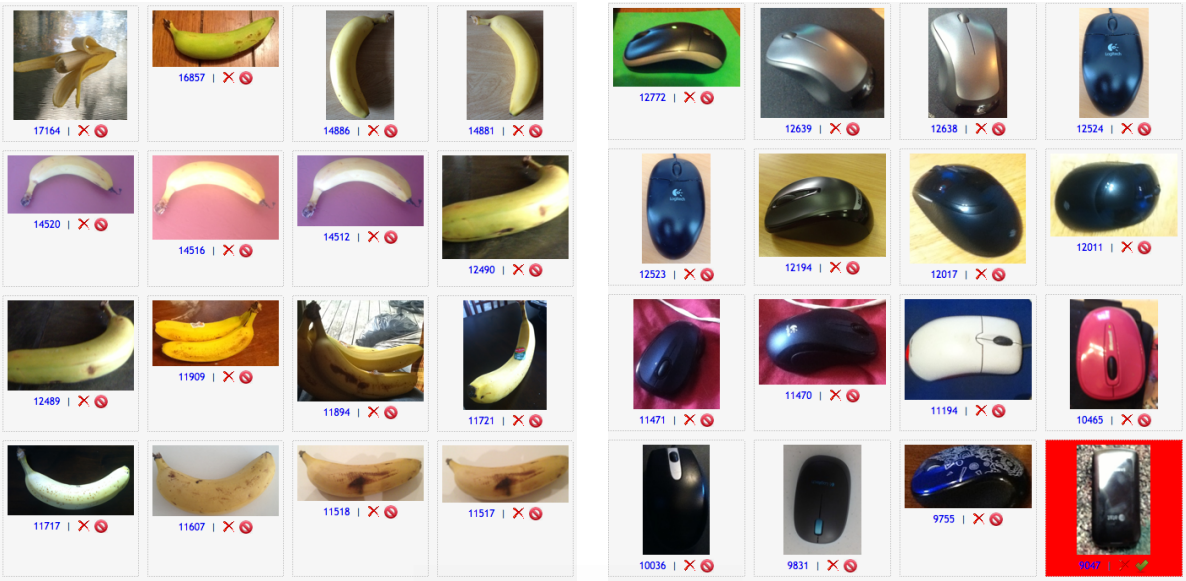


Figura 3.2.: *Captura del panel de administración utilizado para la validación de las imágenes clasificadas por la aplicación Mirbot. En la parte izquierda se muestran ejemplos de la clase "plátano" y en la derecha de la clase "ratón", donde uno de los ejemplos se ha marcado como incorrecto ya que la imagen está desenfocada y probablemente no pertenece a dicha clase.*

estructura jerárquica, empezando por la raíz hasta llegar a la hoja o clase utilizada para la clasificación:

- Objetos artificiales → instrumentalidad → dispositivo → dispositivo electrónico → ratón
- Comida y bebida → comida → fruta comestible → plátano

3.2.1. Metadatos

Como se ha indicado, en esta base de datos, además de las imágenes y de su etiquetado, se almacenan hasta un total de 80 metadatos extraídos del dispositivo móvil y de sus sensores en el momento de tomar la fotografía. Como se muestra en la Tabla 3.1, estos metadatos incluyen información del dispositivo (como el modelo, la versión del sistema operativo, etc.), datos de geolocalización

(latitud, longitud, altitud, localidad, sublocalidad, código postal, país, etc.), activación del flash de la cámara y los valores de los distintos sensores disponibles (acelerómetro, giroscopio, estado de la red, etc.).

Tabla 3.1.: *Metadatos obtenidos del smartphone.*

Metadatos	Valor de ejemplo	Descripción
osversion	6.0	Versión del sistema operativo
model	iPhone	Modelo de dispositivo
timestamp_picture	2012-10-27 10:10:05	Marca de tiempo
reliablelocation	YES	Ubicación adquirida con la precisión adecuada
lat	38,387	Latitud
lng	-0,511615	Longitud
altitude	99,7184	Altitud
name	Universidad de Alicante	Nombre principal de la función
locality	San Vicente del Raspeig	Localidad
sublocality		Sublocalidad
pc	03690	Código Postal
admin_area	Comunidad Valenciana	Nombre de la división administrativa de nivel 1
throroughfare		Nombre de la división administrativa de nivel 2
subthroroughfare		Nombre de la división administrativa de nivel 3
country	ES	Código del país (ISO-639-1)
numaoi	2	Número de áreas de interés cercanas
closestaoi	Eps I	Nombre del área de interés más cercana
horizontalerror	65	Error horizontal aproximado en metros
verticalerror	10	Error de altitud aproximado en metros
wifi	YES	Los datos se enviaron mediante una red WIFI
yaw	0,192139	Rotación (dirección)
pitch	0,677016	Rotación (inclinación)
roll	0,0703392	Rotación (cabeceo)
angle	38,7902	Ángulo con respecto a la horizontal en grados
accelerationx	-0,0283216	Aceleración (x)
accelerationy	0,0819144	Aceleración (y)
accelerationz	0,0515201	Aceleración (z)
globalacceleration	0,161756	Aceleración global
flash	OFF	Flash habilitado

Además, dada una latitud y una longitud, en el servidor se calcula la codificación geográfica inversa utilizando el sistema Gisgraphy (2018) y la base de datos geográfica `GeoNames`. Esto nos permite obtener otra serie de datos muy valiosos para determinar el tipo de localización, por ejemplo, permiten identificar si se trata de una universidad, un parque, un restaurante, un zoológico, etc.

Para obtener esta geocodificación inversa en base a las coordenadas devueltas por el dispositivo se realiza una búsqueda de la característica geográfica o punto de interés más cercano. En la Tabla 3.2 se muestra el listado de los datos recuperados a partir de `Gisgraphy`, incluyendo su definición y un ejemplo del tipo de valor que contiene. Algunos de estos datos, como `name` o `adm1`, pueden parecer redundantes con respecto a los datos de geolocalización devueltos por el *smartphone*. Sin embargo, hay que tener en cuenta que en ocasiones sus valores difieren o se complementan, ya que provienen de fuentes de datos distintas (`Gisgraphy`, `Apple` para la aplicación de `iOS` y `Google Maps` en el caso de Android).

Tabla 3.2.: *Metadatos obtenidos de la codificación geográfica inversa (`Gisgraphy`) en el lado del servidor.*

Metadatos	Valor de ejemplo	Descripción
feature_id	6255088	Identificación única para identificar la característica
name	Universidad de Alicante	Nombre de la característica
adm1	60	Código de división administrativa de nivel 1
adm2	A	Código de división administrativa de nivel 2
amd3	03014	Código de división administrativa de nivel 3
adm4		Código de división administrativa de nivel 4
adm1_name	Comunitat Valenciana	Nombre de la división administrativa de nivel 1
adm2_name	Provincia de Alicante	Nombre de la división administrativa de nivel 2
adm3_name	Alicante	Nombre de la división administrativa de nivel 3
adm4_name		Nombre de la división administrativa de nivel 4
feature_class	S	clase de entidad
feature_code	UNIV	código de característica
country_code	ES	ISO 3166 Código del país
population	0	Cuanta gente vive en ese lugar
elevation	110	Elevación en metros
gtopo30	91	Altitud media del área de 30'x30 'en metros
distance	282,141	Distancia a la entidad en metros

Los parámetros de la cámara y la configuración de captura de las imágenes también se almacenan junto a todos estos metadatos. La información del formato de archivo de imagen intercambiable (EXIF (2016)) que se envía al servidor incluye 23 parámetros adicionales, como son la distancia focal, el valor de apertura, el brillo, la velocidad ISO, el balance de blancos, etc.

3.3. Metodología

Para la combinación de los metadatos descritos en la sección anterior con la información visual extraída de las imágenes se consideraron distintas estrategias, como son la fusión temprana, tardía o híbrida, así como los enfoques independientes o integrados en el modelo. Como se explicó en la sección 2.4.1, cada una de estas aproximaciones tiene una serie de ventajas y desventajas que dependen del problema a abordar. Si bien en el apartado de experimentación compararemos los resultados obtenidos por cada uno de estos enfoques, cabe adelantar que, finalmente, el que mejor resultado obtuvo fue la fusión temprana mediante un enfoque agnóstico o independiente del modelo. Esto se debe principalmente a la naturaleza de los datos a procesar, una base de datos relativamente reducida y muy compleja, con multitud de clases diferentes y metadatos de muy distinta índole. Un enfoque independiente del modelo nos permite preprocesar y filtrar dichos datos para prepararlos, y además aprovechar modelos pre-entrenados especializados en la clasificación de las distintas fuentes o modalidades de información.

En la Figura 3.3 se puede ver el esquema multimodal propuesto, en el que inicialmente se pre-procesan las dos fuentes de información por separado y seguidamente se combinan usando un clasificador, siguiendo, como se ha indicado, una aproximación independiente del modelo de fusión temprana. En las siguientes secciones veremos en detalle cada uno de los pasos de este proceso.

Universitat d'Alacant
Universidad de Alicante

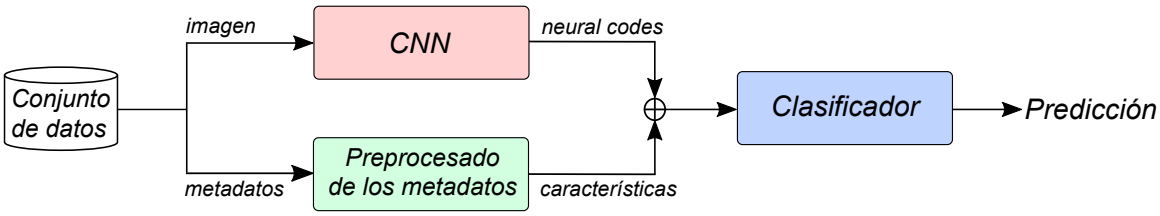


Figura 3.3.: *Arquitectura general del método propuesto.*

3.3.1. Preprocesado de las imágenes

Como se explicó en el capítulo 2, durante el proceso de entrenamiento de las redes CNN, se aprende a extraer una serie de características discriminativas de las imágenes procesadas. Inicialmente detectan características de más bajo nivel, como bordes, gradientes, etc., y, según aumenta la profundidad de la red, estas características se van agrupando para detectar otras de mayor nivel que sean representativas de la clase y además la diferencien de las otras clases (LeCun *et al.*, 2015). Es decir, las capas convolucionales actúan como un extractor de características que mapea la entrada a un nuevo espacio, donde las imágenes que pertenecen a la misma categoría se representan mediante vectores de características similares (Chatfield *et al.*, 2014). Esto permite que las últimas capas de la red después de las convoluciones (formadas por capas de perceptrones, también denominadas capas tipo *fully connected*) puedan clasificar fácilmente la imagen de entrada. Este nuevo espacio de características intermedio aprendido por la red, típicamente referido como espacio embebido, códigos neurales o Neural Codes (NC) (Babenko *et al.*, 2014), tiene muchas utilidades. Además de permitir la posterior clasificación de la imagen, también permite agrupar los datos, indexarlos, realizar búsquedas por similitud o incluso mejorar las tasas de clasificación al combinarlos con modelos de aprendizaje alternativos en un proceso conocido como *transfer learning* (Chatfield *et al.*, 2014; Gallego *et al.*, 2020).

En nuestro caso, vamos a extraer estos códigos neurales o NC a partir de las imágenes para posteriormente combinarlos con los metadatos recopilados y, de esta forma, intentar mejorar el resultado obtenido. Para la extracción de los NC se consideraron diferentes topologías de red del estado del arte, incluyendo AlexNet Krizhevsky *et al.* (2012), VGG-16/19 (Chatfield *et al.*, 2014), ResNet (He *et al.*, 2015) y Xception (Chollet, 2016), entre otras. En la publicación de Mirbot (Pertusa *et al.*, 2018) con la definición del método utilizado para la clasificación de las imágenes, se incluye una completa comparación entre estas arquitecturas, considerando diferentes modos de entrenamiento, como entrenar las redes desde cero o partiendo de pesos pre-entrenados, y diferentes aproximaciones para la clasificación, como cambiar las capas de salida o la clasificación de los NC mediante otros métodos especializados.

En este trabajo, con la intención de comparar los resultados al realizar la extensión multimodal del método, partiremos de la mejor configuración determinada en Pertusa *et al.* (2018), que fue el uso de la arquitectura ResNet (detallada en la sección 2.2.1.3) como extractor de características y calculando a partir de estas los k vecinos más cercanos entre la imagen a clasificar y la base de datos de conocimiento de Mirbot. Además, también se determinó que era mejor pre-entrenar la arquitectura ResNet y realizar un proceso de *fine-tune* (ajuste fino o entrenamiento para adaptar los pesos aprendidos) con las muestras de nuestro conjunto de datos. Por tanto, partiremos de la arquitectura ResNet inicializada con los pesos aprendidos a partir de ImageNet, una base de datos compuesta por millones de imágenes clasificadas en 1000 clases de tipos muy variados, como vehículos, animales, comida, etc, y posteriormente ajustada con las imágenes del dataset de Mirbot.

Al igual que en Pertusa *et al.* (2018), ResNet fue entrenada y ajustada a nuestro dataset utilizando imágenes a color redimensionadas a 224×224 píxeles. Posteriormente, las imágenes del dataset fueron procesadas por esta red para extraer los NC de la última capa oculta (vectores de dimensión 1.256). Estos vectores se normalizaron utilizando ℓ_2 y almacenaron para su uso en el último paso del proceso.

Universitat d'Alacant
Universidad de Alicante

3.3.2. Preprocesado de los metadatos

Los metadatos de MirBot incluyen valores numéricos, datos categóricos y cadenas de texto, por lo que para utilizarlos en la aproximación multimodal propuesta fue necesario realizar una etapa de preprocesado. Par esto se siguieron distintas estrategias en función del tipo de dato:

- Algunas características directamente se eliminaron por estar relacionadas con el usuario (como su identificador) o con el modelo del dispositivo (como *osversion* y *modelo*). De esta forma se evita que se creen relaciones entre los usuarios y sus datos, ya que algunos pueden tender a capturar más fotografías de animales, otros de objetos, etc.

- Los metadatos con valores numéricos (como tono, nitidez, distancia focal, etc.) se normalizaron en el intervalo $[0, 1]$.

- Los metadatos textuales se preprocesaron para, primero, unificar el idioma, y, segundo, transformarlos a vectores numéricos. En la subsección 3.3.2.1 se detalla este proceso.

Por último, una vez pre-procesados los metadatos y antes de suministrarlos al clasificador, se realizó un proceso de selección de características de los mismos. En este paso, que detallamos en la subsección 3.3.2.2, nos permitió filtrar los metadatos que estaban menos relacionados con el etiquetado.

3.3.2.1. Preprocesado del texto

El primer paso seguido para el preprocesado de los metadatos textuales fue unificar el idioma. Hay algunos campos (como la dirección) que pueden aparecer en muchos idiomas. Todas estas cadenas fueron traducidas al inglés de manera automática utilizando la API de *Google Translator*.

Para codificar el texto en vectores numéricos se consideraron distintas aproximaciones. Las más simples son el uso de un diccionario o la codificación *one-hot*, sin embargo, y como se explicó en la sección 2.2.4, ambas conllevan una serie de inconvenientes, como la pérdida del significado semántico de las oraciones. Para abordar este problema, en este trabajo proponemos codificar estos valores usando Doc2Vec (Le y Mikolov, 2014), que es una extensión de Word2Vec (Mikolov *et al.*, 2013). Como sugiere su nombre, Doc2Vec genera un vector numérico que representa los párrafos y oraciones utilizando para ello el contexto de las palabras dentro del texto. De esta forma se consigue preservar el significado semántico dentro del conjunto de la frase. Por lo tanto, Doc2Vec es utilizado en lugar de Word2Vec, ya que varios de los metadatos utilizados pueden contener frases.

En concreto, para nuestra experimentación usamos el modelo Doc2Vec (Lau y Baldwin, 2016) implementado en Gensim (Řehůřek y Sojka, 2010), una biblioteca de Python para el modelado de espacios vectoriales. Esta librería

incluye dos modelos previamente entrenados: *Wikipedia DBOW en inglés* y *Associated Press New DBOW*. Inicialmente evaluamos estos dos modelos con nuestros metadatos usando la configuración de parámetros predeterminada. De este experimento se concluyó que el modelo entrenado con la Wikipedia obtenía la mejor precisión, lo cual también resulta consistente con los resultados reportados en Lau y Baldwin (2016).

Dentro de los metadatos textuales almacenados, también podemos encontrar siglas y abreviaturas (como UK para Reino Unido en el campo con los datos del país o SCH para *School* en el código de característica). En estos casos, determinamos que era más eficaz procesarlos para utilizar el nombre completo representado por dichas siglas o abreviaturas. De este modo, para, por ejemplo, codificar los códigos de característica (tipo de lugar donde se tomó la fotografía), antes de suministrarlos como entrada al modelo Doc2Vec, creamos una oración concatenando el nombre y la descripción correspondiente dada en Geonames (2018). A continuación se muestran algunos ejemplos del preproceso realizado sobre estas características:

- La abreviatura SCH se transforma en "*School, building where instruction in one or more branches of knowledge takes place.*"
- UNIV se transforma en "*University, an institution for higher learning with teaching and research facilities constituting a graduate school and professional schools that award master's degrees and doctorates and an undergraduate division that awards bachelor's degrees.*"
- CH se transforma en "*Church, a building for public Christian worship.*"
- FISH se transforma en "*Fishing area a fishing ground, bank or area where fishermen go to catch fish.*"

De entre todos los metadatos, los campos textuales que finalmente se procesaron usando la metodología descrita para transformarlos en *embeddings* vectoriales fueron: *name, locality, sublocality, admin-area, thoroughfare, gis-name, gis-adm1-name, gis-adm2-name, gis-adm3-name, country, gis-feature-code y gis-feature-class*. También es importante destacar que, antes de procesarlos con

Doc2Vec, se realizó un preprocesado siguiendo los pasos propuestos por Kumar y Sebastian (2012), para transformar el texto a minúsculas y eliminar caracteres especiales, puntuación y las *stop words* (palabras sin significado, como artículos, pronombres, preposiciones, etc.).

Como se ha indicado, como resultado de la codificación con Doc2Vec se obtuvo un vector numérico o *embedding* representativo de cada característica que, gracias al uso de esta codificación, nos permite preservar el significado semántico y sintáctico de las palabras. De esta forma conseguimos dotar de significado también a estas características con la intención de que, al realizar las búsquedas por similitud, las características más similares (como podría ser "escuela" y "universidad", "casa" y "edificio", o "estación de autobús" y "estación de tren") presenten una menor distancia entre ellas.

3.3.2.2. Filtrado de metadatos

Una vez preprocesados los metadatos, se aplicó un proceso de selección de características para filtrar los campos menos relacionados con el etiquetado. Para ello se compararon hasta un total de ocho métodos de selección (Hall *et al.*, 2009): *Best First*, *Genetic search*, *Greedy Stepwise*, *Linear Forward Selection*, *Random Search*, *Scatter Search V1*, *Subset Size Forward Selection*, e *InfoGain*.

Tras evaluar el resultado obtenido con cada una de estas técnicas, se observó que en algunos casos no coincidían. Por lo que finalmente se decidió aplicar un esquema de votación para seleccionar los mejores metadatos (aquellos más votados) de una manera más robusta. Para esto se asignó un voto si el esquema indicaba que la características debía permanecer, este proceso se repitió con los ocho métodos de selección, se sumaron todos los votos y finalmente se estableció un umbral mínimo. En experimentos iniciales se determinó fijar el valor de este umbral a 3, ya que, como veremos en la sección 3.4.1, con 4 o más votos se reducía mucho el conjunto, y con menos de 3 se consideraban características poco robustas.

3.3.3. Paso de clasificación

A partir de los metadatos preprocesados y de las características extraídas de la imagen, se realiza una etapa de clasificación para obtener la predicción final. Para esto, todas estas características se combinan creando un vector que se utiliza como entrada de este clasificador. En este paso último se consideraron además los siguientes clasificadores:

1. k vecinos más cercanos (*k-Nearest Neighbors* o kNN) (Altman, 1992): Es un método de aprendizaje supervisado que estima la probabilidad de pertenencia a una clase para un elemento dado en base a la distancia del mismo con respecto al resto de muestras del conjunto de entrenamiento. De este modo, la clasificación se realiza en función de la clase más repetida entre los k vecinos más cercanos. En nuestra experimentación se evaluaron valores de k en el rango $k \in [1, 100]$.

2. Máquinas de vectores de soporte (*Support Vector Machines* o SVM) (Vapnik, 1998): Es un algoritmo de aprendizaje supervisado que genera un modelo de clasificación a partir de un hiperplano que intenta maximizar la distancia a las muestras más cercanas (vectores de soporte) de cada clase. Hace uso de funciones kernel para mapear los datos de entrada a un nuevo espacio que facilite su clasificación. Además, con el fin de permitir cierta flexibilidad en el aprendizaje del hiperplano, se establece un parámetro C que controla la compensación entre errores de entrenamiento y el margen establecido. En nuestro caso consideraremos el kernel de *función de base radial* (o gaussiano) y ajustaremos el parámetro C en el rango $C \in [1, 1000]$.

3. Bosques Aleatorios (*Random Forests* o RF) (Breiman, 2001): Este método de aprendizaje supervisado genera un clasificador conjunto (*ensemble*) a partir de la combinación de árboles de decisión entrenados de manera aleatoria con distintos subconjuntos de las muestras de entrenamiento. El resultado final se obtiene combinando las decisiones individuales de cada árbol, asignando como predicción la clase más votada por el conjunto de árboles. En nuestro caso consideraremos para la experimentación un número de árboles dentro del rango $[5, 1000]$.

Universitat d'Alacant
Universidad de Alicante

3.4. Resultados obtenidos

En esta sección vamos a evaluar la metodología propuesta, empezando por analizar los resultados obtenidos al clasificar de manera individual los metadatos y las imágenes, y seguidamente compararemos con la aproximación multimodal y con los resultados reportados en la versión de Mirbot publicada en (Pertusa *et al.*, 2018).

Como se ha indicado en la sección 3.2, esta evaluación se realizó utilizando la versión del 23 de octubre de 2016 del dataset Mirbot, para de esta forma poder comparar los resultados obtenidos con los publicados en Pertusa *et al.* (2018). Al igual que en este trabajo, se filtraron las clases que contenían un solo prototipo (quedando 24.794 imágenes distribuidas en 1.180 clases), se aplicó validación cruzada (*cross validation*) con 5 particiones y se utilizó *top-1* como métrica de evaluación, donde se considera un verdadero positivo cuando la clase del prototipo más cercano coincide con la clase de la imagen evaluada.

Además, en este trabajo evaluaremos los resultados considerando tres niveles de etiquetado: nivel raíz (con las 5 categorías principales: animales, comida y bebida, objetos hechos por el hombre, objetos naturales y plantas), el segundo nivel de la jerarquía de WordNet (con 92 clases) y el nivel de hoja (con las 1180 clases). De esta forma podremos analizar la capacidad de generalizar de la metodología propuesta y de las distintas fuentes de información utilizadas al clasificar las imágenes y los metadatos con distintos niveles de precisión, de más genérico a más específico.

Universitat d'Alacant
Universidad de Alicante

3.4.1. Evaluación usando metadatos

En primer lugar vamos a analizar los resultados obtenidos del proceso propuesto para el filtrado de metadatos. En la Tabla 3.3 se muestra el subconjunto de metadatos finalmente seleccionados junto al número de votos recibidos (de un total de 8 posibles votos). Como era de esperar, uno de los metadatos más representativos es el campo "*gis feature code*", correspondiente al código de característica (Geonames, 2018), que almacena el tipo de lugar, como podría ser

zoológico, centro comercial, universidad, playa, etc. Otro de los más relevantes es el "*pitch*", ya que indica la inclinación del dispositivo y nos puede dar ciertas pistas sobre el tamaño del objeto. Lo mismo sucede con "*focal length*", ya que nos permite determinar (a grandes rasgos) si se estaba fotografiando un objeto cercano o lejano. Por tanto, para el resto de experimentos, solo se considerarán estos 14 metadatos filtrados.

Tabla 3.3.: *Metadatos filtrados utilizando diferentes métodos de selección de atributos combinados mediante un esquema de votación. El número de votos de cada metadato se indica entre paréntesis (de un total de 8 posibles votos). Todos los campos corresponden a valores numéricos a excepción de los marcados con un asterisco (*), los cuales se preprocesaron usando Doc2Vec.*

Sensores	**Localización**	**EXIF**
pitch (7)	*gis feature code (7)	focal length (7)
selected area (4)	*gis feature class (6)	brightness value (6)
wifi (3)	reliable location (6)	color space (6)
flash (3)	*country (3)	sharpness (5)
	ocean (3)	subject area (5)

Seguidamente, vamos a evaluar el resultado obtenido al clasificar las muestras de la base de datos utilizando solamente los metadatos. Para esto, además de analizar el resultado en los tres niveles de etiquetado considerados, compararemos también los distintos clasificadores propuestos y la mejora obtenida al aplicar el preprocesado con Doc2Vec a las características textuales y sin aplicarlo (usando codificación *one-hot* en este caso).

La Tabla 3.4 muestra los mejores resultados obtenidos por cada clasificador. En primer lugar destacar que los resultados con preprocesado mejoran en todos los casos a los obtenidos sin preprocesado, alcanzando una mejora bastante significativa al utilizar RF y sobre todo a nivel de clase. Los resultados en los primeros niveles de la jerarquía de etiquetado (raíz y 2º nivel) son sorprendentemente buenos considerando que la clasificación se realiza sin ninguna información visual. Sin embargo, estos resultados empeoran significativamente al clasificar a nivel de clase. Una posible explicación es que los metadatos ofrecen información muy genérica de las muestras que únicamente permite discriminar entre los primeros niveles de etiquetado.

Sobre los tres clasificadores comparados destacar que según el nivel de etiquetado parece ser más conveniente aplicar uno u otro. En los primeros niveles de etiquetado los tres obtienen resultados similares, sin embargo, a nivel de clase sí que se observa una mayor diferencia al usar kNN. Con respecto a la parametrización, en esta tabla se muestran los resultados obtenidos con la mejor configuración de parámetros considerados para cada uno, que fue: kNN con $k = 1$, RF con entre 150 y 300 árboles y SVM con $C = 10$.

Tabla 3.4.: *Comparación de los mejores resultados (utilizando la métrica top-1) para cada clasificador utilizando los metadatos sin preprocesamiento y con preprocesamiento.*

	Sin preprocesado			**Con preprocesado**		
Método	**Raíz**	**2º nivel**	**Clase**	**Raíz**	**2º nivel**	**Clase**
kNN	73,67	51,73	7,31	75,52	52,08	**27,37**
RF	67,80	35,31	9,94	**76,96**	52,77	20,01
SVM	73,70	52,01	6,29	76,09	**54,88**	15,51

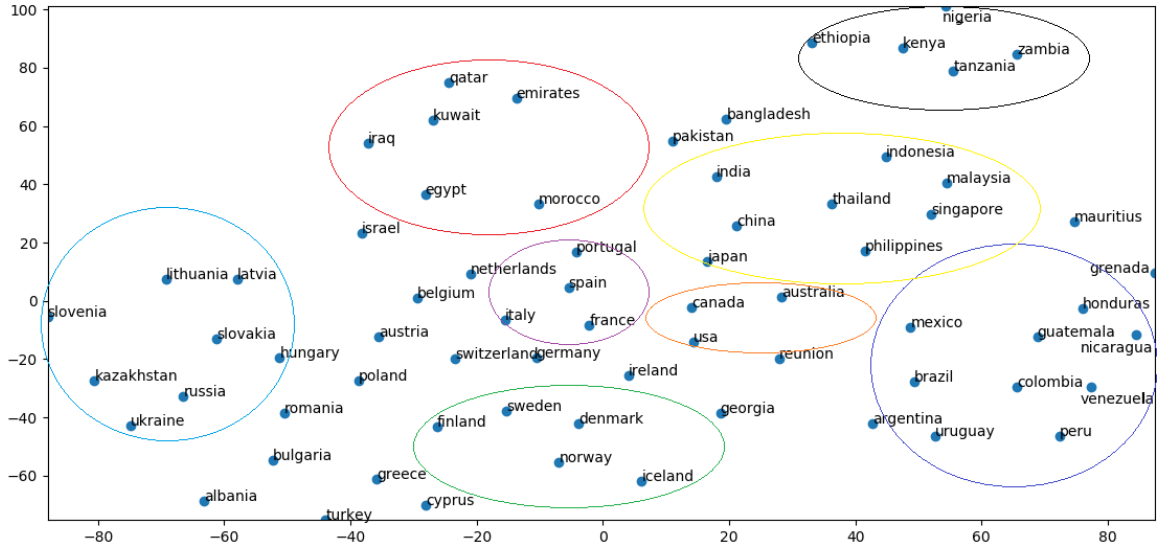
Por último, vamos a analizar el resultado obtenido al aplicar Doc2Vec para procesar los metadatos textuales. La Figura 3.4 muestra un par de ejemplos para las características de país y tipo de característica. Para representar estos embeddings en un espacio 2D se ha calculado la proyección utilizando el método t-Distributed Stochastic Neighbor Embedding (t-SNE (van der Maaten, 2008)). En esta figura se puede ver cómo se agrupan conceptos similares. Por ejemplo, la representación de países de América del Sur, África o Europa aparecen agrupados. Además, podemos comprobar cómo países dentro de un mismo continente se agrupan en función de su cercanía geográfica o cultural. Puede apreciarse cómo los países del este de Europa están más cerca entre ellos que con respecto al resto de países europeos, y lo mismo sucede con los países nórdicos. Países de lenguas romances como España, Portugal y Francia se encuentran próximos y, de igual forma, países con historias entrelazadas como Grecia, Chipre y Turquía. Para el campo con el tipo de característica puede apreciarse cómo elementos de la naturaleza, relacionados con las montañas o altas elevaciones, se agrupan muy cerca (montaña, colina, acantilado, etc.). Zonas propias de las ciudades, como factorías, estación de metro o calle también se agrupan. Áreas verdes o relacionadas con reservas naturales o parques, ya sean zoos, islas o granjas, también se encuentran muy próximas. Construcciones hechas por el

hombre, como casas, iglesias, museos, castillos o bibliotecas, son los puntos que más juntos aparecen (incluso dentro de este conjunto, las construcciones más relacionadas entre sí, como colegio o universidad, están más cerca entre ellos). Otros conjuntos interesantes que se forman son los de las zonas relacionadas con las bahías, puentes, canales, lagos o calas, o los de las palabras dedicadas al transporte, como puertos, aeródromos, aeropuertos, estación de tren o estación de bus.

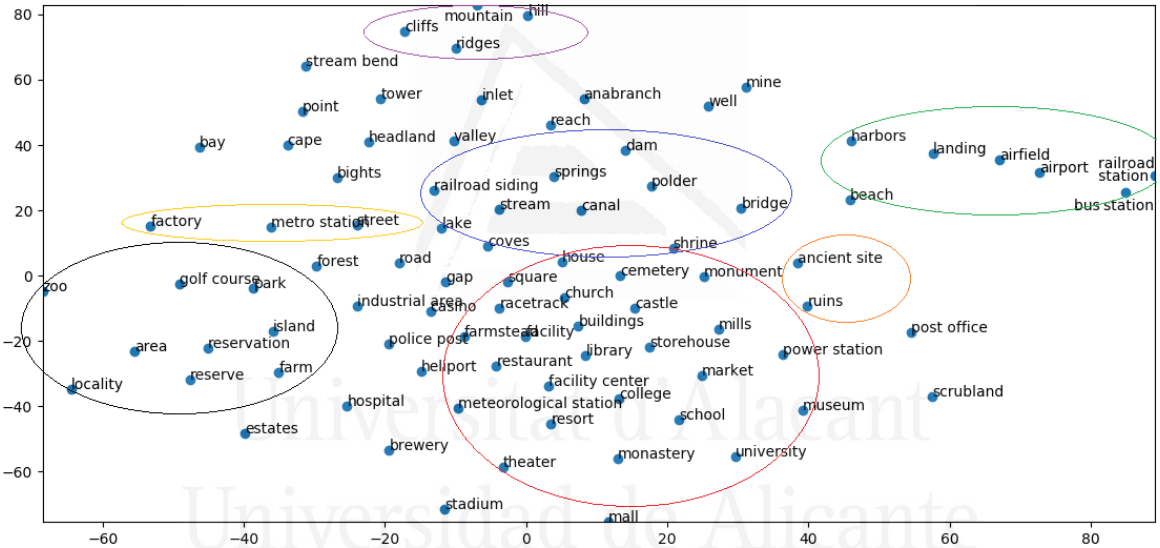
3.4.2. Evaluación usando características visuales

En esta sección vamos a evaluar la precisión (en términos de *top-1*) obtenida al clasificar el conjunto de datos utilizando solamente las características visuales, esto es, los NC extraídos mediante la arquitectura ResNet. La Tabla 3.5 muestra el resultado de clasificación a los tres niveles de etiquetado considerados. Como se indicó previamente, el mejor resultado obtenido en Pertusa *et al.* (2018) para este conjunto de datos fue utilizando la arquitectura ResNet para extraer los NC y después clasificarlos usando kNN. Con esta aproximación se consiguió un acierto del 77,7 % a nivel de clase. En el presente trabajo esta experimentación se ha extendido para considerar también RF y SVM y además evaluar la precisión de estos métodos a dos niveles más de etiquetado.

Como puede apreciarse en la tabla de resultados, RF obtiene los mejores resultados a nivel de clase y al 2º nivel de etiquetado. Si lo comparamos con los resultados reportados en Pertusa *et al.* (2018), este resultado mejora en casi un 1 % al utilizar RF. Sin embargo, para el etiquetado a nivel raíz, con muchas menos clases, se obtiene los mejores resultados mediante SVM. De forma similar a los metadatos, se obtuvieron los mejores resultados utilizando un valor de $k = 1$ para kNN, $C = 10$ para SVM y 150 árboles para RF.



(a) *País.*



(b) *Tipo/código de característica (feature code).*

Figura 3.4.: *Proyección de los word embeddings utilizando t-SNE (van der Maaten, 2008) para los códigos de país y los tipos de características.*

Universitat d'Alacant
Universidad de Alicante

Tabla 3.5.: *Resultados obtenidos (en términos de top-1) a partir de los NC por cada clasificador para los distintos niveles de etiquetado considerados.*

	Neural Codes		
Método	**Raíz**	**2º nivel**	**Clase**
kNN	93,72	83,60	77,70
RF	90,24	**84,24**	**78,68**
SVM	**94,81**	84,02	76,93

3.4.3. Evaluación combinando metadatos y características visuales

Como se ha podido ver en los resultados de las secciones anteriores, aunque la principal fuente de información la dan las características de la imagen, en esta sección vamos a evaluar si los metadatos combinados con estas características siguiendo la metodología propuesta ayudan en el proceso de clasificación.

Inicialmente se consideraron también otras aproximaciones multimodales, como enfoques en los que la fusión la realizaba el propio modelo o arquitectura de red. Sin embargo, estas soluciones obtenían resultados bastante peores (hasta un 20 % peor de media para los distintos niveles de etiquetado). Esto se debe principalmente a que como hemos visto, las características visuales extraídas por la red ResNet realizan la mayor contribución a este resultado. Al utilizar un enfoque integrado, no es posible partir de pesos pre-entrenados, sino que es necesario construir un nuevo tipo de arquitectura con múltiples entradas y entrenarlo desde cero. Por lo que, debido a la complejidad del dataset y a la relativa poca cantidad de muestras para el elevado número de clases, no se conseguía alcanzar un resultado satisfactorio.

En la Tabla 3.6 se muestran los resultados obtenidos al aplicar el algoritmo multimodal propuesto. Si comparamos estos resultados con los reportados en la sección anterior utilizando solo características visuales, se puede observar cómo solo se consigue una ligera mejora a nivel raíz (particularmente usando SVM), empeorando estos resultados según aumenta la especificidad del etiquetado. Estos resultados también coinciden con los reportados en la evaluación de los

metadatos. Por un lado, los metadatos, al aportar información genérica, solo ayudan ligeramente al clasificar en este nivel de detalle. Por otro, al obtener resultados tan pobres al clasificar a nivel de clase, confunden al método cuando se combina con la información de las imágenes. Por ejemplo, si una fotografía fue tomada en un zoo, la probabilidad de que el elemento que aparece en ella sea un animal (nivel raíz) será muy alta, pero a la hora de clasificar a nivel de clase, tendremos distintos animales con el mismo valor, lo que confundirá al clasificador.

Tabla 3.6.: *Resultados obtenidos mediante los distintos clasificadores y niveles de etiquetado considerados al combinar los NC y los medatados del dataset.*

	Datos multimodales		
Método	**Raíz**	**2º nivel**	**Clase**
kNN	94,36	**82,19**	**58,28**
RF	90,60	80,11	51,94
SVM	**94,98**	82,02	52,33

3.5. Conclusiones y trabajo futuro

Universitat d'Alacant
Universidad de Alicante

En este capítulo se propone una aproximación multimodal para la combinación de características visuales y metadatos para el reconocimiento de objetos. Para este proceso se ha propuesto una aproximación multimodal independiente del modelo que nos ha permitido preprocesar y filtrar dichos datos para prepararlos, y además aprovechar modelos pre-entrenados especializados en la clasificación de las distintas fuentes de información por separado.

El procesado de los metadatos textuales usando Doc2Vec ha demostrado obtener mejores resultados que otros tipo de codificaciones más sencillas, como *one-hot*, gracias a que ayuda a preservar el significado semántico de las palabras en la nueva representación vectorial generada. También destacar que los resultados obtenidos al clasificar en los primeros niveles de la jerarquía de etiquetado (raíz y 2º nivel) utilizando solo metadatos son sorprendentemente buenos considerando que la clasificación se realiza sin ninguna información

visual. Sin embargo, para niveles de detalle superiores, los resultados son mucho peores. Estos resultados muestran que la información contenida en los metadatos representa características genéricas de las clases de este dataset.

Al clasificar utilizando solamente características visuales, los resultados mejoran en gran medida en todos los niveles de etiquetado, sobre todo en los niveles con más detalle. Esto muestra que las imágenes resultan mucho más adecuadas que los metadatos para la clasificación en clases específicas. Al evaluar distintos clasificadores también se concluye que considerar otras soluciones para este paso del proceso, como SVM o RF, pueden reportar mejores resultados que los obtenidos en Pertusa *et al.* (2018) mediante kNN.

Al combinar los metadatos con las características visuales, solo se consiguió una ligera mejora en el nivel de etiquetado más genérico, empeorando al aplicarlo en los niveles más específicos. Estos resultados coinciden con los obtenidos al clasificar de manera individual los metadatos, indicando que esta información resulta muy genérica, por lo que confunde al clasificador cuando se intenta etiquetar clases específicas.

La multimodalidad nos ofrece una gran ventaja a la hora de trabajar con técnicas de ML gracias a su capacidad para combinar distintas fuentes de información. Sin embargo, es importante analizar la información representada en las distintas modalidades, bien para preprocesarla o filtrarla, o bien para determinar si resulta adecuada para el etiquetado a realizar, ya que, como se ha visto, en ocasiones su combinación puede ser perjudicial.

Como trabajo futuro, se pretende estudiar otras aproximaciones para la combinación de la información multimodal, así como estudiar la posibilidad de filtrar usando metadatos y después usar NC, o usar solo los metadatos cuando los NC tengan dudas entre dos clases. También planeamos evaluar redes neuronales multimodales para aprender relaciones más complejas entre los datos con el fin de mejorar la clasificación a nivel de clase.

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 4

Estimación del PIB regional mediante información textual

Partiendo de las conclusiones extraídas en el capítulo anterior y con el fin de acercarnos a uno de los principales objetivos de esta tesis, como es la aplicación de la multimodalidad en la economía, en este capítulo daremos un primer paso mediante la propuesta de una metodología para predecir el Producto Interno Bruto (PIB) regional utilizando la información textual almacenada en los mensajes escritos en la red social Twitter. En concreto, nos centraremos en la predicción del PIB de la Comunidad Valenciana utilizando los tweets escritos por los economistas, políticos, periódicos e instituciones más influyentes de la región. El método propuesto se basa en una arquitectura tipo autoencoder híbrida entrenada para minimizar simultáneamente el error de reconstrucción de la información textual de los tweets junto con su correspondiente valor de PIB. De esta forma, nos aseguramos que el espacio latente del autoencoder represente las características más importantes de los tweets que los relacionan con el PIB. Para la evaluación del método se consideraron diferentes tipos de topologías con las que construir la arquitectura de red y se estudiaron cuatro aproximaciones para calcular los *word embedding* con la codificación de los tweets. Los resultados obtenidos muestran que el enfoque propuesto consigue capturar la evolución del PIB con un pequeño margen de error. Este método también se comparó con otros trabajos del estado del arte, logrando en todos los casos una mejor precisión para la predicción del PIB.

Universitat d'Alacant
Universidad de Alicante

4.1. Introducción

Desde la aparición del ML, y más recientemente del DL, el estudio de problemas en campos donde las variables secuenciales son importantes, como en el Procesamiento del Lenguaje Natural o en la Economía, ha experimentado un avance significativo sobre los modelos estadísticos tradicionales, tanto en precisión como en tiempo de ejecución (Mullainathan y Spiess, 2017). Estas técnicas permiten aprender características adecuadas a partir de este tipo de información (LeCun *et al.*, 2015), lo que hace que su uso sea especialmente interesante como alternativa a los enfoques tradicionales utilizados en Economía.

Como se introdujo en el estudio del estado del arte realizado en el capítulo 2, varios trabajos (Llorente *et al.*, 2015; Bokányi *et al.*, 2017) han demostrado la conveniencia del uso de la información almacenada en las redes sociales como reflejo de la actividad económica utilizando técnicas de ML para procesar los datos. Twitter es una enorme base de datos que captura la actividad constante de los usuarios y, por tanto, puede utilizarse como una fuente adicional de información en estudios económicos. En este sentido, una aplicación interesante es la extracción de información relevante a partir de texto para aplicar técnicas de análisis de sentimientos (Joshi y Deshpande, 2018). Esta área del Procesamiento del Lenguaje Natural permite estimar la opinión de las personas y, por tanto, extraer tendencias, lo cual es de especial interés para complementar las soluciones basadas en la econometría (Pagolu *et al.*, 2016). Además, estos resultados compiten en calidad con los enfoques tradicionales y tienen la ventaja de poder calcularse de inmediato.

Como se detalló en la introducción (capítulo 1), las estimaciones oficiales del PIB se centran únicamente en la evolución anual del país en su totalidad y, solo en ocasiones, se consideran también unidades geográficas inferiores. En el caso concreto de España, el PIB está a disposición del público de forma anual y trimestral, pero el PIB regional solo se calcula anualmente y no se proporciona una cifra oficial para provincias o ciudades. En la literatura podemos encontrar algunos enfoques que pretenden llenar este vacío (Junoh, 2004; Richardson *et al.*, 2018; Indaco, 2020), como se revisó en la sección 2.3.2. Sin embargo, bien obtienen estimaciones no muy precisas de la variación del PIB o bien se basan

en alguna variable económica y, por lo tanto, solo se pueden calcular varios meses después del final del trimestre, cuando se publican los datos económicos necesarios.

En este capítulo, se propone un método para predecir el PIB regional utilizando solo la actividad de Twitter publicada por los economistas y políticos más influyentes, así como de periódicos e instituciones de una zona geográfica. El principal objetivo es estudiar en qué medida las interacciones de Twitter contienen información relevante sobre la actividad económica. En particular, nos centraremos en el cálculo del crecimiento del PIB regional de la Comunidad Valenciana utilizando para esto *solo* información clave de esta red social. Como veremos, los resultados obtenidos demuestran que el método propuesto permite capturar las fluctuaciones del PIB a nivel regional. Los resultados se comparan con otros enfoques del estado del arte, reportando una mayor precisión en todos los casos.

El resto de este capítulo está organizado de la siguiente manera. La sección 4.2 describe la metodología utilizada para predecir el PIB. La sección 4.3 presenta el conjunto de datos utilizado. La sección 4.4 detalla los resultados de la evaluación y, por último, la sección 4.5 aborda las principales conclusiones extraídas de este trabajo.

Universitat d'Alacant
Universidad de Alicante

4.2. Metodología

La Figura 4.1 muestra un esquema general del enfoque propuesto para la predicción del PIB. Primero, los tweets se preprocesan y codifican como *word embeddings*. A continuación, se utiliza un autoencoder para obtener una representación intermedia asociada tanto con los tweets como con el PIB. Este autoencoder utiliza una arquitectura híbrida en la que no solo se reconstruye el texto, sino también el PIB correspondiente. De esta forma, nos aseguramos que el espacio latente del autoencoder aprenda una representación de los tweets que se encuentre relacionada con el PIB. Por último, los NC extraídos del autoencoder se procesan mediante un regresor para predecir el PIB. En las siguientes secciones, estos pasos se explicarán en detalle.

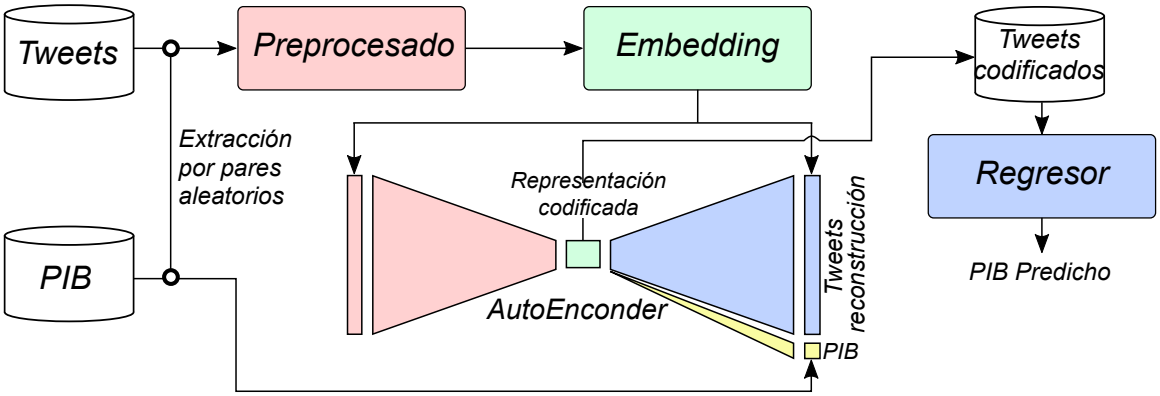


Figura 4.1.: *Arquitectura general del método propuesto.*

4.2.1. Preprocesado de los tweets

Los tweets tienen características únicas, como una longitud limitada (140 caracteres), palabras especiales (por ejemplo: "RT" se usa para indicar que es una copia del tweet de otra persona), abreviaturas, acrónimos, ortografía incorrecta, emoticonos, etiquetas o *hashtags* (el símbolo "#" se usa para marcar temas) o citas a otras cuentas (el símbolo "@" se usa para hacer referencia a la cuenta de otro usuario). Todo esto hace que sea necesario preprocesar los tweets antes de usarlos. Para esto se han seguido los pasos propuestos por Kumar y Sebastian (2012) (ver sección "Preprocesamiento de Tweets"), que nos permiten eliminar todas estas características únicas para quedarnos con una redacción correcta y sencilla de procesar. En resumen, los pasos seguidos en este preprocesado son los siguientes:

1. Eliminar enlaces, etiquetas hash y las palabras especiales de Twitter.
2. Corregir la ortografía.
3. Reemplazar los emoticonos con su polaridad de sentimiento.
4. Eliminar los caracteres especiales y la puntuación.
5. Eliminar las *stop words*.
6. Convertir a minúscula el texto.

A continuación se muestra un ejemplo del resultado obtenido tras aplicar cada uno de los pasos de este preproceso sobre un tweet original:

1. Tweet original: "*Las #exportaciones de la #Comunidadvalenciana a #Brasil, #India, #China y #Rusia crecen 42 % http://www.elperiodico[...] vía #bric*".
2. Eliminamos enlaces, etiquetas hash y palabras especiales de Twitter, obteniendo: "*Las exportaciones de la Comunidadvalenciana a Brasil, India, China y Rusia crecen 42 %*".
3. Corregimos la ortografía: "*Las exportaciones de la Comunidad valenciana a Brasil, India, China y Rusia crecen 42 %*".
4. Eliminamos emoticonos, caracteres especiales y la puntuación: "*Las exportaciones de la Comunidad valenciana a Brasil India China y Rusia crecen 42*".
5. Eliminamos las *stop words*: "*exportaciones Comunidad valenciana Brasil India China Rusia crecen 42*".
6. Y convertimos a minúscula el texto, teniendo como resultado: "*exportaciones comunidad valenciana brasil india china rusia crecen 42*".

4.2.2. Generación de word embedding

Una vez realizado el preprocesado, procedemos a transformar el texto obtenido en una representación vectorial utilizando técnicas de *word embedding*, en la que las palabras o frases del vocabulario se mapean en vectores de números reales. Como vimos en la sección 2.2.4 del estado del arte y comprobamos también en el capítulo anterior, las alternativas como el uso de diccionarios para la asignación de identificadores únicos o la codificación *one-hot* no son adecuadas dado que no preservan el significado semántico de las palabras. Dado que los tweets son cadenas de texto donde el significado de las palabras está condicionado por el contexto de la oración, resulta más conveniente el uso de aproximaciones como Word2Vec (Mikolov *et al.*, 2013), Doc2Vec (Le y Mikolov, 2014) o Sense2Vec (Trask *et al.*, 2015). Los resultados obtenidos en el capítulo 3 comparando la clasificación usando datos textuales demuestran que el enfoque que usa modelos de codificación neuronal mejora significativamente la codificación *one-hot*.

En este capítulo se propone ampliar el estudio anteriormente realizado sobre

este tipo de técnicas comparando cuatro aproximaciones recientes del estado del arte:

1. *Sense2Vec* (Trask *et al.*, 2015): Esta aproximación es una extensión de Word2Vec, la cual mejora los resultados obtenidos gracias a la incorporación de un mecanismo para la desambiguación del sentido de las palabras. De los distintos modelos pre-entrenados disponibles para el idioma español, se seleccionó el que obtuvo los mejores resultados para nuestra tarea, coincidiendo además con los resultados reportados en Trask *et al.* (2015). La arquitectura utilizada en nuestra implementación (spaCy, 2020) recibe una cadena de texto como entrada y la transforma en vectores de características de 50 dimensiones.

2. *Neural Network Language Models (NNLM)* (Abadi *et al.*, 2015): Este modelo genera los *embeddings* utilizando redes neuronales recurrentes basadas en modelos del lenguaje que aprenden de manera simultánea la representación de cada palabra junto con la función de probabilidad para secuencias de palabras (Bengio *et al.*, 2003). La implementación utilizada (TensorFlowHub, 2020) está disponible para el idioma español y transforma cada oración en un vector de características de 50 dimensiones.

3. *Language-Agnostic SEntence Representations (LASER)* (Chaudhary *et al.*, 2019): Este método mapea las oraciones de entrada en un espacio vectorial común con características genéricas que se utiliza como representación intermedia para varios idiomas. El método realiza un procesamiento secuencia a secuencia utilizando una arquitectura de red codificador-decodificador compuesta por capas tipo BiLSTM. La implementación concreta utilizada (LASER, 2020) devuelve un vector de 1.024 dimensiones con el que representar las frases de entrada.

4. *Sentence-BERT (SBERT)* (Reimers y Gurevych, 2020): Este método extiende un trabajo previo (Reimers y Gurevych, 2019) implementando mecanismos de auto-atención a través de Transformers bidireccionales combinados con redes siamesas para mejorar la representación vectorial obtenida. La implementación utilizada para este trabajo está disponible en 50 idiomas (UKPLab, 2020) y, al igual que en los otros casos, recibe

como entrada directamente la cadena de texto y devuelven un vector de características de 512 dimensiones como salida.

En todos los casos, como resultado de la codificación, cada oración se transforma en un vector de características n-dimensional (también denominado *word embedding*). En experimentos preliminares, la normalización de los valores de este vector usando ℓ_2 (Razavian *et al.*, 2014) condujo a una mejora estadísticamente significativa en términos de precisión, lo cual también coincide con resultados obtenidos en aproximaciones similares (Gallego *et al.*, 2018). Dado un vector x, la normalización ℓ_2 se define como:

$$\|x\|_2 = \sqrt{\sum_{k=1}^{n} |x_k|^2} \tag{4.1}$$

siendo n el tamaño del mismo.

4.2.3. Autoencoder híbrido

Los *embeddings* normalizados obtenidos en el paso anterior se suministran a un tipo especial de autoencoder, que a diferencia de la arquitectura tradicional (explicada en la sección 2.2.2), se ha extendido para que reciba una entrada (el *embedding*), pero devuelva dos salidas (la reconstrucción del *embedding* y el PIB correspondiente). Por tanto, este *autocoencoder híbrido* (ver Figura 4.1) está entrenado para minimizar simultáneamente el error de reconstrucción de la información textual de los tweets junto con el valor del PIB correspondiente. La intención de este proceso es transformar el texto de los tweets en una representación que refleje tanto el sentimiento del contenido (su positividad o negatividad) como su relación con las posibles subidas o bajadas del valor del PIB. Es decir, este método combina ambas reconstrucciones en un solo modelo siguiendo un enfoque de aprendizaje multitarea (Zhang y Yang, 2017). Esto significa que el modelo tiene que compartir el espacio latente interno para modelar múltiples propiedades, lo que permite obtener la representación deseada relacionada con el PIB.

Por lo tanto, la arquitectura de red propuesta recibe una entrada con los *embeddings* normalizados (de tamaño n), que representan los tweets, y devuelve dos salidas, una con la reconstrucción de los tweets (del mismo tamaño que la entrada) y una salida adicional para el PIB de dimensión 1. Para el entrenamiento de esta arquitectura se propone además el uso de una función de pérdida $\mathcal{L}_r$ que combine de manera ponderada el resultado de la reconstrucción del tweet y la predicción del PIB, como se describe en la siguiente ecuación:

$$\mathcal{L}_r = \lambda \sum_{i=1}^{m} (t_i - \hat{t}_i)^2 + (1 - \lambda) \sum_{i=1}^{m} (g_i - \hat{g}_i)^2 \tag{4.2}$$

donde $\hat{t}$ es la reconstrucción del tweet, $\hat{g}$ es la reconstrucción del PIB, t y g son los valores reales del tweet y del PIB, respectivamente, m es el número de muestras, y el parámetro λ nos permite ajustar el peso asignado a cada salida. El valor óptimo de este parámetro se estudiará durante la experimentación (consulte la sección 4.4.1), aunque adelantamos que finalmente se seleccionó un peso de $\lambda = 0,1$, es decir, asignando un 90 % del peso de la reconstrucción a la salida del PIB y el 10 % restante a la salida de los tweets.

La topología específica de una arquitectura de este tipo puede ser muy variada, ya que se pueden considerar distintas configuraciones de capas y de parámetros. Como se puede ver en la Tabla 4.1, en nuestra experimentación consideramos cuatro tipos de capas: Vanilla (una red compuesta por capas FC, ver sección 2.1.1), CNN-1D (Kiranyaz *et al.*, 2021), LSTM (Hochreiter y Schmidhuber, 1997) y BiLSTM (Huang *et al.*, 2015), utilizando en todos los casos dos capas para el *encoder* y dos capas para el *decoder*. En la experimentación inicial también se evaluaron otras profundidades de red, pero se determinó que una sola capa no era suficiente y que más de dos capas tampoco mejoraba los resultados.

En todas las capas se utilizó *ReLU* como función de activación, a excepción de las dos salidas, para las cuales se utilizó una función de activación lineal (Goodfellow *et al.*, 2016), ya que se trata de un modelo de regresión que tiene que devolver valores continuos (la reconstrucción de los tweets y del valor del PIB). En comparación con otras funciones de activación similares, como la

Sigmoidea, *ReLU* permite un entrenamiento más rápido y efectivo con grandes conjuntos de datos.

Para el entrenamiento de la red, se utilizó el algoritmo de "retropropagación" (*backpropagation*, explicado en la sección 2.1.1), usando descenso de gradiente estocástico (Bottou, 2010). En concreto, en nuestra experimentación se entrenó la arquitectura de red durante 300 épocas con un tamaño de *batch* de 32 muestras y considerando la función de optimización adaptativa Adam, propuesta por Kingma y Ba (2015).

Tabla 4.1.: *Topologías consideradas para el autoencoder híbrido. La última capa tiene dos salidas: $FC(n)$ (Fully Connected o capa de perceptrones totalmente conectada) para la reconstrucción del embedding n-dimensional de los tweets, y $FC(1)$ para la salida del PIB, la cual se conecta directamente al espacio latente de la red (esta conexión se representa con el símbolo $\rightarrow$).*

Modelo	**Capa 1**	**Capa 2**	**Capa 3**	**Capa 4**	**Salida**
Vanilla	FC(64)	FC(32)	FC(32) $\longrightarrow$	FC(64) $\longrightarrow$	FC(n) FC(1)
CNN-1D	Conv1D(64, 3)	Conv1D(32, 3)	Conv1D(32, 3) $\longrightarrow$	Conv1D(64, 3) $\longrightarrow$	FC(n) FC(1)
LSTM	LSTM(64)	LSTM(32)	LSTM(32) $\longrightarrow$	LSTM(64) $\longrightarrow$	FC(n) FC(1)
BiLSTM	BiLSTM(64)	BiLSTM(32)	BiLSTM(32) $\longrightarrow$	BiLSTM(64) $\longrightarrow$	FC(n) FC(1)

Universitat d'Alacant
Universidad de Alicante

4.2.4. Paso de predicción

En esta última etapa del proceso, se utiliza el autoencoder entrenado en el paso anterior para procesar el conjunto de datos y extraer tanto la representación codificada (obtenida de la capa intermedia de la red) como el PIB estimado (obtenido de la segunda salida del autoencoder). Este conjunto resultante de datos codificados se almacena y se utiliza a continuación para realizar la predicción final, para lo cual se consideraron las siguientes alternativas:

1. Devolver directamente la media de las predicciones calculadas a la salida del autoencoder para el trimestre considerado.

2. Aplicar un algoritmo de ML de regresión para predecir el PIB real ajustado mediante el conjunto de entrenamiento con los PIB estimados por el autoencoder y empleando la representación codificada (los neural codes, NC) de los tweets. Para este paso se consideraron además dos opciones:

   - ϵ-*Support Vector Regression* (ϵ-SVR) (Awad y Khanna, 2015): Esta es una extensión de SVM (Vapnik, 1998) para aplicarlo a problemas de regresión. Como ya explicamos en la sección 3.3.3, SVM mapea los datos de entrada en un hiperplano para maximizar el margen entre las clases a identificar, y así reducir el error de la función de pérdida durante el entrenamiento. En el caso de la regresión, se añade un margen de tolerancia ϵ para la selección de los vectores de soporte, para de esta forma permitir la aproximación a los valores continuos del conjunto de datos.

   - *Random Forest regressor* (RFr) (Liaw y Wiener, 2001): Esta es también una extensión del algoritmo RF (Breiman, 2001) explicado en la sección 3.3.3, en el que también se genera un ensemble de árboles de decisión entrenados de manera aleatoria con distintos subconjuntos de las muestras de entrenamiento. En este caso, en lugar de predecir una clase, cada árbol predice un valor continuo, por lo que finalmente se calcula el promedio del conjunto de predicciones.

Universitat d'Alacant
Universidad de Alicante

Con la intención de analizar la mejora obtenida por el método, estas aproximaciones se compararon además con dos soluciones base sencillas, una considerando simplemente que el PIB no varía de un trimestre a otro, y otra realizando una regresión lineal a partir de los valores del PIB reales de los cuatro trimestres anteriores.

Como se verá en la sección de experimentación, de las distintas aproximaciones consideradas, se determinó que la que mejor resultado obtenía es la basada en ϵ-SVR, por lo que es la que finalmente se seleccionó para incluir en nuestra propuesta.

Durante la etapa de inferencia, para predecir el PIB de un nuevo trimestre, se realizan los siguientes pasos: 1) los tweets se preprocesan, codifican y normalizan como se explica en la sección 4.2.1, 2) los *embeddings* resultantes se procesan mediante el autoencoder para obtener su representación intermedia y 3) se utiliza ϵ-SVR para predecir el PIB. Esto devolverá una predicción para cada tweet del nuevo conjunto, por lo que, para obtener la predicción final a partir de este resultado, calculamos el promedio.

4.3. Conjunto de datos

Para la evaluación del método propuesto vamos a utilizar un conjunto de datos con un total de 30.000 tweets obtenidos de cuentas relacionadas con el ámbito económico de la Comunidad Valenciana. Los tweets se descargaron para un periodo de 5 años, desde el 2012 hasta el 2016, recopilando 1.500 tweets por trimestre, lo que resulta en el total de los 30.000 tweets de la base de datos.

La Tabla 4.2 muestra algunos ejemplos de las cuentas elegidas junto con su tipo y descripción. Como se puede ver, nos centramos en las cuentas influyentes y/o relevantes en el ámbito económico, incluyendo economistas, cadenas de radio, televisiones autonómicas, líderes políticos, instituciones oficiales del gobierno regional, periódicos económicos y sucursales regionales de periódicos nacionales. Cabe señalar también que para los políticos, además de los líderes, también se incluyeron consejeros y portavoces en áreas de gobierno relacionadas con la economía (como el concejal de hacienda).

Además de seleccionar tweets solo de las cuentas indicadas anteriormente, también los filtramos utilizando una serie de palabras clave relacionadas con el PIB, como por ejemplo: *pib, crecimiento económico, previsiones de crecimiento, deuda pública, deuda familiar, creación de empleo, sector servicios, exportaciones, importaciones, inversión, ingresos por turismo, inflación, indicador de confianza*, entre otros. De esta manera, creamos un conjunto de datos con opiniones no solo de personas relacionadas con la economía sino de opiniones que estaban directamente relacionadas con el PIB.

Tabla 4.2.: *Algunos ejemplos de las cuentas de Twitter utilizadas en el conjunto de datos.*

Tipo	**Definición**	**Cuentas**
Periódicos	Periódicos locales o secciones regionales de periódicos nacionales.	@lasprovincias, @larazon_cv, @elmundocv, @elpais_valencia, @ABC_CValenciana
Expertos económicos	Newspapers and groups specialized in economics	@elEconomistaCV, @Economia_3, @InstitutoINECA
Periodistas & economistas	Cuentas personales de economistas, profesores o periodistas	@JMaudos, @DaValero, @Fpiera, @ndelatower
Partidos políticos	Cuentas de partidos políticos y cuentas personales de políticos	@PSOEValencia, @ppcv, @ximopuig, @RubenIBB
Radio & televisión	Cuentas regionales de radio y televisión	@Informacion_TV, @radiovalencia, @Levante_TV
Administración pública & sectores de gobierno	Cuentas del gobierno y la administración regionales	@GVAhisenda, @GVAivace, @GVAeconomia

Universitat d'Alacant
Universidad de Alicante

Dado que el objetivo es predecir el PIB de un trimestre utilizando las opiniones del propio trimestre (tenga en cuenta que el PIB regional solo se publica anualmente de manera oficial (Tituaña Castillo, 2016)), los tweets se etiquetaron con la diferencia entre el PIB del trimestre en el que se escribieron con respecto al PIB del trimestre anterior. De esta forma, el etiquetado refleja si el comentario se debe de asociar a un incremento o a una disminución del valor del PIB. Esta diferencia se calculó en porcentaje y se normalizó en el rango [-1, 1].

En todos los experimentos aplicamos una validación cruzada con 5 particiones, usando un año para la evaluación y el resto para el entrenamiento. Por lo tanto,

el autoencoder fue entrenado y evaluado 5 veces usando estos conjuntos, y finalmente se calculó el resultado promedio.

4.4. Experimentos

En esta sección se incluyen los resultados de la experimentación realizada. Esta evaluación la hemos dividido en tres partes: empezaremos analizando la configuración del autoencoder y los resultados obtenidos con las diferentes topologías propuestas. Seguidamente evaluaremos el segundo paso para la predicción del PIB comparando las técnicas detalladas en la sección 4.2.4. Finalmente, estos resultados se compararán con otros métodos del estado del arte.

4.4.1. Resultados del autoencoder

El primer parámetro a evaluar es el valor de λ (ver Ecuación 4.2), ya que de esta configuración dependerán el resto de resultados. Para esto partiremos de una configuración básica usando la topología de red tipo Vanilla (ver sección 4.2.3) e iremos variando el peso λ de 0 a 1 con incrementos de 0,1. Además, como se ha comentado previamente, en todos los experimentos realizamos una validación cruzada con 5 particiones, creando en este caso una partición adicional de validación con la intención de no ajustar este valor a los conjuntos de test. La Figura 4.2 muestra el resultado promedio de este experimento para cada valor de λ (en términos de error cuadrático medio, *Mean Square Error* o MSE) y para cada una de las salidas del autoencoder, esto es, para el error de reconstrucción del valor del PIB y para la representación codificada de los tweets. Como se puede apreciar, el mejor resultado promedio se obtiene con $\lambda = 0,1$, lo que implica asignar un mayor peso al valor del PIB. Esto puede deberse al hecho de que la salida del tweet tiene una dimensionalidad mucho mayor, además, de esta forma la representación obtenida también estará más estrechamente relacionada con el valor del PIB. Por lo tanto, para el resto de experimentos fijaremos este parámetro a $\lambda = 0,1$.

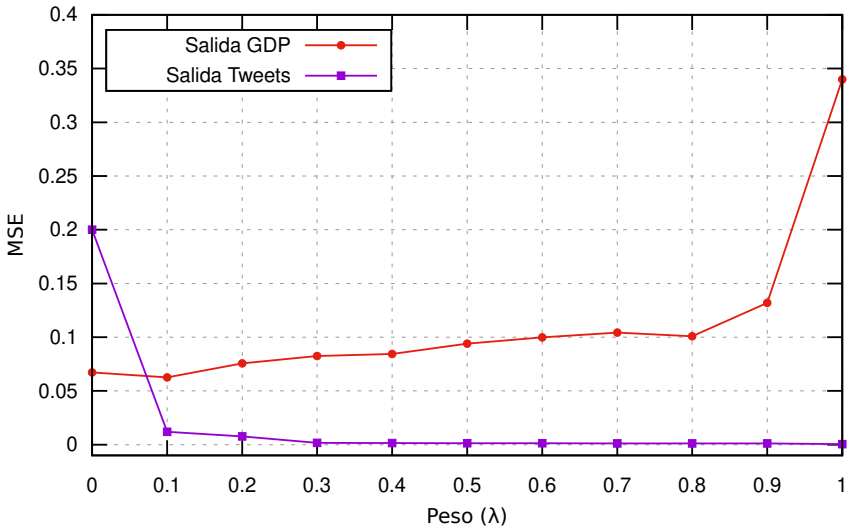


Figura 4.2.: *Resultados (MSE) obtenidos al variar el parámetro λ de la función de pérdida (ver Ecuación 4.2) para ponderar la contribución de las salidas de la red. $\lambda = 1$ significa que solo se usa la salida de los tweets, mientras que $\lambda = 0$ significa que solo se considera la salida del PIB.*

El siguiente aspecto a evaluar es la topología del autoencoder utilizado para calcular la codificación de los tweets. Las cuatro topologías de red descritas en la Tabla 4.1 y los cuatro métodos de *embedding* (Sense2Vec, SBERT, LASER y NNLM) descritos en la sección 4.2.2 para codificar los tweets son, por tanto, comparados utilizando para esto el mejor valor λ determinado en el experimento anterior. La Tabla 4.3 muestra los resultados de este experimento. Además incluimos en esta comparación una red neuronal con una sola salida para realizar la predicción del PIB (denominada "Fully Connected" en la tabla) y usando la misma topología que la red Vanilla descrita en la Tabla 4.1. De esta forma podemos comparar si una topología de red sencilla, centrada en la predicción del PIB, consigue mejorar los resultados obtenidos con la arquitectura híbrida propuesta.

En primer lugar, destacar que el resultado obtenido por la red "Fully Connected" con una sola salida no consigue mejorar a ninguna de las cuatro topologías consideradas para el autoencoder. En lo referente a estas últimas, casi todas obtienen un error de reconstrucción para los tweets muy bajo, que aumenta ligeramente en el caso de la predicción del PIB. El mejor resultado medio se

Tabla 4.3.: *Comparación del error de reconstrucción obtenido (en términos de MSE) con las diferentes topologías de autoencoder (AE) consideradas (ver Tabla 4.1). En negrita se muestran los mejores resultados obtenidos por topología, por método de embedding y en promedio.*

		Métodos de embedding				
Topología AE	**Salida**	**Sense2Vec**	**SBERT**	**LASER**	**NNLM**	**Media**
Fully Connected	PIB	0,1294	0,1211	0,1294	0,1286	0,1271
Vanilla	Tweet	0,0019	0,0012	0,0015	0,0011	0,0014
	PIB	0,1249	0,1187	0,1291	0,1219	0,1237
CNN-1D	Tweet	0,0016	0,0007	0,0009	0,0006	0,0010
	PIB	0,1223	0,1180	0,1193	0,1214	0,1203
LSTM	Tweet	0,0005	0,0001	0,0004	0,0001	0.0003
	PIB	0,1220	0,1178	0,1189	0,1212	0,1200
BiLSTM	Tweet	**0,0001**	**0,0001**	**0,0001**	**0,0001**	**0,0001**
	PIB	0,1217	**0,1177**	0,1187	0,1210	**0,1198**
Media	Tweet	0,0010	**0,0005**	0,0007	**0,0005**	–
	PIB	0,1241	**0,1187**	0,1231	0,1228	–

obtiene utilizando la red BiLSTM, tanto para la reconstrucción de los tweets como para la predicción del PIB. Recientemente, estas arquitecturas han demostrado ser más adecuadas para este tipo de datos (Huang *et al.*, 2015; Plank *et al.*, 2016), lo que se confirma en este experimento. También se observa que el método de *embedding* que obtiene el mejor resultado en promedio es SBERT. Por lo tanto, para realizar los siguientes experimentos utilizaremos BiLSTM y SBERT, además del parámetro $\lambda = 0,1$ previamente establecido.

4.4.2. Predicción del PIB

Una vez entrenada la red, procedemos a evaluar la siguiente parte del método que calcula la predicción del PIB utilizando un regresor (ϵ-SVR) a partir de los resultados obtenidos del AE.

Para evaluar este resultado utilizamos la métrica de Error Absoluto Medio

(*Mean Absolute Error* o MAE), ya que es una métrica más intuitiva para analizar el resultado real. Hay que tener en cuenta que en este apartado evaluamos la predicción final del PIB, mientras que en el apartado anterior medíamos el error cometido a la salida del autoencoder para la reconstrucción de los tweets. La Tabla 4.4 muestra el resultado obtenido con el método propuesto (AE + ϵ-SVR) para cada uno de los años evaluados. En esta tabla también se compara el resultado obtenido con otros cuatro métodos: *baseline*, regresión lineal, usando la salida del autoencoder y comparando con el uso de un regresor diferente. A continuación, se ofrece una mejor descripción de estos métodos:

1. *Baseline*: Supone que el PIB no varía de un trimestre a otro. Es decir, este método predice que el PIB del trimestre actual será igual al PIB real del trimestre anterior.

2. *Regresión Lineal*: La predicción se calcula realizando una regresión lineal a partir del valor del PIB de los cuatro trimestres anteriores.

3. *AutoEncoder (AE)*: Primero se predice el PIB para todos los tweets del trimestre evaluado usando el autoencoder y luego se calcula el promedio de estas predicciones.

4. *AE + RFr*: Este enfoque realiza los mismos pasos que el método propuesto (AE + ϵ-SVR), pero cambiando el regresor por un RFr (Breiman, 2001).

Universitat d'Alacant
Universidad de Alicante

Como se puede ver en los resultados de la Tabla 4.4, el método propuesto (AE + ϵ-SVR) obtiene el mejor resultado en promedio y de manera individual para todos los años considerados, siendo el método *AE + RFr* el que obtiene el segundo mejor resultado. Se puede observar cómo el método propuesto mejora notablemente enfoques más simples, como el *baseline* o la regresión lineal, reduciendo su error medio en 0,92 y 0,40, respectivamente.

La Figura 4.3 muestra un gráfico con la predicción del PIB para cada trimestre (línea roja). Este resultado se compara con el valor real del PIB (línea azul). Se puede ver cómo el método propuesto sigue la correctamente la evolución del PIB a lo largo del tiempo. En particular, es capaz de detectar los altibajos estacionales del PIB regional y su tendencia general utilizando solo tweets

Tabla 4.4.: *Comparación del resultado obtenido (MAE) para la predicción del PIB utilizando el método propuesto y otros cuatro enfoques.*

Método	**2012**	**2013**	**2014**	**2015**	**2016**	**Media**
Baseline	1,25	1,06	1,30	1,14	1,22	1,19
Regresión Lineal	0,51	0,78	0,94	0,55	0,58	0,67
AutoEncoder (AE)	0,41	0,37	0,48	0,38	0,42	0,41
AE + RFr	0,30	0,26	0,32	0,27	0,30	0,29
Nuestra propuesta (AE + ϵ-SVR)	**0,26**	**0,25**	**0,31**	**0,26**	**0,28**	**0,27**

escritos en ese trimestre. Este hecho se indica en la figura mediante la línea de puntos verde, que resalta las diferencias entre los valores estacionales (solo para las estaciones ascendentes). Como se puede observar, además de capturar la tendencia estacional, también detecta las diferencias entre estas subidas y bajadas, es decir, el componente irregular de esta tendencia que indica si el trimestre va a aumentar más o menos respecto al trimestre de tendencia ascendente anterior.

4.4.3. Comparación

Universitat d'Alacant
Universidad de Alicante

En esta sección vamos a comparar el método propuesto con los siguientes tres métodos del estado del arte, los cuales también se revisaron en la sección 2.3.2:

- Indaco (2020) estima el PIB de diferentes países del mundo haciendo una correlación entre el volumen de imágenes geolocalizadas compartidas en Twitter y una serie de indicadores, como densidad de población o población con acceso a Internet. Utiliza el número de tweets de imágenes tomadas de ese país en cada uno de esos años para estimar el PIB e incluye un factor que representa las diferencias en el uso de Twitter de un año a otro, así como los cambios en la economía global.

- Richardson *et al.* (2018) evalúa múltiples enfoques para predecir el PIB mediante el uso de diferentes indicadores económicos y comparando además

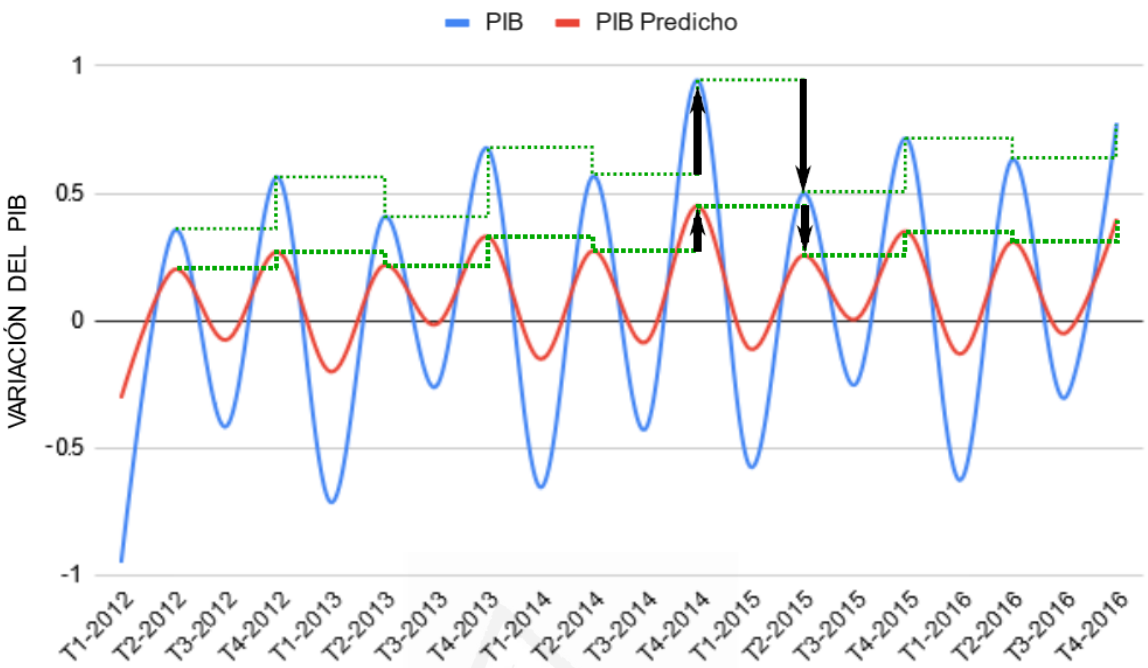


Figura 4.3.: *Comparación entre el valor real del PIB (línea azul) y la predicción del PIB (línea roja) realizada por el método propuesto para todos los trimestres del periodo considerado. Además se ha marcado mediante líneas punteadas en color verde las diferencias entre los valores estacionales (solo para los ciclos ascendentes).*

varios modelos de ML de regresión, como AR, SVR o NN. Para obtener la predicción final se combinan las predicciones individuales de estos modelos considerando para esto varias estrategias de ponderación, siendo la ponderación de mínimos cuadrados la que genera los mejores resultados.

- Junoh (2004) utiliza una red neuronal tipo Fully Connected entrenada para predecir el PIB utilizando un conjunto de cuatro indicadores económicos: usuarios de teléfonos móviles, usuarios de Internet, número de ordenadores y ordenadores personales por cada mil habitantes. La red consta de 4 entradas (correspondientes a los indicadores elegidos), una capa oculta y una salida para la predicción del PIB.

La Tabla 4.5 muestra el resultado de esta comparación utilizando la raíz del error cuadrático medio (*Root Mean Square Error* o RMSE), ya que era una de las métricas comunes implementadas en estos tres métodos. Además, es

importante destacar que esta comparación se tuvo que realizar para la predicción del PIB anual, ya que la mayoría de los métodos comparados (y las fuentes de datos que utilizan) solo están disponibles para esta periodicidad. Como puede observarse de esta comparación, el método propuesto mejora los resultados obtenidos por estos otros trabajos.

Tabla 4.5.: *Comparación del método propuesto con otros tres métodos del estado del arte en términos de RMSE.*

Método	**RMSE**
Indaco (2020)	0,53
Richardson *et al.* (2018)	0,42
Junoh (2004)	0,57
Nuestra propuesta	**0,36**

4.5. Conclusiones y trabajo futuro

En este capítulo se propone un método para la predicción del PIB regional a partir de la actividad económica recogida en la red social Twitter. En concreto, se estudia el caso de la Comunidad Valenciana utilizando cuentas de economistas, políticos, periódicos y de las instituciones más influyentes de esa zona geográfica. El método propuesto consta de una serie de pasos: inicialmente se preprocesan los tweets y se codifican en *word embeddings*. Seguidamente se utiliza una arquitectura tipo autoencoder híbrida con dos salidas, para de esta forma forzar al aprendizaje de una representación intermedia que esté relacionada tanto con la opinión que contiene el tweet como con el valor del PIB correspondiente al trimestre en el que se escribió. En un último paso, se utiliza el regresor ϵ-SVR entrenado con las representaciones generadas por la red y los valores del PIB reales para realizar la predicción final.

Durante la experimentación se compararon diferentes opciones para cada uno de los pasos del proceso, evaluando cuatro métodos de *word embedding* para la codificación de los tweets, varias topologías de red para el autoencoder y diferentes configuraciones para los hiperparámetros. Los resultados obtenidos

muestran que el método propuesto consigue capturar las fluctuaciones del PIB a nivel regional. Además, el método se comparó con otros enfoques, incluyendo tres métodos del estado del arte, reportando un menor error de estimación en todos los casos.

El método propuesto consigue por tanto predecir el PIB regional basándose para esto en las opiniones vertidas en Twitter por usuarios y medios relevantes relacionados con el ámbito económico. Estos resultados demuestran que las redes sociales proporcionan casi en tiempo real un reflejo bastante cercano de las perspectivas económicas, permitiendo además analizar estas opiniones para unidades geográficas donde las estadísticas oficiales no están disponibles.

Como continuación de este trabajo se pretende mejorar la solución propuesta incorporando diferentes fuentes de información usando para esto una aproximación multimodal. También se pretende analizar mejor el resultado obtenido en cada etapa del proceso, estudiando la representación generada y considerando posibles estrategias para filtrar opiniones aisladas.

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 5

Aproximación multimodal para la predicción del PIB regional

Tras los resultados obtenidos en el capítulo anterior a partir de información textual, en este capítulo se pretende extender esta propuesta para considerar varias fuentes de información siguiendo un enfoque multimodal. Para esto se plantea una nueva arquitectura dividida en dos etapas de procesamiento. Inicialmente se realiza un proceso similar al del método anterior para obtener una representación de los tweets relacionada con el PIB, añadiendo un paso de filtrado para eliminar los valores atípicos y calcular la predicción usando solo las opiniones que lleguen a un consenso. En una segunda etapa, este resultado se combina con la secuencia de valores históricos del PIB utilizando una red recurrente multimodal. El método se evalúa en cuatro regiones diferentes de España utilizando tweets escritos por los economistas, políticos, periódicos e instituciones más relevantes de cada una. Los resultados muestran como este nuevo enfoque consigue mejorar la aproximación previa así como a otros métodos del estado del arte. Este método también permite determinar las opiniones más y menos influyentes a la hora de realizar una predicción. Como ejercicio adicional se evalúa qué tan bien este método hubiera predicho el efecto de la pandemia de COVID-19 sobre la evolución del valor del PIB.

Universitat d'Alacant
Universidad de Alicante

5.1. Introducción

En el capítulo anterior se presentó un método con el que predecir el PIB regional de la Comunidad Valenciana utilizando solo la información obtenida de la red social Twitter. Esta información se procesó mediante una arquitectura de red para determinar si el conjunto de opiniones de un trimestre indicaba una posible subida o bajada de este valor. A pesar de obtener resultados interesantes utilizando solo información textual, la experimentación realizada para probar el método fue limitada, considerando solo una región de España (Ortega-Bastida *et al.*, 2020) y sin analizar de manera exhaustiva la robustez del método ante ruido o fluctuaciones inesperadas. Además, los resultados obtenidos tampoco resultan competitivos si se comparan con las predicciones basadas en datos econométricos.

En este capítulo se pretende extender esta propuesta para incorporar una nueva fuente de información utilizando para esto una aproximación multimodal. El objetivo principal es obtener una solución robusta, que mejore los resultados anteriores y que además pueda realizar predicciones en frecuencias de tiempo y niveles geográficos que las alternativas existentes apenas consideran.

Cabe recordar que, como ya se introdujo en capítulos anteriores, las Oficinas Nacionales de Estadística de todo el mundo calculan y publican los indicadores económicos oficiales principalmente a nivel nacional y de manera anual, considerando solo en ocasiones unidades geográficas o frecuencias inferiores. En países como España, el PIB a nivel regional se calcula solo anualmente, y no se proporciona una cifra oficial para las provincias o ciudades. Sin embargo, dado que la formulación de políticas a menudo requiere información con mayor frecuencia (trimestral o mensual) y también para unidades geográficas más desagregadas (regiones o ciudades), se suelen utilizar metodologías de interpolación que no siempre son públicas y/o consistentes. Estas metodologías se basan en indicadores macroeconómicos que bien no están disponibles trimestralmente o se publican meses después de finalizar el trimestre, lo que implica tener que esperar para calcular este valor.

Como se introdujo en el capítulo 2, hasta donde sabemos, el único trabajo

metodológico que considera el pronóstico trimestral del PIB a nivel regional es Cuevas *et al.* (2015), que en España es implementado por la Autoridad Independiente de Responsabilidad Fiscal (AIReF). Sin embargo, sí que podemos encontrar otras aproximaciones basadas en ML para el pronóstico del PIB de todo un país y de manera anual (Junoh, 2004; Richardson *et al.*, 2018; Jung *et al.*, 2018; Cicceri *et al.*, 2020), los cuales, basándose también en indicadores macroeconómicos, consiguen mejorar los resultados de los enfoques estadísticos tradicionales. Para esta misma tarea también existen propuestas basadas en otras fuentes de información, como el trabajo de Jaromír (2016) a partir de datos históricos, o el de Indaco (2020) a partir de los mensajes e imágenes publicados en Twitter. Sin embargo, en estos últimos trabajos se obtiene una baja precisión, ya que estas fuentes de información son insuficientes por sí mismas, pues no incorporan los valores del PIB de los últimos meses y no pueden, por tanto, capturar el valor real y la tendencia del PIB.

La aproximación multimodal propuesta en este capítulo se basa únicamente en series temporales con valores históricos del PIB y en información extraída de la red social Twitter. Este enfoque multimodal busca mejorar el resultado final al permitir que el método detecte información que no es visible en cada modalidad individual y, en consecuencia, tome decisiones más informadas. Al combinar estas fuentes de información podemos capturar tanto la tendencia del valor del PIB como las posibles fluctuaciones en el último trimestre gracias a las opiniones vertidas en los comentarios de Twitter.

También es importante destacar que esta propuesta no se basa en indicadores económicos que, como hemos visto, se publican tiempo después de la finalización del trimestre (por ejemplo, la información de la encuesta de población activa para el segundo trimestre, abril-junio, se publica a finales de julio). En comparación, la información de Twitter está disponible casi instantáneamente, lo que permite que el método haga predicciones con alta frecuencia, incluso semanalmente. Aunque nos centramos en regiones, cabe señalar que este procedimiento también puede estimar las condiciones económicas en otros niveles geográficos, como provincias o ciudades, para los que rara vez se dispone de indicadores económicos. Además, la propuesta también se puede utilizar incluso en ausencia de datos históricos del PIB, ya que el método de la AIReF Cuevas *et al.* (2015) nos permite calcularlos (al ser datos históricos, la información necesaria ya está

disponible, por lo que esto no retrasa ni influye en la frecuencia de predicción del método).

En resumen, este capítulo hace las siguientes contribuciones:

- Como primera contribución importante, la información de Twitter se combina con la evolución histórica del PIB utilizando un enfoque multimodal basado en una red neuronal recurrente. Esto permite que el método propuesto pueda predecir el PIB por regiones o cualquier otra unidad geográfica y con diferentes frecuencias, ya que no depende de la disponibilidad de indicadores económicos para el período a pronosticar.

- En segundo lugar, evaluamos el método considerando cuatro regiones de España con diferentes tipos de estructura económica, incluyendo casos uniprovinciales y pluriprovinciales, como son las comunidades autónomas de Madrid, Andalucía, la Comunidad Valenciana y Murcia. Además, hemos utilizado un conjunto de datos mucho mayor que otros métodos del estado del arte. Los resultados muestran que el método propuesto puede capturar la evolución del crecimiento del PIB regional y mejorar la precisión del pronóstico de varias aproximaciones *baseline* y métodos del estado del arte.

- Como tercer aporte, se realiza una evaluación exhaustiva considerando diferentes tipos de arquitecturas de redes, realizando un estudio de ablación y analizando la robustez del método. Además, añadimos un método de filtrado automático de los tweets que nos permite mejorar la precisión al excluir los mensajes que se alejan del consenso general, pudiendo hacer un análisis de aquellos tweets que más contribuyen a la predicción (y los que menos).

- Como aporte final, también se evalúa el método propuesto en cuanto a su capacidad para predecir el impacto del COVID-19 en las economías regionales a través de los tweets redactados en los dos primeros trimestres de 2020. Este es un caso particularmente interesante de analizar, ya que su efecto en el PIB no se puede detectar utilizando solo valores históricos.

Universitat d'Alacant
Universidad de Alicante

El resto de este capítulo está organizado de la siguiente manera. La sección 5.2 presenta la metodología propuesta utilizada para predecir el PIB. La sección 5.3 explica la configuración de los experimentos, incluyendo la descripción del conjunto de datos utilizado. Seguidamente, la sección 5.4 presenta los resultados obtenidos de la experimentación realizada. Finalmente, la sección 5.5 aborda nuestras conclusiones y las posibles líneas de trabajo futuro.

5.2. Metodología

Proponemos extender el método presentado en el capítulo 4 a una arquitectura compuesta de dos etapas de procesamiento para estimar el PIB regional utilizando tweets y valores históricos del PIB. Estas dos fuentes de información se combinan mediante un método multimodal para mejorar la predicción final. Los tweets se procesan inicialmente para obtener un indicador de la tendencia del PIB y luego esta información se combina con los valores históricos del PIB para ajustar la predicción.

La Figura 5.1 muestra el esquema general del método propuesto. En la primera etapa, el proceso sigue una serie de pasos similares a los de la metodología presentada en el capítulo anterior (pasos del 1 al 3 de esta figura) para transformar los tweets en una representación relacionada con el PIB. Es decir, una representación basada en la positividad o negatividad de los comentarios, lo que permite determinar si indica una posible subida o bajada del PIB. Una de las novedades incluidas en esta etapa, consiste en procesar adicionalmente esta representación codificada (paso 4) para agrupar opiniones similares (aquellas que logran un mayor consenso) y eliminar de esta forma los valores atípicos (es decir, opiniones aisladas). En la segunda etapa, la predicción obtenida a partir de los tweets (paso 5) se combina con la secuencia de valores históricos del PIB (paso 6) para mejorar la estimación final. En las siguientes secciones se proporcionará una explicación detallada de estos pasos.

Universitat d'Alacant
Universidad de Alicante

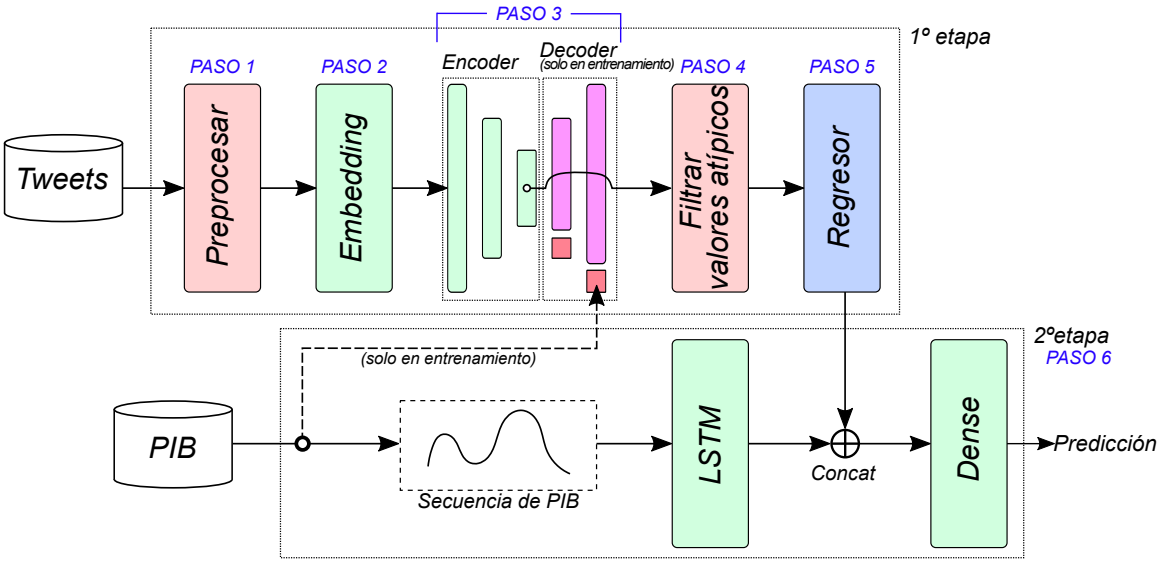


Figura 5.1.: *Esquema general del método propuesto. El orden de los pasos se marca en azul.*

5.2.1. Paso 1. Preprocesar tweets

De igual forma a como se explicó en la sección 4.2.1 del capítulo anterior, el primer paso del método es preprocesar los tweets de entrada (eliminar palabras especiales de Twitter, eliminar enlaces, etiquetas hash, corregir ortografía, etc.). En dicho capítulo, ya se adelantó que estos tweets no son de propósito general, sino que seleccionamos las cuentas de Twitter más relevantes en el campo de la economía. También es importante señalar que los tweets de estas cuentas se filtran mediante una lista de palabras clave. La sección 5.3.1 detalla el proceso seguido para seleccionar estas cuentas, incluyendo el listado completo de cuentas utilizadas junto con la lista de palabras clave (ambos ampliados para esta versión).

5.2.2. Paso 2. Embedding

Con respecto a la técnica de *word embedding* utilizada, en el capítulo 4 vimos que los tweets, al ser cadenas de texto en las que el significado de las palabras está condicionado por el contexto de la oración, la codificación neuronal era

una opción preferible al uso de diccionario o a la codificación tipo *one-hot*, ya que se perdía el sentido semántico de las palabras u oraciones. A este respecto, en el capítulo anterior evaluamos varios enfoques del estado del arte (Sense2Vec (Trask *et al.*, 2015), NNLM (Abadi *et al.*, 2015), LASER (Chaudhary *et al.*, 2019) y SBERT (Reimers y Gurevych, 2020), cuyas descripciones se pueden encontrar en la sección 4.2.2), determinando que finalmente el método que mejores resultados obtuvo fue SBERT.

Además, al igual que en el capítulo anterior, el vector de características resultante de esta codificación se normaliza utilizando ℓ_2, lo que conlleva una ligera mejora en la predicción.

5.2.3. Paso 3. Codificación de los tweets

El objetivo de este paso es transformar los *embeddings* obtenidos previamente para los tweets a una representación relacionada con el valor del PIB del momento que se escribieron. Para esto, al igual que en el capítulo anterior, se propone el uso de una arquitectura tipo autoencoder entrenada de manera simultánea para minimizar el error de reconstrucción de la información textual de los tweets junto con su correspondiente valor de PIB.

En la sección 4.2.3 experimentamos con cuatro arquitecturas para la definición de las capas de esta arquitectura: *Vanilla* con solo capas tipo *fully connected*, CNN-1D, LSTM y BiLSTM. Como vimos, la topología que obtuvo el mejor resultado fue la basada en capas BiLSTM, por lo que para este trabajo nos centraremos también en este tipo de capas.

Una vez el autoencoder ha sido entrenado, la parte del decodificador (*decoder*) se elimina de la red, lo que significa que, durante la inferencia, los *embeddings* calculados en el paso 2 para los tweets se pasan solo a través del codificador (o *encoder*), devolviendo la representación intermedia de la red como salida.

5.2.4. Paso 4. Filtrado de valores atípicos

Para esta nueva versión del método, se ha añadido un paso intermedio entre el resultado de la codificación del autoencoder y la predicción del regresor. Con el fin de mejorar la precisión del método, los vectores de características extraídos del espacio latente de la red son filtrados utilizando el método *Local Outlier Factor* (LOF) propuesto por Breunig *et al.* (2000) para eliminar valores atípicos. Este algoritmo consiste en hallar puntos anómalos dentro de un conjunto de datos midiendo la desviación local de estos puntos con respecto a la densidad local de los datos vecinos. Esto permite que los vectores cuyos valores se encuentran más alejados del valor medio general sean excluidos en la predicción final, lo que reduce el ruido en el conjunto de datos empleado.

La idea de este proceso de filtrado es, por tanto, procesar el conjunto de tweets de un trimestre a partir de su representación codificada, la cual, en principio, indica si la opinión representa una posible subida o bajada del valor del PIB. De esta manera, es posible determinar las opiniones atípicas que se salen del consenso general. Al eliminar estas opiniones conseguimos una predicción más robusta y, a la vez, nos permite conocer tanto los tweets que más han influenciado en la predicción como los que más se han alejado de la opinión general y cuya aportación solo genera ruido en la predicción. En la sección 5.4.6 se realiza un análisis sobre este proceso, mostrando gráficamente los *clusters* de opiniones formados, además de realizar un análisis de las opiniones más y menos influyentes para cada año considerado.

Universitat d'Alacant
Universidad de Alicante

5.2.5. Paso 5. Regresor

A continuación, a partir de las opiniones filtradas obtenidas en el paso anterior, se realiza un proceso de regresión para calcular una estimación inicial del PIB. Para este paso, y al igual que en el capítulo anterior, también se consideraron varias alternativas, como el uso de la salida del autoencoder con y sin filtrado, una regresión lineal simple, un regresor Random Forest (RFr) (Breiman, 2001), además de ϵ-Support Vector Regression (ϵ-SVR) (Vapnik, 1998).

Para entrenar estos métodos se preprocesaron los tweets del conjunto de entrenamiento, realizando los pasos del 1 al 4 de este método. Es decir, a partir de las opiniones codificadas y filtradas y usando como etiquetado el valor del PIB asociado. En la sección de experimentación veremos los resultados de la comparación de estos métodos. Adelantar, que al igual que en la aproximación anterior, el método que obtuvo el mejor resultado fue ϵ-SVR y, por tanto, es el método seleccionado para ser utilizado en esta propuesta.

5.2.6. Paso 6. Red recurrente multimodal

En la segunda etapa del método propuesto, la predicción obtenida mediante tweets se combina con una serie temporal con los valores históricos del PIB, para, de esta forma, mejorar la estimación final. El interés por la fusión multimodal ha surgido de los beneficios que esta puede brindar, ya que el uso de diferentes tipos de información nos permite capturar información complementaria (algo que no es visible en cada modalidad de manera individual) o incluso filtrar ruido o corregir valores erróneos (Lahat *et al.*, 2015). En nuestra tarea, la información histórica del PIB fue capaz de corregir el pronóstico para ajustarlo a la tendencia de los últimos trimestres. La información obtenida de los tweets fue igualmente capaz de ajustar la tendencia al alza o a la baja si los comentarios pronosticaban una mayor subida o bajada.

Universitat d'Alacant
Universidad de Alicante

Como vimos en el capítulo 2, en la literatura se pueden encontrar múltiples enfoques con los que combinar información multimodal, como técnicas de fusión tempranas, tardías e híbridas, junto con enfoques agnósticos y basados en modelos (Baltrušaitis *et al.*, 2019). Procesar la información por separado utilizando métodos agnósticos del modelo y su combinación temprana, tardía o híbrida puede tener varias ventajas (como ya comentamos en el capítulo 3), sin embargo, también implica varios problemas: el uso de métodos unimodales que originalmente no fueron diseñados para la multimodalidad, y el ajuste de la contribución de cada modalidad, que no es una tarea fácil, y es por eso que muchos de estos métodos generalmente consideran solo la misma contribución. Los enfoques basados en modelos, como los enfoques basados en kernel, los modelos gráficos o las redes neuronales, abordan explícitamente el proceso

de fusión en su construcción. Esto, por lo tanto, permite aprender funciones de fusión más complejas, por ejemplo, ponderando la contribución de las modalidades a partir de sus características (Ngiam *et al.*, 2011).

Por lo tanto, para esta propuesta nos centraremos en un enfoque basado en el modelo con el objetivo de conseguir una mejor fusión de la información, que pueda aprender relaciones más complejas entre los datos, así como la contribución de cada modalidad en función de sus características. La topología propuesta se compone específicamente de una red neuronal con una serie de capas recurrentes para procesar la secuencia con los valores históricos del PIB. Este resultado se concatena posteriormente con el obtenido en la primera etapa del método, es decir, con la predicción basada en el consenso de opiniones en los tweets (paso 5, descrito en el apartado anterior). Esta concatenación finalmente se pondera mediante el empleo de una capa FC que aprende la contribución de cada modalidad. La Figura 5.1 muestra el diagrama general de esta arquitectura multimodal, mostrando cómo se combinan las dos fuentes de información en los pasos 5 y 6.

Como ocurre con el autocoencoder híbrido, la topología de esta red puede ser bastante variada. En este caso, consideraremos los tipos de capa Conv1D, LSTM y BiLSTM para la primera parte de la red, y una capa densa o FC para la parte final. Se pueden encontrar más detalles sobre las topologías comparadas en la sección 5.3.2, mientras que los resultados de la comparación se muestran en la sección 5.4.3. En este caso, el mejor resultado se obtuvo utilizando las capas LSTM. La topología seleccionada, por tanto, consta de dos capas LSTM con 32 y 16 filtros cada una, seguidas de la capa de concatenación, y una capa densa con 32 neuronas. En todas las capas se utilizó *ReLU* como función de activación, a excepción de la salida, para la cual se utilizó una función de activación lineal.

La etapa de entrenamiento de esta red se realizó de forma *end-to-end*, es decir, entrenando todo el modelo a la vez, mediante descenso de gradiente estocástico (Bottou, 2010) y considerando la función de pérdida MSE. El entrenamiento se realizó durante 300 épocas con un tamaño de *batch* de 32 muestras y utilizando la función de optimización adaptativa propuesta por Kingma y Ba (2015).

Universitat d'Alacant
Universidad de Alicante

5.2.7. Etapas de entrenamiento e inferencia

En esta sección se resumen los pasos explicados anteriormente, detallando el proceso seguido para entrenar las redes y preparar los datos, junto con el uso del método en la etapa de inferencia.

El Algoritmo 1 muestra la formalización del proceso de entrenamiento usando pseudocódigo. El método recibe como entrada el conjunto de entrenamiento T, el encoder `Enc` y el decoder `Dec` (según la topología), la red multimodal `MNet`, y los parámetros de entrenamiento (épocas e y el tamaño de *batch* b). El conjunto de entrenamiento $T = \{(x_i^t, \mathbf{x}_i^g, y_i)\}_{i=1}^{|T|}$ contiene $|T|$ muestras de entrenamiento compuestas de tweets (x_i^t), la secuencia con los valores históricos del PIB de longitud k ($\mathbf{x}_i^g = (x_{i,1}^g, x_{i,2}^g, \ldots, x_{i,k}^g)$) y la etiqueta y_i correspondiente al valor del PIB en la fecha de publicación del tweet de la muestra.

Algoritmo 1: Etapa de entrenamiento.

Input : $T \leftarrow \{(x_i^t, \mathbf{x}_i^g, y_i)\}_{i=1}^{|T|}$
`Enc`, `Dec` $\leftarrow$ Topologías de Encoder y decoder
`MNet` $\leftarrow$ Topología de red multimodal
$e, b \leftarrow$ Épocas, tamaño de batch

1 $T \leftarrow$ `Preprocesar`(T) ▷ Paso 1 - Sección 5.2.1
2 $T_p \leftarrow \ell_2($`UKPLab`$(T))$ ▷ Paso 2 - Sección 5.2.2
3 `Enc`, `Dec` $\leftarrow$ Entrenar `Enc` y `Dec` con $\{T_p, e, b\}$ ▷ Paso 3 - Sección 4.4.1
4 $S \leftarrow \varnothing$
5 **foreach** x_i^t *in* T_p **do**
6 | $S \leftarrow S \cup$ `Enc`(x_i^t)
7 **end foreach**
8 $S_f \leftarrow$ `LOF`$(S, neigh = 35, cont = 0{,}5)$ ▷ Paso 4 - Sección 5.2.4
9 ϵ-`SVR` $\leftarrow$ Entrenar ϵ-`SVR` con $\{S_f\}$ ▷ Paso 5 - Sección 5.2.5
10 $S_f \leftarrow \epsilon$-`SVR`(S_f)
11 `MNet` $\leftarrow$ Entrenar `MNet` con $\{seq(T, k), S_f, e, b\}$ ▷ Paso 6 - Sección 5.2.6
12 **return** `Enc`, ϵ-`SVR`, `MNet`

El conjunto de entrenamiento primero se preprocesa, se transforma en un vector de características n-dimensional (*embedding*) y se normaliza siguiendo los pasos descritos en las secciones 5.2.1 y 5.2.2. Este resultado se utiliza como base para entrenar el autoencoder (formado por `Enc` y `Dec`, línea 3) utilizando

Universitat d'Alacant
Universidad de Alicante

el conjunto T_p con los *embeddings* normalizados y el valor del PIB asociado (y_i). El *encoder* entrenado `Enc` se usa posteriormente para transformar los tweets de T_p en una representación relacionada con el PIB (obteniendo el conjunto S como resultado, líneas 4-7). Seguidamente, este conjunto se filtra para eliminar los valores atípicos (línea 8) y se usa para entrenar el método de regresión ϵ-`SVR` (línea 9). Finalmente, la red multimodal `MNet` se entrena usando dos entradas: secuencias de longitud k con valores históricos del PIB (obtenidos del conjunto de entrenamiento usando la función $seq(T, k)$) y el conjunto S_f con la predicción obtenida de los tweets (línea 10). Para seleccionar el tamaño de las secuencias de entrada, se realizó una experimentación inicial considerando longitudes en el rango $k \in [1, 5]$, determinando finalmente que el tamaño $k = 2$ (es decir, los dos trimestres anteriores al considerado) era con el que se obtenían los mejores resultados, ya que esta longitud parece resultar suficiente para ajustar el valor y la tendencia para la predicción final del PIB.

El algoritmo de entrenamiento devuelve como resultado `Enc`, ϵ-`SVR` y `MNet`, que son las partes del método que se han entrenado y que se utilizan en la etapa de inferencia. El Algoritmo 2 muestra la formalización de esta última etapa. En este caso, además del algoritmo previamente entrenado, recibe como entrada el conjunto de prueba $Q = \{(x_i^t, \mathbf{x}_i^g)\}_{i=1}^{|Q|}$ con los tweets del trimestre para el que deseamos predecir el PIB (x_i^t) y las secuencias de valores del PIB de los trimestres anteriores. El algoritmo procesa inicialmente los tweets en el conjunto Q (líneas 1-4) para obtener una predicción S_f con ellos. Luego usa `MNet` para combinar S_f con los valores históricos del PIB (usando la función $seq(Q, k)$, línea 5) para hacer la estimación final.

Universitat d'Alacant
Universidad de Alicante

Algoritmo 2: Etapa de inferencia.

Input : `Enc`, ϵ-`SVR`, `MNet`
$Q \leftarrow \{(x_i^t, \mathbf{x}_i^g)\}_{i=1}^{|Q|}$

1 $Q_p \leftarrow \ell_2(\texttt{UKPLab(Preprocesar}(Q)))$ ▷ Pasos 1 y 2
2 $S \leftarrow \texttt{Enc}(Q_p)$ ▷ Paso 3
3 $S_f \leftarrow \texttt{LOF}(S, neigh = 35, cont = 0{,}5)$ ▷ Paso 4
4 $S_f \leftarrow \epsilon\texttt{-SVR}(S_f)$ ▷ Paso 5
5 **return** $\texttt{MNet}(seq(Q, k), S_f)$ ▷ Paso 6

5.3. Configuración para los experimentos

5.3.1. Conjunto de datos

Para la evaluación del método se crearon conjuntos de datos para cuatro de las regiones más importantes de España, como son Madrid, Comunidad Valenciana, Andalucía y Murcia. Cada conjunto de datos contiene tweets relacionados con el campo de la economía etiquetados con los valores del PIB correspondientes. Al igual que en el capítulo anterior, cada tweet se etiquetó con la diferencia entre el PIB del trimestre en el que se escribió respecto al PIB del trimestre anterior. Esta diferencia se calculó como porcentaje y se normalizó en el rango $[-1, 1]$.

Dado que los datos del PIB regional proporcionados por IAES están disponibles solo hasta 2016 Tituaña Castillo (2016), se recopilaron los tweets también para ese período. Descargamos 1.500 tweets para cada trimestre y región en el período 2012 a 2016, lo que hace un total de 120.000 tweets, 30.000 para cada región. En este caso, los valores proporcionados por IAES están calculados mediante el método propuesto en Cuevas *et al.* (2015), que es el que utiliza la AIReF en España debido a su precisión para estimar el PIB. Aunque utiliza indicadores que se publican meses después del final del trimestre, esto no es un inconveniente para calcular los valores históricos que necesita nuestra propuesta.

Para recopilar los tweets de cada una de las cuatro regiones consideradas, nos centramos en las cuentas de Twitter más influyentes y relevantes en el ámbito económico (incluidas las de economistas, líderes políticos, periódicos, etc.). Para seleccionar estas cuentas, buscamos primero las cuentas oficiales relevantes, junto con los *rankings* oficiales de las cuentas más populares de cada región. Los criterios seguidos se resumen a continuación:

1. *Ranking* de los periódicos más leídos de la región.
2. *Ranking* de las televisiones y emisoras de radio regionales seguidas con mayor frecuencia.
3. *Ranking* de los economistas más influyentes según revistas especializadas.
4. *Ranking* de las instituciones de estudios económicos más importantes.

Universitat d'Alacant
Universidad de Alicante

5. Cuenta oficial de la administración autonómica, Ministerio de Economía, Hacienda, Turismo, Energía y Empleo.
6. Cuentas del presidente regional y de los directores que ocupen los cargos indicados en el punto 5.
7. Cuentas de los partidos políticos representados en el parlamento autonómico y líderes políticos.
8. Cuentas de consejeros y portavoces de áreas del gobierno relacionadas con la economía (como el consejero de hacienda).

Se seleccionó un promedio de 80 cuentas para cada región. Las tablas 5.1, 5.2, 5.3 y 5.4 muestran el listado completo de las cuentas elegidas para cada región junto con su tipo.

Tabla 5.1.: *Cuentas de Twitter seleccionadas para la Comunidad Valenciana.*

VALENCIA
Periódicos, radio y televisión: @lasprovincias, @larazon_cv, @elmundocv, @elpais_valencia, @ABC_CValenciana, @valenciaplaza, @levante_emv, @eldiariocv, @ABC_Alicante, @elperiodicoaqui, @informacion_es, @elmundoalc, @epmediterraneo, @csinformacion, @elmundocs, @LevanteCastello, @alicanteplaza, @castellondiario, @Informacion_TV, @radiovalencia, @apunt_media, @Levante_TV, @AlacantiTv, @radioalicante, @COPEAlicante, @Informacion_TV, @copecastellon, @OCR_Castellon, @radiocastellon
Periodistas y expertos económicos: @elEconomistaCV, @Economia_3, @InstitutoINECA, @COEV_COM, @EconomistasOrg, @Economistas_ALC, @CEV_CV, @AEFALICANTE, @cerscv, @Ivie_news, @AJEValencia, @jvvillaverde, @JMaudos, @DaValero, @Fpiera, @Boronavarro, @ndelatower, @Boronavarro, @MatildeMas, @bernardoguzman, @AnaDuCos, @municiman, @_ctoledo_, @borras_xavier, @HectrSanjuan, @ndelatower, @paco__moreno, @carlosnc7, @jlobrador, @Monrealmartinez, @jmanpastor
Partidos políticos, administración pública y sectores de gobierno: @PSOEValencia, @ppcv, @ximopuig, @isabelbonig, @RubenIBB, @monicaoltra, @joanbaldovi, @SocialistesVal, @popularescorts, @alicantepp, @JuandeNavarro, @popularsval, @mjosecatala, @alfcastello, @JorgeBellver, @javiermoliner, @Julia_company, @GVAhisenda, @GVAivace, @GVAeconomia, @generalitat, @GVA112, @dipuAlicante, @alicanteayto, @dipcas, @camaravalencia, @CamaraAlicante

Los tweets descargados de estas cuentas también se filtraron para que contuviesen una serie de palabras clave relacionadas con el PIB, como se explica en la sección 5.2.1. En la Tabla 5.5 se puede consultar la lista completa de las palabras clave seleccionadas.

Tabla 5.2.: *Cuentas de Twitter seleccionadas para Madrid.*

MADRID
Periódicos, radio y televisión: @el_pais_madrid, @ElMundo_Madrid, @abc_madrid, @eldiario_Madrid, @somosmalasana, @EPMadrid, @Madridiario, @madridactual, @Gacetinmadrid, @ConfiAutonomico, @ElSaltoMadrid, @ondamadrid, @telemadrid, @SERMadridNorte, @RadioMadrid, @SERMadridSur, @ondaceromn, @InformativosT
Periodistas y expertos económicos: @UCM_ECONOMICAS, @EconomiaUCM, @UAM_Economicas, @mercamadrid_hoy, @CamaradeMadrid, @_CEIM_, @IEE_madrid, @CircularECMad, @cemad_es, @monicamelle, @DavidRobles4, @diegodelacruz, @ChicoDeDerechas, @1FelipeBenjumea
Partidos políticos, administración pública y sectores de gobierno: @ppmadrid, @PSOEAlcobendas, @IdiazAyuso, @AlmeidaPP_, @EsperanzAguirre, @ManuelaCarmena, @ccifuentes, @Cs_Madrid, @CsarZafra, @ignacioaguado, @begonavillacis, @angelgarridog, @psoe_m, @UGTMadrid, @CCOOMadrid, @psacera, @PSOEMadridAyto, @conJoseMFranco, @MasEconomia_, @CCOOhaciendamad, @GrupoPPMadrid, @DeMairena, @ComunidadMadrid, @MADRID, @ForosLocales, @EconomiaAytoMad, @sansecomunica, @ALCBDS_Ayto, @HaciendaCoslada, @Ayto_Coslada, @Ayto_Las_Rozas, @AytoAlcalaH, @EmpleoCMadrid, @aempleomadrid, @DGobiernoMadrid, @TurismoMadrid, @MadridCapital

Tabla 5.3.: *Cuentas de Twitter seleccionadas para Andalucía.*

ANDALUCÍA
Periódicos, radio y televisión: @elpaisandalucia, @diariosevilla, @elCorreoWeb, @eldiarioAnd, @DiarioSUR, @elmundoand, @abcdesevilla, @EPAndalucia, @andaluciainf, @MalagaEP, @cordoba, @SER_Malaga, COPESevilla, @RadioCadiz, @CSurNoticias, @CanalSurRadio, @adirecto
Periodistas y expertos económicos: @CCAndalucia, @CSIFAndalucia, @Agenda_Empresa, @ProgramaMinerva, @Manuj_Hidalgo, @EconomistasSE, @LoyolaEcon, @DValderasS, @Gllds
Partidos políticos, administración pública y sectores de gobierno: @psoedeandalucia, @susanadiaz, @ppandaluz, @JuanMa_Moreno, @PP_JavierArenas, @UGT_Andalucia, @rogeliovelascop, @pacodelatorrep, @ruiz_espejo, @EmpleoJunta, @DGobAndalucia, @SubdeGobSevilla, @TurismoAND, @diputacioncadiz, @TurismoJerez, @RadioCordobaSER, @AndaluciaJunta, @MalagaJunta, @JuntaSevilla, @HaciendAND, @FomentoAND, @EconomiaAnd, @EnergiaAnd, @CCAndalucia, @CESAndalucia, @IDEAJunta, @AndaluciaDig, @SAEmpleo

La Figura 5.2 analiza la longitud de los tweets. Como puede verse, la longitud media por región está entre 80 y 100 caracteres, siendo 91,51 $\pm$ 24,77 la media

Tabla 5.4.: *Cuentas de Twitter seleccionadas para Murcia.*

MURCIA
Periódicos, radio y televisión: @diariolaopinion, @ocrmurcia, @SER_Murcia, @ORMurcia, @Murdiario, @cope_murcia, @PopularTvMurcia, @RTVEMurcia, @eldiariomurcia, @La7_tv, @laverdad_es, @DailyMur, @EPMurcia, @infoRMurcia, @larazon_murcia, @abc_murcia, @SER_Cartagena, @LaOpiCartagena, @laverdad_ct
Periodistas y expertos económicos: @EconomicasUMU, @umumuac, @Catedra_RSC, @CEF_MN, @PedroJ_Cuestas, @murciaeconomia, @hostemur_, @AFRomero_, @ucomur, @AJERegionMurcia, @CeeiMurcia, @circuloeconomia, @economistasrm, @economistasrm, CEEE_UMU, @juanpedroamusal, @UmuODSesiones, @UMUEmprende, @RmadridRamon, @ICREFRM, @UPCTnoticias, @MurmiaCom, @Amefmur, @marcosantonr, @JorgeEduardoMP, @coieum, @FranciscoMiro60, @RLinaresDiego, @IMartinezConesa
Partidos políticos, administración pública y sectores de gobierno: @gppmur, @Ballesta_Murcia, @jacello, @AytoMurcia, @PPRMurcia, @PSOE_RM, @GrupoPSOEMurcia, @MerBernabe, @CsRegionMurcia, @CCOORM, @UGTMurcia, @podemosmurcia, @mjoserosolivo, @pepelopezct, @cienfuegosmur, @CarolinaE_G, @regiondemurcia, @camaramurcia, @TurismoRMurcia, @sef_carm, @asambleamurcia, @AJERegionMurcia, @Fomento_RM, @Empleo_RM, @Hacienda_RM, @Empresa_RM, @RM_Presidencia, @TurismoRM_, @CamaraCartagena, @MurciaEmpresa, @murciaBORM, @TuriCartagenaES

general para las cuatro regiones. La longitud máxima es de 140 caracteres (límite máximo establecido por Twitter) y la longitud mínima es de 7.

La Tabla 5.6 muestra un resumen de la composición de los conjuntos de datos preparados para la evaluación del método propuesto, detallando el número de muestras, cuentas y palabras clave. También incluye estadísticas sobre la longitud de los tweets utilizados. Además, en la sección de experimentación se pueden consultar gráficos con los valores del PIB de cada una de estas regiones (ver Figura 5.5) así como algunos ejemplos de tweets de estos conjuntos de datos (ver Tablas 5.12 y 5.14).

5.3.2. Topologías de la red multimodal

La Tabla 5.7 muestra las diferentes topologías consideradas para la red multimodal utilizada en la segunda etapa del método. En este caso se evaluaron

Tabla 5.5.: *Listado de palabras clave utilizadas para filtrar los tweets.*

Palabras clave de Twitter:
PIB, crecimiento económico, previsiones crecimiento, deuda publica, deuda privada, deuda familias, deuda externa, creación empleo, contratación, destrucción empleo, productividad, competitividad, producto interior bruto, déficit, actividad económica, hipotecarios, hipotecas, precio vivienda, compra vivienda, venta vivienda, revalorización, valor vivienda, cerámica, industria, sector industrial, sector cerámico, sector servicios, sector construcción, sector agricultura, sector agrícola, sector automóvil, sube paro, baja paro, descenso paro, mejora economía, exportaciones, importaciones, valor acciones, inversión, ingresos turismo, recaudación impuestos, gasto familia, beneficios empresariales, beneficio empresas, ingresos fiscales, poder adquisitivo, plan pive, previsiones económicas, IPC, IPI, producción industrial, precio gasolina, precio luz, proveedores, tipo de interés, prima riesgo, IRPF, costes laborales, inflación, euribor, bursátil, carburantes, mercados, tasa anual, gasto social, impuestos, precios, salarios, sector azulejo, acreedores, producción empresas, ingreso autonómico, cierre empresas, destrucción empresas, venta coches, compra coches, recaudación IVA, compra automóviles, venta automóviles, compra vehículos, venta vehículos, sector inmobiliario, FMI, BCE, banco españa, empeora economía, sueldos, petroleo, precio gas, indicador confianza, consumo familia, acciones suben, acciones bajan, TAE, tasa anual, cuentas publicas, contrato temporal, tasa actividad, EPA, población activa, flujo crédito, tiempo parcial, tiempo completo, creación empresas, contrato fijo, contrato indefinido, intereses deuda, sube producción, incertidumbre económica, acreedores, consumo luz, consumo eléctrico, sube consumo, baja consumo, valor bolsa, oferta demanda, ciclo económico, desigualdad, despido, negocio, competencia, expectativas, remuneración, incentivos, comercio, bancos, bancario, afiliados seguridad social, duración desempleo, duración paro, dinero, financiero, rentabilidad, mercantil, recesión, depresión, monetario, alimentaria, precio alquiler, economía valencia, crecimiento valencia, administración, clase media, riqueza, propiedad, baja pobreza, gasto familia, ahorro familia, desempleo, pago deuda, afiliación seguridad social, estructura salarial, tasa participación, SEPE, cambio estructural, reforma estructural, macro prudencial, índice IESE

varios tipos de capas para la primera parte de la red, manteniendo la misma parte final para todas. La operación “Concat” conecta el resultado de la primera parte de esta red con el resultado obtenido tras procesar los tweets en la primera etapa del método propuesto.

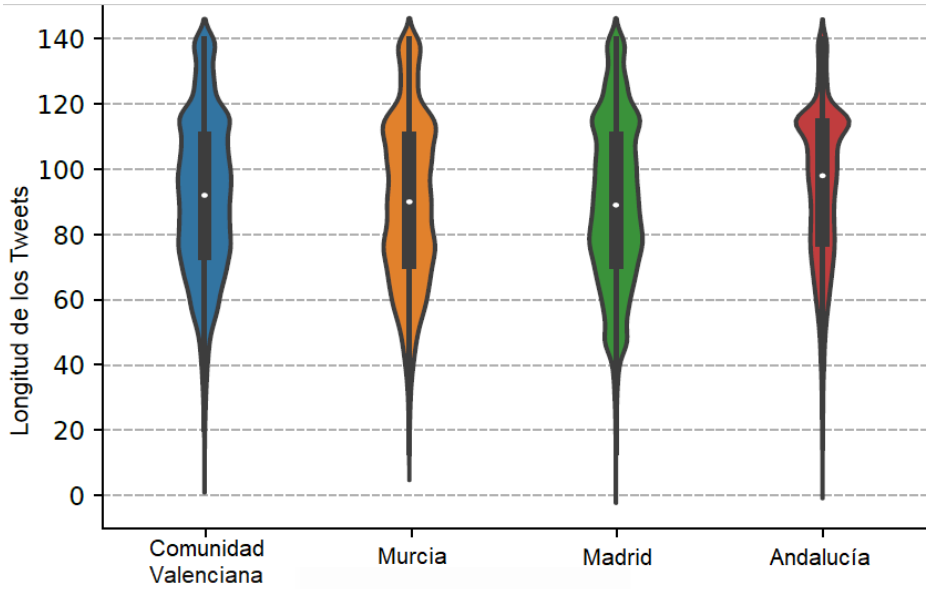


Figura 5.2.: *Análisis de la distribución de longitud de los tweets para cada uno de los conjuntos de datos considerados.*

Tabla 5.6.: *Resumen de la composición de los conjuntos de datos utilizados en la experimentación.*

Regiones:	4 (Madrid, Comunidad Valenciana, Andalucía y Murcia)
Periodo:	2012–2016 (5 años)
Total de tweets descargados:	120.000
Tweets por región:	30.000
Tweets por trimestre y región:	1.500
Total de cuentas de Twitter:	320
Cuentas por región:	80
Total de palabras clave:	160
Estadísticas de longitud de los tweets:	min: 7; max: 140; media: 91,51 ± 24,77

Para la arquitectura del autoencoder del paso 3 del método nos basaremos en la mejor topología determinada en el capítulo anterior, la cual se puede consultar en la sección 4.4.1. Recordar que esta red estaba compuesta por cuatro capas (dos capas de *encoder* y dos capas de *decoder*) de tipo BiLSTM, con 64, 32, 32 y 64 neuronas cada una, respectivamente, utilizando la función de activación ReLU en todas ellas, a excepción de las dos salidas, para las cuales se utilizó una función de activación lineal.

Tabla 5.7.: *Topologías consideradas para la red multimodal. La notación seguida es la misma que la empleada en la Tabla 4.1. La operación "⊕ Concat" concatena el resultado de la capa #3 con el resultado obtenido en la primera etapa del método propuesto después de procesar los tweets (ver la Figura 5.1).*

Modelo	Capa 1	Capa 2	Capa 3	Ultimas capas
CNN-1D	Conv1D(64, 3)	Conv1D(32, 3)	Conv1D(16, 3)	⊕ Concat → FC(32) → FC(1)
LSTM	LSTM(64)	LSTM(32)	LSTM(16)	
BiLSTM	BiLSTM(64)	BiLSTM(32)	BiLSTM(16)	

5.4. Experimentos

En esta sección vamos a evaluar el método propuesto utilizando el conjunto de datos descrito en la sección 5.3.1. Primero evaluaremos los hiperparámetros del autoencoder. Seguidamente llevaremos a cabo un estudio de ablación para analizar las diferentes partes del modelo por separado. Finalmente, se utiliza la mejor configuración encontrada para predecir el PIB y analizar los resultados obtenidos. También realizaremos una evaluación con datos de los dos primeros trimestres de 2020 para predecir los efectos de la pandemia de COVID-19.

En todos los experimentos se aplicó la validación cruzada considerando 5 particiones. Para esto se utilizó un año de cada conjunto de datos para la evaluación y el resto de años para el entrenamiento (cabe recordar que los conjuntos de datos contienen datos de 5 años). Por lo tanto, los métodos comparados se entrenaron y evaluaron 5 veces, después de lo cual se calculó el resultado promedio.

5.4.1. Evaluación de los hiperparámetros

En esta sección, como ya se realizó en el capítulo anterior, evaluamos la influencia del hiperparámetro λ usado en la función de pérdida del autoencoder híbrido (ver Ecuación 4.2) para las cuatro CCAA seleccionadas. Para esto se creó una partición adicional de validación a partir de la partición de entrenamiento

con la intención de no ajustar este valor a los conjuntos de test. La Figura 5.3 muestra los resultados promedio obtenidos para este experimento (en términos de MSE más desviación estándar) para las cuatro regiones consideradas. Como puede verse, el mejor resultado se sigue obteniendo con un $\lambda = 0,1$, por lo que lo fijaremos a este valor para el resto de experimentos.

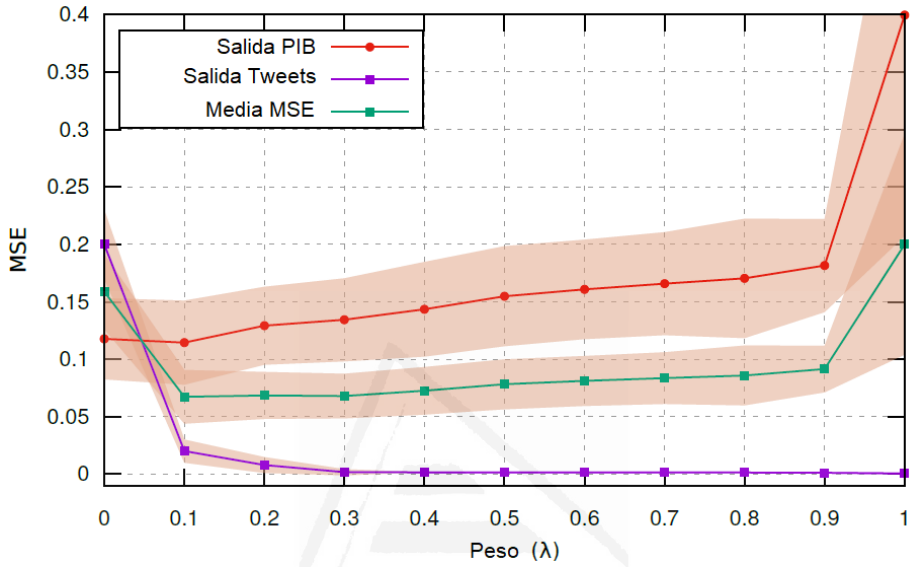


Figura 5.3.: *Resultados promedio (en términos de MSE más desviación estándar) obtenidos después de variar el parámetro λ de la función de pérdida en las cuatro CCAA consideradas.*

Universitat d'Alacant
Universidad de Alicante

5.4.2. Evaluación del proceso de filtrado

En el siguiente paso del método se realiza un proceso de filtrado sobre las representaciones generadas para los tweets de entrada del método. El objetivo de este paso es agrupar las opiniones comunes, que lleguen a un mayor consenso, y eliminar los *outliers*, que representarían las opiniones atípicas o aisladas.

Este paso se lleva a cabo utilizando el método de detección de anomalías LOF (Breunig *et al.*, 2000) explicado previamente. En nuestra implementación, establecemos un margen de 35 vecinos y una contaminación de 0.5. Como experimentación preliminar, también consideramos otro tipo de filtros (como *Isolation Forest* (Liu *et al.*, 2008), *Elliptic Envelope* (Rousseeuw y Driessen,

1999), *One Class SVM* (Schölkopf *et al.*, 2001)) y configuraciones de parámetros, que resultaron ser menos adecuados para esta tarea.

Para analizar la representación de los tweets generada por la red vamos a usar la proyección *t-Distributed Stochastic Neighbor Embedding* (t-SNE (van der Maaten, 2008), que ya vimos en la sección 3.3.2). Esto nos permitirá visualizar la distribución de características aprendidas por el espacio latente del autoencoder. La Figura 5.4 muestra dos ejemplos de dos trimestres aleatorios. Cada representación se calcula por separado, por lo que solo debería de aparecer un clúster, sin embargo, existe diversidad de opiniones para un mismo PIB en el que algunas de ellas se alejan del resto o crean pequeños clústeres (marcados en rojo). Como veremos, el resultado obtenido mejora al eliminar estos valores atípicos.

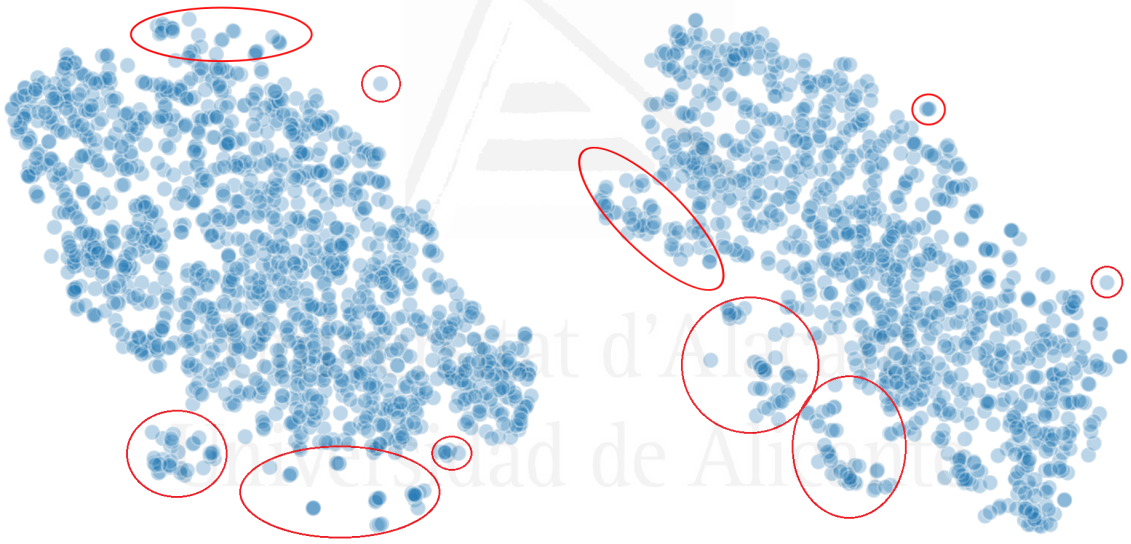


Figura 5.4.: *Distribución de características aprendida por el espacio latente del autoencoder para dos trimestres aleatorios usando la representación obtenida con t-SNE (van der Maaten, 2008).*

Ser capaces de detectar cuáles han sido los tweets que más han influenciado una predicción y cuáles menos, nos permite realizar un análisis de su contenido para conocer los temas más relevantes en la evolución de la economía. En la sección 5.4.6 realizaremos un análisis de los tweets más y menos influyentes para los cinco años de predicción.

5.4.3. Estudio de ablación

En esta sección vamos a analizar las distintas partes del algoritmo propuesto por separado con el objetivo de estudiar su contribución. Comenzaremos evaluando el filtrado y los pasos del regresor (ver Figura 5.1) usando solo tweets como entrada. Luego pasaremos a evaluar la conexión con la red multimodal y la combinación de los tweets con la secuencia de datos históricos del PIB.

Cabe señalar que en los experimentos anteriores se utilizó MSE para evaluar el error de reconstrucción del autoencoder, pero, dado que ahora vamos a evaluar una tarea diferente (la precisión de la predicción del PIB), utilizaremos la métrica MAE (*Mean Absolute Error* o Error absoluto medio) debido a su fácil interpretación en este contexto, ya que representa directamente el error en valor absoluto entre la predicción y el valor esperado.

La Tabla 5.8 muestra el resultado obtenido para la predicción del PIB utilizando la primera parte del método propuesto, es decir, el autoencoder (AE), el proceso de filtrado y el regresor considerado (ϵ-SVR), que denotamos como "*AE + Filtrado + ϵ-SVR*" en la tabla. Por tanto, es posible evaluar el resultado obtenido utilizando solo tweets.

Este resultado se compara con el obtenido usando solo el autoencoder (denotado por "*AutoEncoder (AE)*" en la tabla), con el obtenido al agregar el proceso de filtrado (*AE + Filtrado*), y el obtenido mediante el uso de un RFr (Breiman, 2001) para el paso de regresión (*AE + Filtrado + RFr*). Además, a esta comparación se añaden tres resultados *baseline*, que se calcularon utilizando los valores históricos del PIB: 1) *Baseline trimestral*: se supone que el PIB del trimestre actual es igual al del trimestre anterior; 2) *Baseline estacional*: similar al baseline trimestral, pero asignando el PIB del mismo trimestre del año anterior, y 3) *Regresión lineal*: realiza una regresión lineal con el PIB de los cuatro trimestres anteriores.

Como se puede ver en la Tabla 5.8, el método propuesto (AE + Filtrado + ϵ-SVR) obtiene el mejor resultado de todos los métodos que usan solo tweets como entrada, con la aproximación *AE + Filtrado + RFr* logrando el segundo mejor resultado. Estos resultados también muestran que las diferentes partes

del método ayudan a mejorar progresivamente el resultado: mejora en 0,03 cuando se agrega el paso de filtrado y en 0,17 cuando se agrega el regresor. Si lo comparamos con los resultados obtenidos en el capítulo anterior para la Comunidad Valenciana (0,27 de media, ver sección 4.4.2), en los que se aplicaba también el autoencoder y el regresor ϵ-SVR pero sin el proceso de filtrado, observamos que este resultado mejora en un 0,02. Por tanto, el proceso de filtrado ayuda notablemente a eliminar el ruido (ya que reduce en 0,17 el error cuando se evalúa directamente la salida del autoencoder), sin embargo, parece que el proceso de regresión también colabora a eliminar este ruido, ya que cuando se compara incluyendo este paso la reducción del error es menor.

Cuando se compara con los métodos que utilizan los valores históricos del PIB, nuestra propuesta solo consigue mejorar los enfoques más simples, como el baseline trimestral o la regresión lineal, mejorándolos en 0,99 y 0,31, respectivamente. Sin embargo, se obtiene un peor resultado cuando se compara con el baseline estacional y, por lo tanto, lo usaremos como una simple referencia en los siguientes experimentos. Este hecho resalta la importancia del uso de los datos históricos, ya que son los que realmente marcan la tendencia y el valor real del PIB. Por tanto, esperamos que al combinar esta fuente de información con la información extraída de los tweets se consiga una mejora notable en el resultado.

Universitat d'Alacant
Universidad de Alicante

Tabla 5.8.: *Evaluación del resultado obtenido (en términos de MAE) por la primera parte del método propuesto (autoencoder + filtrado + regresor, ver Figura 5.1). Nuestra propuesta se compara con otros seis enfoques: tres baselines calculados a partir de datos históricos del PIB y otras tres propuestas calculadas a partir de tweets. Los mejores resultados se marcan en negrita y se muestran por separado para cada tipo de dato de entrada.*

Datos		**Comunidades Autónomas**				
de entrada	**Método**	**Valencia**	**Madrid**	**Andalucía**	**Murcia**	**Media**
PIB histórico	Baseline trimestral	1,1945	1,2827	1,4765	1,2712	1,3062
	Baseline estacional	**0,1412**	**0,1623**	**0,1758**	**0,2195**	**0,1747**
	Regresión lineal	0,6757	0,6114	0,6487	0,5755	0,6278
Tweets	AutoEncoder (AE)	0,4131	0,5585	0,5841	0,5162	0,5180
	AE + Filter	0,3819	0.5337	0,5616	0,4825	0,4899
	AE + Filter + RFr	0,2726	0,3668	0,4784	0,3524	0,3676
	AE + Filter + ϵ-SVR	**0,2511**	**0,3439**	**0,3721**	**0,2943**	**0,3154**

Analizaremos ahora la combinación de los resultados anteriores (obtenidos utilizando solo tweets) con la última parte del método propuesto: la red multimodal. Esto se hará evaluando adicionalmente la topología de esta red variando el número de capas y su tipo (consideraremos los mismos tipos que en el caso del autoencoder: Conv1D, LSTM y BiLSTM; consulta la Tabla 5.7 para ver los detalles de las redes evaluadas).

La Tabla 5.9 muestra los resultados de este experimento. El baseline estacional descrito anteriormente también se incluye en la comparación a modo de referencia. En primer lugar destacar que las diferentes combinaciones del método propuesto siempre obtienen un mejor resultado con respecto al baseline estacional. Con respecto a los diferentes tipos y número de capas consideradas, se observa que la mejor combinación se obtiene con 2 capas LSTM. En este caso, BiLSTM funciona un poco peor, probablemente debido a la naturaleza secuencial de los datos.

Tabla 5.9.: *Evaluación del resultado obtenido (en términos de MAE) con el método propuesto al agregar la última parte de la red: la red multimodal que combina el resultado de los tweets con el PIB histórico (ver Figura 5.1). Para esta segunda red, también se evalúan diferentes tipos de capas y número de capas. Los mejores resultados están marcados en negrita.*

		Comunidades autónomas				
Método	**# capas**	**Valencia**	**Madrid**	**Andalucía**	**Murcia**	**Media**
Baseline estacional	–	0,1369	0,1646	0,1706	0,2072	0,1698
CNN-1D	1	0,1158	0,1398	0,1526	0,1571	0,1413
	2	**0,1019**	**0,1252**	**0,1513**	**0,1458**	**0,1310**
	3	0,1040	0,1453	0,1559	0,1487	0,1384
LSTM	1	0,0781	0,1321	0,1346	0,1370	0,1204
	2	**0,0716**	**0,1103**	**0,1314**	**0,1333**	**0,1116**
	3	0,0799	0,1324	0,1368	0,1396	0,1221
BiLSTM	1	0,0963	0,1276	0,1535	0,1566	0,1335
	2	**0,0949**	**0,1226**	**0,1404**	**0,1371**	**0,1237**
	3	0,0952	0,1283	0,1516	0,1531	0,1320

Universitat d'Alacant
Universidad de Alicante

5.4.4. Evaluación final del modelo

Habiendo determinado la mejor configuración de los hiperparámetros y la mejor topología para las redes utilizadas en el método propuesto, procedemos a evaluar el resultado final obtenido y a compararlo con el de otros métodos del estado del arte.

En los experimentos anteriores, MSE y MAE se utilizaron como métricas de evaluación, ya que estábamos analizando los errores de reconstrucción y predicción y comparándolos con el valor del PIB real. Sin embargo, en Economía y Estadística, para analizar la evolución del PIB es más habitual utilizar las tasas de crecimiento intertrimestral e interanual. La tasa de crecimiento intertrimestral (comparando el trimestre actual con el anterior) no elimina la variación estacional del PIB (ciclos de subida y bajada de este valor que se alternan, ver Figura 4.3), lo que es fácilmente predecible. Por tanto, utilizaremos la tasa de crecimiento interanual (comparando el trimestre actual con el mismo trimestre del año anterior), que contiene el único componente verdaderamente impredecible del PIB.

La Tabla 5.10 muestra la tasa de error cometido al predecir el crecimiento interanual del PIB mediante el método propuesto. Este resultado se compara con los obtenidos en el estado del arte, incluyendo el baseline estacional y los métodos propuestos por Jaromír (2016) y Cicceri *et al.* (2020) (descritos en el capítulo 2). Es necesario entender que no todos los métodos del estado del arte para esta tarea son aplicables a la predicción del PIB en las regiones de España, bien porque utilizan valores solo disponibles anualmente, o porque utilizan indicadores no disponibles regionalmente o no disponibles en España.

En primer lugar destacar que nuestro enfoque duplica la precisión del baseline y el método propuesto por Jaromír (2016). Esto significa que, por ejemplo, al calcular el PIB de Madrid (que es superior a 50.000M de euros en los años incluidos en la comparativa), el error de previsión se reduce en una media de más de 621M de euros. Si comparamos este resultado con el obtenido por la aproximación propuesta por Cicceri *et al.* (2020), podemos ver que nuestro enfoque también obtiene una mayor precisión, aunque, en este caso, la mejora es menor. Nótese que la propuesta de Cicceri se basa en indicadores económicos

Tabla 5.10.: *Tasa de error al predecir el crecimiento interanual del PIB mediante el método propuesto en comparación con el baseline estacional y otros enfoques del estado del arte.*

	Comunidades autónomas				
Método	**Valencia**	**Madrid**	**Andalucía**	**Murcia**	**Media**
Baseline estacional	2,4584	2,4196	2,3639	2,5052	2,4368
Jaromír (2016)	2,0696	2,1502	2,1145	2,4037	2,1845
Cicceri *et al.* (2020)	1,7827	1,4205	1,5628	1,9024	1,6671
Método	1,0336	1,1587	1,1187	1,0715	1,0956

(como tasa de inflación, producción industrial, tasa de desempleo o deuda bruta) que se publican meses después del cierre del trimestre. Esto le permite realizar predicciones más informadas pero a cambio de meses de espera (lo que no ocurre con el método propuesto).

También cabe señalar que la aproximación multimodal se comporta igualmente bien en las cuatro regiones, mientras que los métodos comparados muestran una mayor variabilidad y, en particular, un peor comportamiento en el caso de Murcia. Hemos elegido cuatro regiones con diferentes estructuras económicas. Por ejemplo, los servicios financieros y del conocimiento son mucho más importantes en Madrid que en cualquier otra región, el turismo es particularmente importante en Valencia, etc. Pero Murcia tiene características mucho más distintivas: es una región pequeña con un PIB particularmente volátil y la participación de la agricultura en la economía regional es en general alta. Por un lado, se espera que todos los métodos tengan un peor desempeño ante una fuerte volatilidad, pero esto es aún peor para los métodos basados en NARX (como es el caso del método propuesto por Cicceri *et al.* (2020)), ya que es difícil separar las no linealidades de la variabilidad analizando solo datos históricos. Por otro lado, algunas de las variables económicas que utilizan los métodos comparados (por ejemplo, la producción industrial) pueden ser indicadores más pobres para la economía de Murcia. Estos métodos podrían mejorar su desempeño mediante el uso de otros indicadores específicos, pero eso requeriría un ajuste caso por caso. En cambio, el método propuesto explora las mismas fuentes de datos para todas las regiones, independientemente de la estructura económica subyacente.

Para analizar rigurosamente los resultados obtenidos vamos a realizar una comparación de la significancia estadística considerando el test no paramétrico de Wilcoxon a partir de muestras pareadas (Demsar, 2006; Li *et al.*, 2019). La idea es evaluar si la mejora observada en el método propuesto es estadísticamente significativa en comparación con los resultados obtenidos por los otros métodos. Los resultados obtenidos para las cuatro comunidades autónomas consideradas y para cada partición de la validación cruzada se compararon, por tanto, por pares, utilizando un umbral de significación estadística de $p < 0,05$.

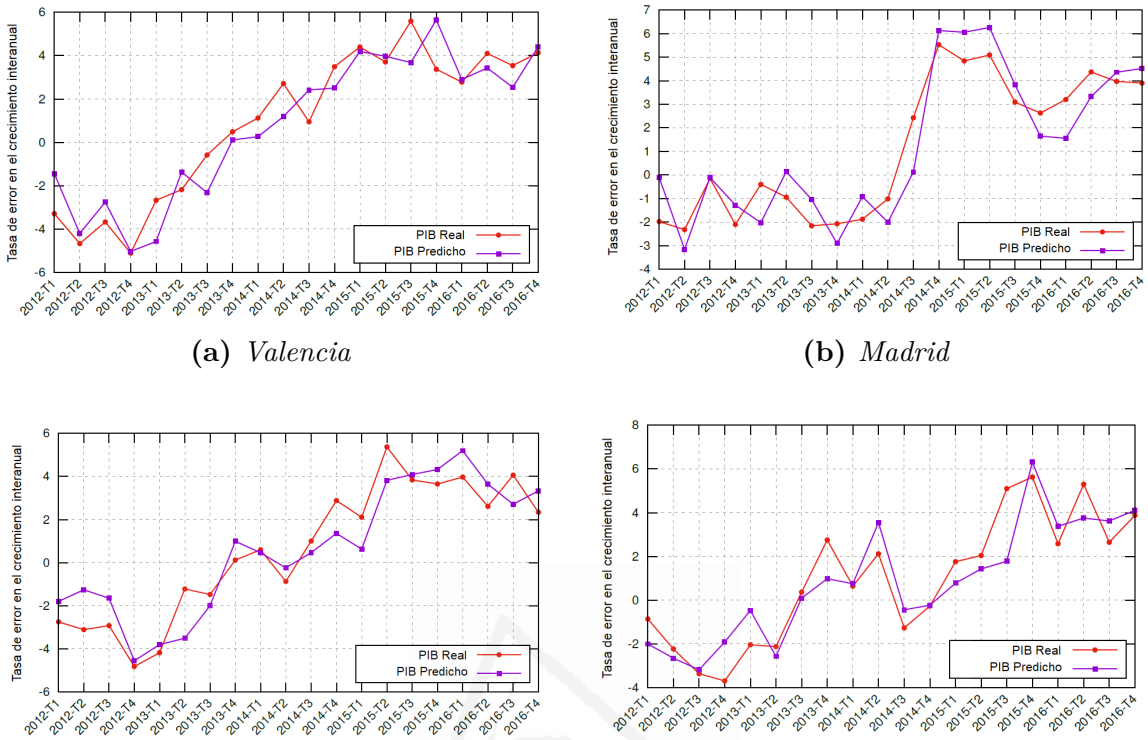
La Tabla 5.11 muestra los resultados de esta comparación, donde el símbolo ✓indica que el método en la fila mejora significativamente el método en la columna. Como se ve, el método propuesto (última fila) mejora significativamente a todas las soluciones comparadas del estado del arte. Por su parte, el baseline estacional y el método de Jaromír (2016) no mejoran a ninguno de los otros. Sin embargo, la propuesta de Cicceri *et al.* (2020), sí que consigue una mejora significativa con respecto a estos, pero no con respecto a la metodología presentada en este capítulo.

Tabla 5.11.: *Resultados del test no paramétrico de Wilcoxon a partir de muestras pareadas (Demsar, 2006). El símbolo ✓establece que es test se supera significativamente (es decir, el esquema de la fila es mejor que el esquema de la columna). La significancia se ha establecido en $p < 0{,}05$.*

	Test			
	Baseline estacional	Jaromír (2016)	Cicceri *et al.* (2020)	Método propuesto
Baseline estacional	–			
Jaromír (2016)		–		
Cicceri *et al.* (2020)	✓	✓	–	
Método propuesto	✓	✓	✓	–

En la Figura 5.5 se presenta la evolución secuencial de la tasa de crecimiento interanual del PIB para las cuatro comunidades autónomas consideradas, donde la línea roja representa el valor del PIB real y la línea azul la predicción realizada por el método propuesto. Al analizar estos resultados, se puede observar que la predicción sigue estrechamente la evolución del valor real del PIB a lo largo del tiempo. Incluso detecta los altibajos trimestrales y, especialmente, la tendencia general.

Universitat d'Alacant
Universidad de Alicante



(a) *Valencia* **(b)** *Madrid*

(c) *Andalucía* **(d)** *Murcia*

Figura 5.5.: *Comparación de la evolución del valor real del PIB (medido en tasa de crecimiento interanual) con la predicción realizada por el método propuesto para las cuatro comunidades autónomas consideradas.*

Universitat d'Alacant
Universidad de Alicante

5.4.5. Evaluación de la influencia de la entrada

Esta sección estudia el efecto que tanto el ruido en los atributos como la cantidad de información disponible tienen sobre la precisión del resultado obtenido.

Primero vamos a analizar la influencia de introducir ruido en los atributos en cada una de las entradas del método (tweets y secuencia con el historial del PIB). Realizamos esta evaluación siguiendo la propuesta de Zhu y Wu (2004), en el que el ruido en los atributos se genera sintéticamente para estudiar sus efectos en un entorno controlado. El proceso consiste en alterar un porcentaje de los atributos del conjunto de datos modificando su valor aleatoriamente dentro del rango de los posibles valores de ese atributo en el dominio del problema. El

proceso está guiado por el parámetro *índice de ruido del atributo*, que indica la probabilidad de que un atributo haya sido modificado.

La Figura 5.6 muestra el error cometido al predecir la tasa de crecimiento interanual del PIB mediante el método propuesto en presencia de ruido en los atributos (con tasas de 0 %, 10 %, 20 %, 30 % y 40 % de ruido, indicado en el eje horizontal de la gráfica) para cada una de sus entradas por separado. Como era de esperar, estos resultados empeoran en gran medida según aumenta el ruido. Sin embargo, tenga en cuenta que las tendencias no son similares, ya que el ruido introducido en la entrada del tweet afecta inicialmente mucho más el resultado, demostrando así la importancia de esta información. A partir del 20 %, el ruido parece afectar a ambas entradas de forma similar.

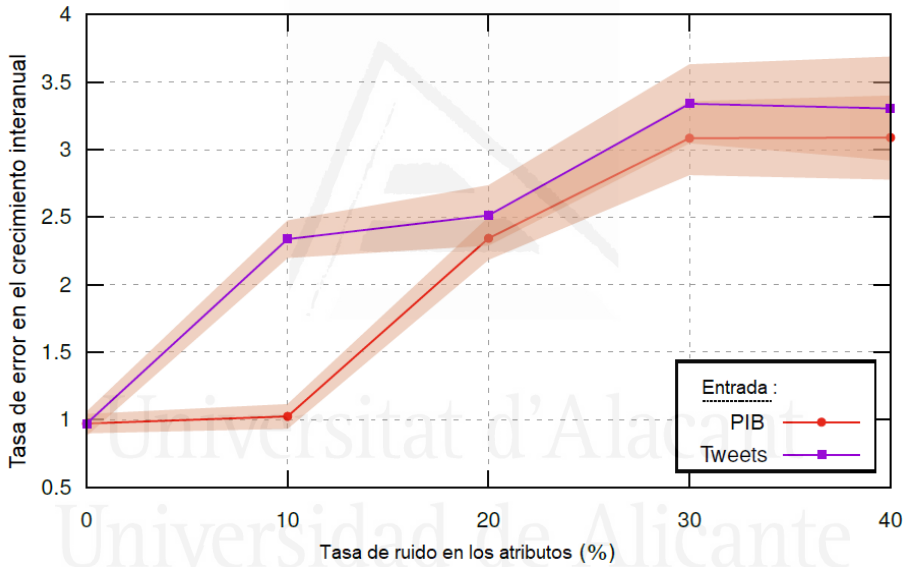


Figura 5.6.: *Resultado del método propuesto en presencia de diferentes niveles de ruido en los atributos (eje horizontal), introducido en cada una de las entradas por separado.*

Otro aspecto importante a valorar es la precisión del método propuesto en función de la semana del trimestre en que se utilice. Cada semana se escriben nuevos tweets que pueden influir en el pronóstico del PIB, y los comentarios escritos al final del trimestre pueden, en principio, proporcionar una opinión más informada de la tendencia del PIB. Nuestro conjunto de datos contiene 1.500 tweets para cada trimestre, es decir, aproximadamente 125 tweets por semana. Por tanto, evaluaremos los resultados obtenidos a medida que transcurran las

semanas del trimestre utilizando únicamente los tweets que se hayan redactado hasta esa semana.

Los resultados de este experimento se muestran en la Figura 5.7. Como puede verse, el error disminuye hacia el final del trimestre. También es posible observar que la desviación estándar se reduce, lo que podría indicar una mayor diversidad de opiniones al inicio y más consenso al final. Si comparamos estos resultados con el baseline estacional (línea horizontal), el método propuesto tiene una mayor precisión desde la primera semana. Cabe destacar también la ventaja de contar con esta predicción desde el inicio del trimestre, ya que, como se explicó anteriormente, las cifras oficiales se pueden obtener solo varios meses después de que finalice el trimestre.

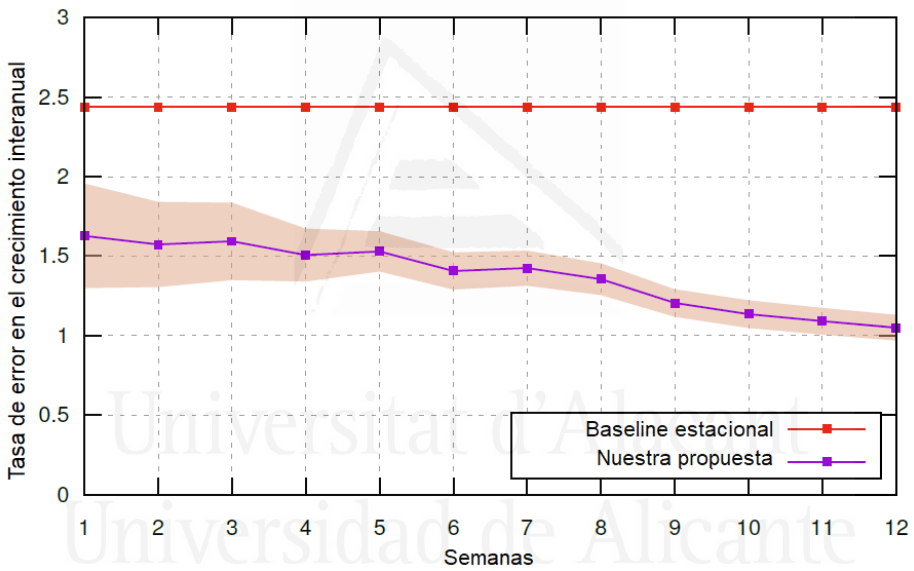


Figura 5.7.: *Tasa de error al predecir el crecimiento interanual del PIB mediante el método propuesto en función de la semana del trimestre en el que se realiza la previsión. El baseline estacional se incluye a modo de referencia.*

5.4.6. Análisis de los tweets

Otra ventaja adicional del método propuesto es la posibilidad de analizar los tweets utilizados, ya que permite identificar aquellos tweets que son más (y menos) influyentes para la predicción del valor del PIB. Un análisis del conjunto

de los tweets más y menos influyentes permite descubrir aquellos temas que son relevantes en la evolución económica de cada año.

Como ejemplo, la Tabla 5.12 muestra algunos de los tweets más (y menos) influyentes obtenidos para cada uno de los años estudiados. Al analizar los tweets más influyentes, podemos seguir la senda de la economía española desde los últimos años de profunda crisis económica hasta la recuperación. En 2012, estos tweets contenían términos negativos, como recesión, caída del PIB, caída de la inversión, gastos adicionales para pagar a proveedores, etc. Si bien las noticias negativas aún prevalecieron en 2013, aparecieron algunos temas positivos. Por un lado, los precios de la vivienda siguieron cayendo, mientras que la deuda de la administración aumentó. Por otro lado, la contracción del PIB no fue tan severa como se pronosticó. El comienzo de las buenas noticias macroeconómicas apareció en 2014: se esperaba que el PIB creciera, las deudas con los proveedores comenzaron a estabilizarse y la prima de riesgo disminuyó. Esta dinámica continuó en 2015, con un saldo global mayoritariamente positivo. Los tweets nos informan que las exportaciones iban bien y el desempleo estaba disminuyendo. Sin embargo, el gobierno regional siguió pidiendo préstamos y superó los objetivos de déficit. En 2016, la recuperación del PIB fue finalmente acompañada por la creación de empleo sustancial. Esto condujo a una disminución del desempleo de más de 500.000 personas en el conjunto de la región. Sin embargo, algunas zonas, y en especial la importante ciudad de Castellón (Comunidad Valenciana), continuaron con una tendencia negativa, perjudicando el saldo medio.

Universidad de Alicante

El conjunto de tweets con peor precisión de predicción (ver la segunda columna de la Tabla 5.12) se caracteriza por unas características comunes en todos los años estudiados. Primero, encontramos tweets que contienen información general o que analizan todo el país y no brindan información específica sobre la región; se trata, por ejemplo, de tweets emitidos por instituciones como el Banco Central Europeo o que discuten cuestiones macroeconómicas generales como los precios del petróleo. En segundo lugar, los tweets de políticos, que son bastante inespecíficos y, a menudo, sobre propuestas no cumplidas, también se encuentran entre los que no ayudan al método a extraer información relevante. Finalmente, también hemos notado que los tweets que comentan información

económica sobre trimestres o años anteriores (es decir, el crecimiento del PIB observado al final del año anterior) tampoco contribuyen a nuestra predicción.

5.4.7. Impacto económico del coronavirus

Finalmente, dado el impacto significativo que ha tenido la pandemia de COVID-19 en la economía global, provocando un colapso generalizado del PIB de todos los países, evaluaremos la capacidad del método propuesto para detectar este cambio drástico en la tendencia del PIB. Este es un caso particularmente interesante de analizar, ya que este cambio no puede detectarse utilizando solo valores históricos del PIB.

Para este experimento, creamos un nuevo conjunto de datos para los años 2012-2020, ya que el utilizado anteriormente (con las cifras regionales oficiales del PIB proporcionadas por IAES) está disponible solo hasta 2016 (consulte la sección 5.3.1). Para esto seguimos la misma metodología propuesta en Tituaña Castillo (2016) para calcular los valores del PIB regional. Como antes, separamos los datos en conjuntos de entrenamiento y prueba, pero en este caso, nuestro objetivo era evaluar los primeros trimestres de 2020.

La Tabla 5.13 muestra los resultados de este experimento, incluyendo el valor real de la tasa de crecimiento interanual del PIB y las predicciones realizadas tanto por el baseline estacional como por el método propuesto. Como puede observarse, el método propuesto consigue predecir este cambio de tendencia, con un pequeño error en el segundo trimestre. El baseline estacional predice un aumento del PIB, por lo que se comete un error muy alto en comparación con la tasa de crecimiento del PIB real. Esto se debe a que este método se basa únicamente en los valores históricos del PIB, que han experimentado una tendencia ascendente desde 2013.

El método propuesto consigue detectar la caída del PIB en el primer trimestre, aunque no la detecta de forma tan pronunciada. Esto puede deberse a que los tweets sobre la pandemia comenzaron en marzo, es decir, hacia el final del trimestre. Esta información negativa no tiene, por tanto, mucha influencia en la predicción. En el segundo trimestre sí que detecta una caída mucho más

Tabla 5.12.: *Algunos ejemplos de los tweets más y menos influyentes detectados por el método propuesto para calcular el valor del PIB para el período 2012-2016. También se incluye la cuenta que escribió cada tweet.*

Año	Tweets más influyentes	Tweets menos influyentes
2012	*@lasprovincias:* España, a un paso de la recesión tras caer su PIB un 0,3 %. *@elmundocv:* El precio medio de la vivienda nueva disminuyó un 3,2 %: De las tres capitales, Alicante lidera el descenso. *@epmediterraneo:* A. Fabra anuncia 420 millones del Gobierno para pagar a proveedores.	*@epmediterraneo:* Montoro anuncia que el próximo verano todas las administraciones públicas pagarán a proveedores a 30 días. *@epmediterraneo:* El BCE rechaza dar las garantías que pide Rajoy para el rescate. *@elperiodicoaqui:* CSIF considera escandalosamente inasumibles los más de 640.000 valencianos en paro según la EPA.
2013	*@valenciaplaza:* El PIB modera su caída hasta el 0,5 % en el primer trimestre. *@lasprovincias:* Sanidad estima entre 700 y 800 millones la deuda con proveedores. *@carlosnc78:* El precio de la vivienda nueva en la Comunitat registra una caída del 35,4 % desde su máximo.	*@generalitat:* El Presidente anuncia la creación de un plan de empleo con las diputaciones que creará 15.000 empleos. *@valenciaplaza:* El BCE mantiene los tipos en el mínimo histórico del 0,75 %. *@EconomistasOrg:* El BCE abre la puerta a retirar los billetes de 500 euros.
2014	*@GVAhisenda:* El Consell ahorrará 90,4 millones en 2014 por la bajada de la prima de riesgo. *@@popularescorts:* De Guindos anuncia que el PIB español crecerá un 1,5 % en 2014 y 2015 de media. *@DaValero* El BBVA duplica su previsión de crecimiento para la Comunitat y la cifra en el 0,4 % del PIB.	*@EconomistasOrg:* De Guindos avanza que el PIB creció un 0,3 % en el cuarto trimestre de 2013. *@PSOEValencia:* Proponemos Plan Estratégico de Empleo y Plan de Reforma Interior para Orriols y Torrefiel. *@csinformacion:* La Cámara de Comercio prevé que las exportaciones en el primer trimestre del 2014 mejoren.
2015	*@levante_emv:* La Generalitat pulveriza el tope de déficit en 7 meses y gasta 1.032 millones más de lo que ingresa. *@GVAeconomia:* Las exportaciones de la Comunidad Valenciana crecieron un 6 % entre enero y noviembre. *@eldiariocv:* El paro baja en la Comunidad Valenciana en julio con 7.743 desempleados menos.	*@valenciaplaza:* El BCE mantiene los tipos de interés en el mínimo histórico del 0,05 %. *@eldiariocv:* La corrupción y la manipulación del déficit en la Comunitat Valenciana llegan al Congreso. *@valenciaplaza* Montoro insiste en que las CCAA tendrán que cumplir el déficit y si no se exigirá con 'consecuencias'.
2016	*@carlosnc78:* La Comunidad Valenciana baja de los 500.000 parados por primera vez desde 2009, según la EPA. *@csinformacion:* Cevisama 2016 confirma la recuperación del sector cerámico, que deja atrás el miedo. *@copecastellon:* Castellón cuenta con 52.500 parados según los datos de la Encuesta de Población Activa del 3r trimestre.	*@JMaudos:* Previsiones poco optimistas del BdE. En 2018, España tendrá tasa de paro 17.3 % y déficit público del 2.9 %. *@valenciaplaza:* Puig pide otra reunión con Montoro por el FLA y la multa de Bruselas por el déficit. *@elmundocs:* CCOO ve un repunte del 1,8 % en el salario del sector azulejero en 2017.

pronunciada: 9,4 puntos desde el baseline. Esto representa una diferencia de 15 puntos con respecto al PIB real, sin embargo, es importante tener en cuenta que esta ha sido la mayor caída del PIB en cien años. El PIB normalmente tiende a variar un máximo de 1 punto de un trimestre a otro, pero el método propuesto consigue detectar una caída de más de 9 puntos.

Tabla 5.13.: *Tasa de crecimiento del PIB real para los primeros trimestres de 2020 en comparación con la predicción realizada por el baseline estacional y el método propuesto.*

Método	**2020-T1**	**2020-T2**
Tasa de crecimiento del PIB real	-2,17	-20,07
Predicción del baseline estacional	3,99	4,37
La predicción de nuestra propuesta	**3,36**	**-5,03**

Para completar este estudio vamos a analizar también los tweets utilizados para calcular el PIB en estos trimestres. La Tabla 5.14 muestra algunos ejemplos de los tweets más y menos influyentes detectados por el método propuesto.

Como se puede ver, en el primer trimestre, no muchos tweets comentan sobre el impacto económico de la enfermedad por coronavirus. Incluso encontramos algunos tweets relativamente positivos (por ejemplo, "el PIB per cápita ha aumentado en algunas ciudades"), y es por esta razón que el método aún no consigue detectar la caída tan pronunciada del PIB. En el segundo trimestre, encontramos un mayor número de tweets sobre el impacto negativo de la pandemia en la economía. Esto lleva al método a cambiar su tendencia con respecto al PIB histórico y predecir el colapso del PIB.

5.5. Conclusiones y trabajo futuro

En este capítulo se ha presentado un enfoque multimodal con el que predecir el PIB regional combinando valores históricos del PIB con la información almacenada en los mensajes de Twitter sobre el estado de la economía. El método propuesto se basa en una arquitectura de dos etapas. En la primera etapa, se utiliza un autoencoder multitarea para codificar tweets en una representación

Tabla 5.14.: *Algunos ejemplos de los tweets más y menos influyentes detectados por el método propuesto para calcular el valor del PIB para el primer y segundo trimestre de 2020.*

Trimestre	Tweets más influyentes	Tweets menos influyentes
2020-T1	*@alicanteplaza:* Ineca: el PIB per cápita de Alicante crece pero la provincia retrocede en el ranking nacional. *@radiovalencia:* el presidente de Camara-Valencia, José Vicente Morata pide ampliar mercados con motivo del coronavirus *@Economia_3:* Labora incentiva la contratación de menores de 30 años cualificados con subvenciones que cubren el 100 % del SMI.	*@radiocastellon:* PortCastelló gana competitividad durante 2019 a pesar de la rebaja de tasas. *@valenciaplaza:* Economía se alía con Hábitat Valencia y Home Textiles Premium para atraer visitantes internacionales a su próxima cita feria. *@levante_emv:* La justicia europea abre la puerta a anular las hipotecas vinculadas al IRPH si son abusivas.
2020-T2	*@valenciaplaza:* El PIB español caerá un 11,4 % en 2020 y hasta un 14,4 % si hay una nueva ola, según la OCDE. *@epmediterraneo:* La deuda pública se disparó en abril al récord de 1,22 billones, el 98,9 % del PIB. *@alicanteplaza:* Eurocaja Rural adelanta el pago de prestaciones por desempleo a sus clientes para paliar los efectos del coronavirus.	*@valenciaplaza:* Educación suma obras del Plan Edificant en 31 centros de 23 municipios con una inversión de 56,6 millones. *@lasprovincias:* El presidente de Mercadona considera que para avanzar en la recuperación es necesario «que la sociedad vea» que las empresas son «generadoras de riqueza y bienestar». *@Economia_3:* Comercio pide que se amplíe el límite de 400 metros para ciertas actividades que precisan de mayor superficie para poder exponer productos.

Universitat d'Alacant
Universidad de Alicante

relacionada con el PIB; luego se filtran para eliminar los valores atípicos y obtener la predicción sobre la base del consenso de opiniones. En una segunda etapa, el resultado obtenido anteriormente se combina con los valores históricos del PIB de la región utilizando una arquitectura de red multimodal.

La propuesta ha sido evaluada en cuatro comunidades autónomas españolas diferentes utilizando un conjunto de datos compuesto por los datos del PIB regionales proporcionados por el Instituto de Análisis Económico y Social y 120.000 tweets escritos por economistas, políticos, periódicos y las instituciones más relevantes de cada una de las regiones consideradas.

Se evaluaron varias topologías de red, métodos de *embedding* y valores de hiperparámetros con el fin de obtener la mejor configuración para el método propuesto. Los resultados muestran que esta aproximación aprende con éxito la tendencia del PIB utilizando solo tweets e información histórica, lo que significa que no es necesario esperar varios meses para obtener el PIB oficial. El método proporciona resultados significativamente mejores (analizado mediante test estadísticos) que los obtenidos por otros métodos del estado del arte, obteniendo una estimación más precisa incluso cuando el método se utiliza durante las primeras semanas del trimestre. Además, evaluamos los efectos que la pandemia de COVID-19 ha tenido en el PIB regional. Este experimento muestra la capacidad del método para predecir caídas significativas del PIB, junto con la importancia de la información multimodal, ya que esta predicción no es posible realizarla utilizando únicamente datos históricos.

Una ventaja adicional de la propuesta es la posibilidad de analizar el espacio latente aprendido por el autoencoder para identificar aquellos tweets que son más (o menos) influyentes en la predicción del PIB. El análisis de esta información permite detectar tanto las opiniones que alcanzan un mayor consenso como aquellas que se alejan de la opinión general, lo que nos permite descubrir los temas relevantes en la evolución económica de un período.

Como trabajo futuro, se propone mejorar la precisión del método multimodal combinándolo con variables económicas de diferentes niveles territoriales, como las explicadas en Cuevas *et al.* (2015) y que no se utilizan en el cálculo nacional del PIB (por ejemplo, el consumo de los hogares, pero en el periodo a predecir), o con los demás indicadores económicos utilizados en los métodos con los que hemos comparado nuestra propuesta. También se pretende extender la experimentación realizada a otras regiones de España.

Universitat d'Alacant
Universidad de Alicante

CAPÍTULO 6

Conclusiones

Finalizados los capítulos sobre el trabajo realizado, a continuación procedemos a revisar las conclusiones generales de esta disertación. El objetivo principal ha sido el estudio de los beneficios que la multimodalidad puede aportar a las técnicas de Machine Learning (ML) y Deep Learning (DL). Para ello, se han considerado dos casos de estudio, un trabajo inicial sobre la combinación de imagen y metadatos, y un segundo trabajo, como tema central de la tesis, para el estudio de la aplicación de la multimodalidad en la economía, en concreto, para la predicción del Producto Interior Bruto (PIB) a partir de mensajes escritos en Twitter y secuencias de valores históricos del PIB.

Universitat d'Alacant
Universidad de Alicante

6.1. Aplicación de la multimodalidad al reconocimiento de imágenes

En el capítulo 3, como experimentación inicial para comenzar a diseñar sistemas multimodales, se partió de un conjunto de datos disponible, como es el del sistema Mirbot, para el reconocimiento de objetos. El método combina las características visuales de las imágenes junto a una serie de metadatos para efectuar la clasificación. De esta tarea aprendimos la importancia de preprocesar los valores textuales para preservar su significado semántico a través de una codificación neuronal ofrecida por modelos tipo *word embedding* pre-entrenados. Estos modelos también demostraron obtener mejores resultados que otros tipos de codificaciones, como las basadas en diccionario o la codificación tipo one-hot.

En este trabajo estudiamos la técnica de fusión temprana mediante un enfoque agnóstico o independiente del modelo para la combinación multimodal de los datos de entrada. Si bien se compararon también otros enfoques, este fue el que mejores resultados obtuvo. Esto se debió principalmente a la naturaleza de los datos a procesar, una base de datos relativamente reducida y muy compleja, con multitud de clases diferentes y metadatos de muy distinta índole. Un enfoque independiente del modelo nos permitió preprocesar y filtrar dichos datos para prepararlos, y además aprovechar modelos pre-entrenados especializados en la clasificación de las distintas fuentes o modalidades de información por separado.

En este trabajo, las imágenes se clasificaron a tres niveles según la jerarquía de etiquetado utilizada: nivel raíz (con 5 categorías generales: animales, comida y bebida, objetos hechos por el hombre, objetos naturales y plantas), un segundo nivel (con 92 clases) y el nivel de hoja (con 1180 clases). Los resultados obtenidos empleando únicamente metadatos fueron prometedores para los niveles inferiores de la jerarquía (el nivel raíz y el segundo nivel, los más generarles), aunque bajos al nivel específico de clase (a pesar de mejorar en más de un veinte por ciento los resultados obtenidos utilizando la representación one-hot). Estos resultados ya ilustraban la poca información que los metadatos del dispositivo móvil aportaban para un etiquetado muy específico, debido a su contenido genérico. Por el contrario, la clasificación usando exclusivamente las imágenes reportó unos resultados mucho mejores a todos los niveles de etiquetado. Al combinar estas dos fuentes de información mediante la solución multimodal propuesta solo se consiguió mejorar los resultados en el nivel raíz de etiquetado, para el que, según la experimentación previa, los metadatos resultaban más informativos. Sin embargo, en los niveles de etiquetado más específicos, estos resultados empeoraron. Esto pudo deberse a la diferencia de información que aporta cada modalidad, ya que al mezclar información muy genérica con otra muy específica solo se consiguió confundir al clasificador.

Por tanto, de este trabajo se concluyó que, por un lado, cuando se disponen de pocos datos como para entrenar de forma efectiva una aproximación multimodal basada en el modelo, puede resultar adecuado el uso de metodologías multimodales independientes del modelo para aprovechar otras soluciones pre-entrenadas. Por otro lado, también se observó que es muy importante filtrar y preprocesar las características de las distintas modalidades y que la información que aporten

se pueda relacionar, ya que, características generales, poco informativas, en formatos no adecuados, o directamente no relacionadas, pueden entorpecer el correcto funcionamiento de la solución.

6.2. Estimación del PIB mediante información textual

A partir de los resultados y conclusiones obtenidas en la primera aproximación propuesta, en el capítulo 4 se realizó la versión inicial del método para la predicción del PIB. En este caso, el método se centró en la predicción del PIB regional y con una frecuencia trimestral para la Comunidad Valenciana utilizado solamente información de la red social Twitter. Se eligió el periodo de años 2012-2016 para crear el conjunto de datos de entrenamiento y evaluación. El uso de Twitter no se popularizó hasta mediados del año 2011, además, el contenido relacionado con una temática tan específica como es la económica no empezó a ser considerable hasta el año 2012. Para recopilar la base de datos utilizada se seleccionó además un conjunto de cuentas influyentes del ámbito económico y se filtraron los comentarios utilizando una serie de palabras clave. De esta manera, se consiguió que la información almacenada reflejase correctamente la situación económica de la región.

Para esta solución se propuso una arquitectura de red tipo autoencoder, pero modificada para que realizara la reconstrucción simultanea de los tweets y del valor del PIB, siguiendo para esto un enfoque multitarea. De esta forma, se forzó a la red a que condensase en su espacio latente las relaciones más significativas entre el indicador económico y la información extraída de los tweets. Para la construcción del autoencoder, se evaluaron distintas topologías, configuración de hiperparámetros así como cuatro metodologías para el cálculo de los *word embedding*. Los resultados preliminares obtenidos confirmaron que el método era capaz de realizar un seguimiento en tiempo real de la evolución económica, usando únicamente la información textual descargada de Twitter. Estos resultados demostraron que las redes sociales proporcionan casi en tiempo real un reflejo bastante cercano de las perspectivas económicas, permitiendo

además analizar estas opiniones para unidades geográficas donde las estadísticas oficiales no están disponibles.

6.3. Solución multimodal para la predicción del PIB

A partir de las conclusiones extraídas de la primera aproximación propuesta para la predicción del PIB, en el capítulo 5 se presentó una extensión multimodal, la cual, además de la información textual, también consideraba para su predicción secuencias con los valores históricos del PIB. Además se amplió el marco de estudio a un total de cuatro regiones españolas (Madrid, Andalucía y Murcia, además de la Comunidad Valenciana también considerada previamente). Se incluyó un paso adicional para el filtrado de los tweets utilizados para la predicción del PIB de cada trimestre. Este paso permitió la eliminación de los valores atípicos, que en este caso reflejaban las opiniones aisladas, alejadas del consenso general. Lo cual, además de reportar una mejora en los resultados obtenidos, también nos permitió analizar las opiniones más y menos influyentes consideradas en cada predicción.

A este método se le añadió una segunda fase para combinar el resultado obtenido a partir de los tweets con las secuencias de valores históricos del PIB. El objetivo es que esta arquitectura de red multimodal fuese capaz de capturar información complementaria (datos no visibles en cada modalidad de manera individual) o incluso filtrar ruido. Gracias a esto, la información histórica del PIB fue capaz de corregir el pronóstico para ajustarlo a la tendencia de los últimos trimestres y, a su vez, la información contenida en los tweets fue igualmente capaz de ajustar la tendencia al alza o a la baja si los comentarios pronosticaban una mayor subida o bajada.

Para la evaluación se incorporaron nuevos experimentos enfocados en analizar cada una de las etapas del método por separado, así como la influencia del ruido en las entradas y la evolución de la precisión a lo largo de las semanas, demostrando que el método es capaz de realizar predicciones con frecuencias menores a la trimestral (aunque con una precisión algo inferior). También se

realizó una comparación con otros trabajos del estado de arte, reportando mejores resultados en todos los casos validados mediante test estadísticos. Por último, también se probó en un entorno real marcado por la pandemia del COVID-19, obteniendo buenos resultados a pesar de representar la mayor caída del PIB en casi un siglo. Este experimento demostró la capacidad del método para predecir caídas significativas del PIB, junto con la importancia de la información multimodal, ya que esta predicción no es posible realizarla utilizando únicamente datos históricos.

6.4. Conclusiones generales

En el primer trabajo multimodal desarrollado para el sistema Mirbot, aprendimos que procesar la información por separado utilizando métodos agnósticos del modelo y su combinación temprana, tardía o híbrida puede tener varias ventajas. Una de estas ventajas es posibilitar el uso de otros modelos pre-entrenados especializados en el procesamiento de cada modalidad por separado, lo cual, cuando no se disponen de datos de entrenamiento suficientes, puede ser incluso necesario.

Sin embargo, estas aproximaciones también implican varios problemas: el uso de métodos unimodales que originalmente no fueron diseñados para la multimodalidad, y el ajuste de la contribución de cada modalidad, que no es una tarea fácil. Es por eso que muchos de estos métodos generalmente consideran solo la misma contribución. Además, en los enfoques agnósticos se corre el peligro de que, al preprocesar los datos o filtrarlos, se elimine información que quizás podría haberse aprovechado en la combinación.

Los enfoques basados en modelos abordan explícitamente el proceso de fusión en su construcción. Esto, por lo tanto, permite aprender funciones de fusión más complejas, por ejemplo, ponderando la contribución de las modalidades a partir de sus características. Estos enfoques multimodales buscan mejorar el resultado final al permitir que el método detecte información que no es visible en cada modalidad individual o incluso que pueda filtrar ruido o corregir valores erróneos de una modalidad aprovechando la información de la otra.

Los resultados de los trabajos expuestos han demostrado los beneficios de la multimodalidad. Estos beneficios han permitido la construcción de un sistema para la predicción del PIB regional en frecuencias temporales menores a un año. Esta propuesta, como ya se ha detallado a lo largo de esta tesis, supone una gran ventaja para el análisis financiero, empresarial y político, ya que permite conocer la situación económica de zonas geográficas que no son consideradas de manera oficial. Es de especial interés incidir también en el hecho de disponer de este dato al momento, al contrario de lo que sucede con los métodos tradicionales, los cuales dependen de la publicación de indicadores económicos que solo se encuentran disponibles varios meses después de finalizar el periodo a analizar.

Podemos afirmar que los objetivos propuestos en la introducción de esta tesis han sido cumplidos. Las contribuciones de la disertación consisten pues, en primer lugar, en un estudio de diferentes técnicas de ML y DL combinadas con distintas aproximaciones para integrar la multimodalidad. Además, se han utilizado como fuente de información datos de muy distinta índole, como imágenes, texto, valores numéricos y secuencias de valores. Pero la principal contribución de esta tesis consiste en haber sido capaces de construir un método multimodal para la predicción del PIB de forma regional y en periodicidades menores a las oficiales. Se ha demostrado además que el uso de redes sociales como reflejo de las dinámicas económicas a través de la actividad de sus usuarios es una estrategia acertada en tareas del ámbito macroeconómico.

Universitat d'Alacant
Universidad de Alicante

6.5. Trabajo futuro

Como trabajo futuro, y con el fin de mejorar los resultados del método propuesto, se pretende añadir una serie de indicadores económicos (como por ejemplo el consumo de los hogares) recopilados de las zonas geográficas a predecir. Es de esperar que el empleo de más información sobre las regiones contribuya a mejorar la predicción final. Además, también queremos extender los experimentos a otras regiones de España. De forma complementaria, se plantea el uso de nuevas fuentes de información textual, como otras redes sociales (por ejemplo, Facebook) o noticias de periódicos digitales o foros especializados en economía o finanzas.

CAPÍTULO 7

Bibliografía

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., y Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., y Zheng, X. (2016). Tensorflow: A system for large-scale machine learning. En *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 265–283.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185.

Andrew, G., Arora, R., Bilmes, J., y Livescu, K. (2013). Deep canonical correlation analysis. En Dasgupta, S. y McAllester, D., editores, *Proceedings of the 30th International Conference on Machine Learning*, volumen 28 de *Proceedings of Machine Learning Research*, pp. 1247–1255, Atlanta, Georgia, USA. PMLR.

Atrey, P. K., Hossain, M. A., El Saddik, A., y Kankanhalli, M. S. (2010). Multimodal fusion for multimedia analysis: a survey. *Multimedia Systems*, 16(6):345–379.

Universitat d'Alacant
Universidad de Alicante

Awad, M. y Khanna, R. (2015). *Support Vector Regression*, pp. 67–80. Apress, Berkeley, CA.

Babenko, A., Slesarev, A., Chigorin, A., y Lempitsky, V. (2014). Neural codes for image retrieval. En *European conference on computer vision*, pp. 584–599. Springer.

Bahrick, L. E. (1983). Infants' perception of substance and temporal synchrony in multimodal events. *Infant Behavior and Development*, 6(4):429–451.

Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. En *JMLR W&CP*, volumen 27, pp. 37–49.

Baltrušaitis, T., Ahuja, C., y Morency, L. (2019). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443.

Bank, D., Koenigstein, N., y Giryes, R. (2020). Autoencoders.

Barrett, P., Hunter, J., Miller, J. T., Hsu, J. C., y Greenfield, P. (2005). matplotlib – A Portable Python Plotting Package. En Shopbell, P., Britton, M., y Ebert, R., editores, *Astronomical Data Analysis Software and Systems XIV*, volumen 347 de *Astronomical Society of the Pacific Conference Series*, p. 91.

Bengio, S. (2004). Multimodal speech processing using asynchronous hidden markov models. *Information Fusion*.

Bengio, Y., Ducharme, R., Vincent, P., y Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Bengio, Y., Lamblin, P., Popovici, D., y Larochelle, H. (2006). Greedy layer-wise training of deep networks. En *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, p. 153–160, Cambridge, MA, USA. MIT Press.

Billings, S. A. (2013). *Nonlinear system identification NARMAX methods in the time, frequency, and spatio-temporal domains*. Wiley.

Bokányi, E., Lábszki, Z., y Vattay, G. (2017). Prediction of employment and unemployment rates from twitter daily rhythms in the us. *EPJ Data Science*, 6(1).

Universitat d'Alacant
Universidad de Alicante

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. En *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer.

Boutell, M. y Luo, J. (2005). Beyond pixels: Exploiting camera metadata for photo classification. *Pattern Recognition*, 38(6):935–946.

Bre, F., Gimenez, J., y Fachinotti, V. (2017). Prediction of wind pressure coefficients on building surfaces using artificial neural networks. *Energy and Buildings*, 158.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., y Sander, J. (2000). Lof: Identifying density-based local outliers. *SIGMOD Rec.*, 29(2):93–104.

Bullock, J., Luccioni, A., Hoffman Pham, K., Sin Nga Lam, C., y Luengo-Oroz, M. (2020). Mapping the landscape of artificial intelligence applications against covid-19. *Journal of Artificial Intelligence Research*, 69:807–845.

Burnham, K. P. y Anderson, D. R. (2002). *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer New York, New York, NY.

Cambouropoulos, E. (2010). The musical surface: Challenging basic assumptions. *Musicae Scientiae*, 14:131–147.

Carrillo-Larco, R. y Castillo-Cara, M. (2020). Using country-level variables to classify countries according to the number of confirmed covid-19 cases: An unsupervised machine learning approach. *Wellcome Open Research*, 5(56).

Chang, C.-C. y Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3).

Chatfield, K., Simonyan, K., Vedaldi, A., y Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. *CoRR*, abs/1405.3531.

Chaudhary, V., Tang, Y., Guzmán, F., Schwenk, H., y Koehn, P. (2019). Low-resource corpus filtering using multilingual sentence embeddings. *arXiv preprint arXiv:1906.08885*.

Universitat d'Alacant
Universidad de Alicante

Cheng, C., Tan, F., Hou, X., y Wei, Z. (2019). Success prediction on crowdfunding with multimodal deep learning. En *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 2158–2164. International Joint Conferences on Artificial Intelligence Organization.

Chollet, F. (2016). Xception: Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357.

Chung, J., Gulcehre, C., Cho, K., y Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling.

Cicceri, G., Inserra, G., y Limosani, M. (2020). A machine learning approach to forecast economic recessions—an italian case study. *Mathematics*, 8(2).

Colaboratory (2021). Google Colaboratory. https://colab.research.google.com/github/tensorflow/examples/blob/master/courses/udacity_intro_to_tensorflow_for_deep_learning/l01c01_introduction_to_colab_and_python.ipynb. Accessed: 15/02/2021.

Cuevas, A., Quilis, E. M., Frutos, R., Perez-Quiros, G., y Marin, J. (2015). Quarterly regional gdp flash estimates for the spanish economy (metcap model). Working paper, Independent Authority for Fiscal Responsibility, Spain.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

Dinakaran, B.; Annapurna, J. y Kumar, C. A. (2010). Interactive image retrieval using text and image content. *Cybernetics and Information Technologies*, 10(3):20–30.

D'mello, S. K. y Kory, J. (2015). A review and meta-analysis of multimodal affect detection systems. *ACM Comput. Surv.*, 47(3).

Do, T. H., Nguyen, D. M., Tsiligianni, E., Cornelis, B., y Deligiannis, N. (2017). Multiview deep learning for predicting twitter users' location.

Dubois, P. F. (2007). Guest editor's introduction: Python: Batteries included. *Computing in Science Engineering*, 9(3):7–9.

Universitat d'Alacant
Universidad de Alicante

EXIF (2016). Exchangeable image file format for digital still cameras : Exif Version 2.31. CIPA DC-008-Translation. http://www.cipa.jp/std/documents/e/DC-008-Translation-2016-E.pdf. Accessed: 14/6/2018.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., y Lin, C.-J. (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(61):1871–1874.

Fellbaum, C. (1998). WordNet: An Electronic Lexical Database.

Fels, S. S. y Hinton, G. E. (1993). Glove-talk: a neural network interface between a data-glove and a speech synthesizer. *IEEE Transactions on Neural Networks*, 4(1):2–8.

Feng, G., He, J., y Polson, N. G. (2018). Deep learning for predicting asset returns.

Fukui, A., Park, D. H., Yang, D., Rohrbach, A., Darrell, T., y Rohrbach, M. (2016). Multimodal compact bilinear pooling for visual question answering and visual grounding. En *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 457–468, Austin, Texas. Association for Computational Linguistics.

Gallego, A.-J., Calvo-Zaragoza, J., y Rico-Juan, J. R. (2020). Insights into efficient k-nearest neighbor classification with convolutional neural codes. *IEEE Access*, 8:99312–99326.

Gallego, A.-J., Pertusa, A., y Calvo-Zaragoza, J. (2018). Improving convolutional neural networks' accuracy in noisy environments using k-nearest neighbors. *Applied Sciences*, 8(11).

Gencoglu, O. (2018). Deep representation learning for clustering of health tweets.

Geonames (2018). Geonames feature codes. http://www.geonames.org/export/codes.html. Accessed: 14/6/2018.

GetOldTweets3 (2020). GetOldTweets3 library. https://pypi.org/project/GetOldTweets3/. Accessed: 20/12/2020.

Universitat d'Alacant
Universidad de Alicante

Gisgraphy (2018). Gisgraphy: Open source geocoder and addresses / POIs databases. http://www.gisgraphy.com/. Accessed: 14/6/2018.

Glorot, X., Bordes, A., y Bengio, Y. (2011). Deep Sparse Rectifier Neural Networks. *Journal of Machine Learning Research (JMLR) W&CP*, 15:315–323.

Gönen, M. y Alpaydin, E. (2011). Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12(64):2211–2268.

Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep Learning*. MIT Press.

Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., y Schmidhuber, J. (2017). Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232.

Gu, S., Kelly, B. T., y Xiu, D. (2019). Autoencoder asset pricing models. *Available at SSRN*.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., y Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Hansen, B. (2011). Threshold autoregression in economics. *Statistics and Its Interface*, 4:123–127.

Hardy, T. (2001). (ia: Inteligencia artificial). *POLIS, Revista Latinoamericana*.

Hassan, S.-U., Aljohani, N. R., Idrees, N., Sarwar, R., Nawaz, R., Martínez-Cámara, E., Ventura, S., y Herrera, F. (2020). Predicting literature's early impact with sentiment analysis in twitter. *Knowledge-Based Systems*, 192:105383.

He, K., Zhang, X., Ren, S., y Sun, J. (2015). Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*.

Hinton, G. E., Osindero, S., y Teh, Y.-W. (2006). A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., y Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors.

Universitat d'Alacant
Universidad de Alicante

Hinton, G. E. y Zemel, R. S. (1994). Autoencoders, minimum description length and helmholtz free energy. En *Advances in Neural Information Processing Systems*, pp. 3–10.

Hochreiter, S. y Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9:1735–80.

Hoffman-Plotkin, D. y Twentyman, C. T. (1984). A multimodal assessment of behavioral and cognitive deficits in abused and neglected preschoolers. *Child development*, pp. 794–802.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., y Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *CoRR*.

Huang, Z., Xu, W., y Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

ImageNet (2021). ImageNet: an image database organized according to the WordNet hierarchy. https://image-net.org/. Accessed: 13/07/2021.

Indaco, A. (2020). From twitter to gdp: Estimating economic activity from social media. *Regional Science and Urban Economics*, 85:103591.

Jaromír, V. (2016). Predicting future gdp development by means of artificial intelligence. *Littera Scripta*.

Jordan, M. I. (1986). Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986.

Joshi, S. y Deshpande, D. (2018). Twitter sentiment analysis system. *International Journal of Computer Applications*, 180(47):35–39.

Jozefowicz, R., Zaremba, W., y Sutskever, I. (2015). An empirical exploration of recurrent network architectures. En *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 2342–2350. JMLR.org.

Jung, J.-K., Patnam, M., y Ter-Martirosyan, A. (2018). An algorithmic crystal ball: Forecasts-based on machine learning.

Universitat d'Alacant
Universidad de Alicante

Junoh, M. Z. H. M. (2004). Predicting gdp growth in malaysia using knowledge-based economy indicators: a comparison between neural network and econometric approaches. *Sunway Academic Journal*, pp. 39–50.

Karpathy, A., Joulin, A., y Fei-Fei, L. (2014). Deep fragment embeddings for bidirectional image sentence mapping.

Keras (2021). Keras API reference. https://keras.io/api/. Accessed: 08/03/2021.

Kim, M. H., Kim, J. H., Lee, K., y Gim, G.-Y. (2021). The prediction of covid-19 using lstm algorithms. *International Journal of Networked and Distributed Computing*, 9:19–24.

Kingma, D. P. y Ba, J. (2015). Adam: A method for stochastic optimization. En Bengio, Y. y LeCun, Y., editores, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., e Inman, D. J. (2021). 1d convolutional neural networks and applications: A survey. *Mechanical Systems and Signal Processing*, 151:107398.

Krizhevsky, A., Sutskever, I., y Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. En *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, p. 1097–1105, Red Hook, NY, USA. Curran Associates Inc.

Krumhansl, C. L. (2001). *Cognitive foundations of musical pitch*. Oxford University Press.

Kumar, A. y Sebastian, T. (2012). Sentiment analysis on twitter. *International Journal of Computer Science Issues*, 9:372–378.

Lahat, D., Adali, T., y Jutten, C. (2015). Multimodal data fusion: An overview of methods, challenges, and prospects. *Proceedings of the IEEE*, 103(9):1449–1477.

Lan, Z.-Z., Bao, L., Yu, S.-I., Liu, W., y Hauptmann, A. G. (2014). Multimedia classification and event detection using double fusion. *Multimedia Tools Appl.*, 71(1):333–347.

Universitat d'Alacant
Universidad de Alicante

LASER (2020). Language-Agnostic SEntence Representations. https://github.com/facebookresearch/LASER. Accessed: 5/12/2020.

Lau, J. H. y Baldwin, T. (2016). An empirical evaluation of doc2vec with practical insights into document embedding generation. *CoRR*, abs/1607.05368.

Lazarus, A. A. (1976). *Multimodal behavior therapy / [edited by] Arnold A. Lazarus ; with eleven contributors*. Springer Pub. Co New York.

Le, Q. V. y Mikolov, T. (2014). Distributed representations of sentences and documents. *CoRR*, abs/1405.4053.

LeCun, Y., Bengio, Y., y Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., y Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551.

Lee, S. y Yoo, S. (2020). Multimodal deep learning for finance: integrating and forecasting international stock markets. *The Journal of Supercomputing*, 76:8294–8312.

Li, M.-W., Geng, J., Hong, W.-C., y Zhang, L.-D. (2019). Periodogram estimation based on lssvr-ccpso compensation for forecasting ship motion. *Nonlinear Dynamics*, 97(4):2579–2594.

Liaw, A. y Wiener, M. (2001). Classification and regression by randomforest. *Forest*, 23.

Liu, F. T., Ting, K. M., y Zhou, Z.-H. (2008). Isolation forest. En *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422.

Liu, W., Zheng, W.-L., y Lu, B.-L. (2016). Emotion recognition using multimodal deep learning. En Hirose, A., Ozawa, S., Doya, K., Ikeda, K., Lee, M., y Liu, D., editores, *Neural Information Processing*, pp. 521–529, Cham. Springer International Publishing.

Llorente, A., Garcia-Herranz, M., Cebrian, M., y Moro, E. (2015). Social media fingerprints of unemployment. *PLOS ONE*, 10(5):e0128692.

Universitat d'Alacant
Universidad de Alicante

L'Heureux, A., Grolinger, K., Elyamany, H. F., y Capretz, M. A. M. (2017). Machine learning with big data: Challenges and approaches. *IEEE Access*, 5:7776–7797.

Ma, L., Lu, Z., Shang, L., y Li, H. (2015). Multimodal Convolutional Neural Networks for Matching Image and Sentence. En *IEEE International Conference on Computer Vision (ICCV)*.

Mark, W. y Perrault, R. (2005). Calo: Cognitive assistant that learns and organizes.

Marsella, S., Xu, Y., Lhommet, M., Feng, A., Scherer, S., y Shapiro, A. (2013). Virtual character performance from speech. En *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '13, p. 25–35, New York, NY, USA. Association for Computing Machinery.

McCarthy, J., Minsky, M. L., Rochester, N., y Shannon, C. E. (2006). A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI Magazine*, 27(4):12.

McCulloch, W. S. y Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., y Dean, J. (2013). Distributed representations of words and phrases and their compositionality.

Millman, K. J. y Aivazis, M. (2011). Python for scientists and engineers. *Comput. Sci. Eng.*, 13:9–12.

Mills, T. y Mills, T. (1990). *Time Series Techniques for Economists*. Cambridge University Press.

Moghar, A. y Hamiche, M. (2020). Stock market prediction using lstm recurrent neural network. *Procedia Computer Science*, 170:1168–1173. The 11th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops.

Morency, L.-P., Quattoni, A., y Darrell, T. (2007). Latent-dynamic discriminative models for continuous gesture recognition. *Proceedings / CVPR,*

Universitat d'Alacant
Universidad de Alicante

IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition.

Mullainathan, S. y Spiess, J. (2017). Machine learning: An applied econometric approach. *Journal of Economic Perspectives*, 31(2):87–106.

Mulligan, R. M. y Shaw, M. L. (1980). Multimodal signal detection: Independent decisions vs. integration. *Perception & Psychophysics*, 28(5):471–478.

Nelson, D. M. Q., Pereira, A. C. M., y de Oliveira, R. A. (2017). Stock market's price movement prediction with lstm neural networks. En *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1419–1426.

Neverova, N., Wolf, C., Taylor, G., y Nebout, F. (2014). Moddrop: Adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., y Ng, A. Y. (2011). Multimodal deep learning. En *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, p. 689–696, Madison, WI, USA. Omnipress.

Nie, F., Huang, H., Cai, X., y Ding, C. (2010). Efficient and robust feature selection via joint l2,1-norms minimization. En Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., y Culotta, A., editores, *Advances in Neural Information Processing Systems*, volumen 23. Curran Associates, Inc.

Ochoa, S. F., Pino, J. A., Guerrero, L. A., y Collazos, C. A. (2006). Ssp: A simple software process for small-size software development projects. En Ochoa, S. F. y Roman, G.-C., editores, *Advanced Software Engineering: Expanding the Frontiers of Software Technology*, pp. 94–107, Boston, MA. Springer US.

Ortega-Bastida, J., Gallego, A.-J., Rico-Juan, J. R., y Albarrán, P. (2020). Regional gross domestic product prediction using twitter deep learning representations. En *IADIS International Conference Applied Computing 2020*, pp. 89–96.

Pagolu, V. S., Challa, K. N. R., Panda, G., y Majhi, B. (2016). Sentiment analysis of twitter data for predicting stock market movements.

Park, G. e Im, W. (2016). Image-text multi-modal representation learning by adversarial backpropagation. *CoRR*, abs/1612.08354.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., y Édouard Duchesnay (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Pertusa, A. y Gallego, A. J. (2018). Mirbot: A Multimodal Interactive Image Information Retrieval project. http://www.mirbot.com. Accessed: 14/6/2018.

Pertusa, A., Gallego, A.-J., y Bernabeu, M. (2013). MirBot: A Multimodal Interactive Image Retrieval System. En *Pattern Recognition and Image Analysis*, volumen 7887 de *Lecture Notes in Computer Science. 6th Iberian Conference, IbPRIA*, pp. 197–204.

Pertusa, A., Gallego, A.-J., y Bernabeu, M. (2018). MirBot: A collaborative object recognition system for smartphones using convolutional neural networks. *Neurocomputing*, 293:87 – 99.

Petajan, E. D. (1984). *Automatic Lipreading to Enhance Speech Recognition (Speech Reading)*. Tesis doctoral, University of Illinois at Urbana-Champaign, USA. AAI8502266.

Plank, B., Søgaard, A., y Goldberg, Y. (2016). Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. En *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 412–418, Berlin, Germany. Association for Computational Linguistics.

Potamianos, G., Neti, C., Gravier, G., Garg, A., y Senior, A. W. (2003). Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326.

Python (2020). Python programming language. https://es.python.org/. Accessed: 20/12/2020.

Universitat d'Alacant
Universidad de Alicante

Raina, R., Madhavan, A., y Ng, A. Y. (2009). Large-scale deep unsupervised learning using graphics processors. En *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, p. 873–880, New York, NY, USA. Association for Computing Machinery.

Rajendran, J., Khapra, M. M., Chandar, S., y Ravindran, B. (2016). Bridge correlational neural networks for multilingual multimodal representation learning. En *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 171–181, San Diego, California. Association for Computational Linguistics.

Ranzato, M. A., Boureau, Y.-L., y LeCun, Y. (2007). Sparse feature learning for deep belief networks. En *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, p. 1185–1192, Red Hook, NY, USA. Curran Associates Inc.

Razavian, A. S., Azizpour, H., Sullivan, J., y Carlsson, S. (2014). CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. En *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, CVPRW '14, pp. 512–519, Washington, DC, USA. IEEE Computer Society.

Řehůřek, R. y Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. En *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45–50, Valletta, Malta. ELRA.

Reimers, N. y Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. En *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Reimers, N. y Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813*.

Richardson, A., van Florenstein Mulder, T., y Vehbi, T. (2018). Nowcasting new

Universitat d'Alacant
Universidad de Alicante

zealand gdp using machine learning algorithms. CAMA Working Papers 2018-47, Centre for Applied Macroeconomic Analysis, The Australian National University.

Roondiwala, M., Patel, H., y Varma, S. (2017). Predicting stock prices using lstm. *International Journal of Science and Research (IJSR)*, 6.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408.

Rousseeuw, P. J. y Driessen, K. V. (1999). A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223.

Rumelhart, D., Hinton, G., y Williams, R. (1985). *Learning Internal Representations by Error Propagation.* ICS report. Institute for Cognitive Science, University of California, San Diego.

Sardelich, M. y Manandhar, S. (2018). Multimodal deep learning for short-term stock volatility prediction. *arXiv preprint arXiv:1812.10479.*

Sawhney, R., Mathur, P., Mangal, A., Khanna, P., Shah, R. R., y Zimmermann, R. (2020). Multimodal multi-task financial risk forecasting. En *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, p. 456–465, New York, NY, USA. Association for Computing Machinery.

Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A., y Williamson, R. (2001). Estimating support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E., Menon, V., y Soman, K. P. (2017). Stock price prediction using lstm, rnn and cnn-sliding window model. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1643–1647.

Sezer, O. B. y Ozbayoglu, A. M. (2018). Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Applied Soft Computing*, 70:525–538.

Shi, X., Chen, Z., Wang, H., Yeung, D., Wong, W., y Woo, W. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. En *NIPS*, pp. 802–810.

Universitat d'Alacant
Universidad de Alicante

Shynkevich, Y., McGinnity, T., Coleman, S. A., Belatreche, A., y Li, Y. (2017). Forecasting price movements using technical indicators: Investigating the impact of varying input window length. *Neurocomputing*, 264:71–88. Machine learning in finance.

spaCy (2020). Industrial-strength NLP. Pretrained statistical models for Spanish. https://spacy.io/models/es. Accessed: 5/12/2020.

Srivastava, N. y Salakhutdinov, R. R. (2012). Multimodal learning with deep boltzmann machines. En Pereira, F., Burges, C. J. C., Bottou, L., y Weinberger, K. Q., editores, *Advances in Neural Information Processing Systems*, volumen 25. Curran Associates, Inc.

Stock, J. y Watson, M. (2016). Chapter 8 - dynamic factor models, factor-augmented vector autoregressions, and structural vector autoregressions in macroeconomics. En Taylor, J. B. y Uhlig, H., editores, *Handbook of Macroeconomics*, volumen 2 de *Handbook of Macroeconomics*, pp. 415–525. Elsevier.

Su, C.-J. y Li, Y. (2019). Recurrent neural network based real-time failure detection of storage devices. Technical report, Microsystem Technologies.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., y Rabinovich, A. (2014). Going deeper with convolutions.

TensorFlow (2021). TensorFlow library. https://www.tensorflow.org/?hl=es-419. Accessed: 08/03/2021.

TensorFlow Core (2021). Keras — TensorFlow Core. https://www.tensorflow.org/guide/keras?hl=es. Accessed: 08/03/2021.

TensorFlowHub (2020). Token based text embedding trained on spanish google news 50b corpus. https://tfhub.dev/google/tf2-preview/nnlm-es-dim50-with-normalization/1. Accessed: 5/12/2020.

Tituaña Castillo, M. (2016). Contabilidad Regional Trimestral de España. Instituto Universitario de Análisis Económico y Social (IAES). http://www.iaes.es/contabilidadregionaltrimestral.html. Accessed: 30/11/2020.

Trask, A., Michalak, P., y Liu, J. (2015). sense2vec - a fast and accurate method for word sense disambiguation in neural word embeddings.

Universitat d'Alacant
Universidad de Alicante

Turing, A. M. (1950). I.—Computing Machinery and Intelligence. *Mind*, LIX(236):433–460.

UKPLab (2020). Sentence Transformers: Multilingual Sentence Embeddings using BERT, RoBERTa, XLM-RoBERTa & Co. with PyTorch. https://github.com/UKPLab/sentence-transformers. Accessed: 5/12/2020.

van der Maaten, Laurens; Hinton, G. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9(nov):2579–2605.

Vapnik, V. (1998). *Statistical learning theory*. John Wiley and Sons.

Vidal, A. H. T. y Casacuberta, F. (2011). *Multimodal Interactive Pattern Recognition and Applications*. Springer-Verlag London.

Waldrop, M. M. (2019). News feature: What are the limits of deep learning? *Proceedings of the National Academy of Sciences of the United States of America*, 116(4):1074–1077.

Waliszewski, K. y Zieba-Szklarska, M. (2020). Robo-advisors as automated personal financial planners – swot analysis. *Finanse i Prawo Finansowe*, 3:155–173.

Wang, W., Arora, R., Livescu, K., y Bilmes, J. (2015). On deep multi-view representation learning. En *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 1083–1092. JMLR.org.

Wehrmann, J., Becker, W., Cagnini, H., y Barros, R. (2017). A character-based convolutional neural network for language-agnostic twitter sentiment analysis. En *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2384–2391.

Werbos, P. (1975). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Harvard University.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

Wikipedia Commons (2015). Typical CNN architecture. https://commons.wikimedia.org/wiki/File:Typical_cnn.png. Accessed: 19/07/2021.

Williams, T., Colin, K., Bröker, H.-B., Campbell, J., Cunningham, R., Denholm, D., Elber, G., Fearick, R., Grammes, C., Hart, L., y otros (2020). *Gnuplot 5.4. An Interactive Plotting Program.* gnuplot, gnuplot-info@lists.sourceforge.net.

Wu, Z., Cai, L., y Meng, H. (2005). Multi-level fusion of audio and visual features for speaker identification. En Zhang, D. y Jain, A. K., editores, *Advances in Biometrics*, pp. 493–499, Berlin, Heidelberg. Springer Berlin Heidelberg.

Xing, F. Z., Cambria, E., Malandri, L., y Vercellis, C. (2019). Discovering bayesian market views for intelligent asset allocation. En Brefeld, U., Curry, E., Daly, E., MacNamee, B., Marascu, A., Pinelli, F., Berlingerio, M., y Hurley, N., editores, *Machine Learning and Knowledge Discovery in Databases*, pp. 120–135, Cham. Springer International Publishing.

Xing, F. Z., Cambria, E., y Welsch, R. E. (2018). Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1):49–73.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., y Bengio, Y. (2016a). Show, attend and tell: Neural image caption generation with visual attention.

Xu, T., Zhang, H., Huang, X., Zhang, S., y Metaxas, D. N. (2016b). Multimodal deep learning for cervical dysplasia diagnosis. En Ourselin, S., Joskowicz, L., Sabuncu, M. R., Unal, G., y Wells, W., editores, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pp. 115–123, Cham. Springer International Publishing.

Yao Wang, Zhu Liu, y Jin-Cheng Huang (2000). Multimedia content analysis-using both audio and visual clues. *IEEE Signal Processing Magazine*, 17(6):12–36.

Yu, P. y Yan, X. (2020). Stock price prediction based on deep neural networks. *Neural Computing and Applications*, 32(6):1609–1628.

Zhang, Y. y Yang, Q. (2017). A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*.

Universitat d'Alacant
Universidad de Alicante

Zhao, R., Deng, Y., Dredze, M., Verma, A., Rosenberg, D. S., y Stent, A. (2018). Visual attention model for cross-sectional stock return prediction and end-to-end multimodal market representation learning. *CoRR*, abs/1809.03684.

Zhao, Y., Li, J., y Yu, L. (2017). A deep learning ensemble approach for crude oil price forecasting. *Energy Economics*, 66:9–16.

Zhu, X. y Wu, X. (2004). Class noise vs. attribute noise: A quantitative study. *Artificial Intelligence Review*, 22(3):177–210.

Universitat d'Alacant
Universidad de Alicante

APÉNDICE A

Lista de acrónimos

AE - Autoencoder.

AIReF - Autoridad Independiente de Responsabilidad Fiscal.

API - Application Programming Interfaces.

AR - Autoregresivo.

ARIMA - Autoregressive Integrated Moving Average model.

ARIMAX - Autoregressive Integrated Moving Average with exogenous variables model.

AVSR - Audio-Visual Speech Recognition.

BD - Base de Datos.

BERT - Bidirectional Encoder Representations from Transformers.

BiLSTM - Bidirectional Long Short Term Memory.

CALO - Cognitive Assistant that Learns and Organizes.

CCAA - Comunidad Autónoma.

CNN - Convolutional Neural Network.

Universitat d'Alacant
Universidad de Alicante

CNN-1D - Convolutional Neural Network One Dimension.

CPU - Central Processing Unit.

CV - Cross Validation.

DCCAE - Deep Canonically Correlated Autoencoders.

DL - Deep Learning.

DLSI - Departamento de Lenguajes y Sistemas Informáticos.

EXIF - Exchangeable Image File Format.

FC - Fully Connected.

GDP - Gross Domestic Product.

GPU - Graphics Processing Unit.

GRFIA - Grupo de Reconocimiento de Formas e Inteligencia Artificial.

GRU - Gated Recurrent Unit

HCI - Human-Computer Interaction.

IA - Inteligencia Artificial.

IAES - Instituto Universitario de Análisis Económico y Social.

ISO - International Organization for Standardization.

KNN - K-Nearest Neighbors.

LASER - Language-Agnostic SEntence Representations.

LSTM - Long Short Term Memory.

MAE - Mean Absolute Error.

Universitat d'Alacant
Universidad de Alicante

MKL - Multiple kernel learning.

ML - Machine Learning.

MLP - Multilayer Perceptron.

MSE - Mean Square Error.

NARX - Nonlinear Autoregressive with exogenous variables model.

NC - Neural Codes.

NLP - Natural Language Processing.

NN - Neural Network.

NNLM - Neural Network Language Models.

PIB - Producto Interior Bruto.

R-CNN - Region Based Convolutional Neural Networks.

RBF - Radial Basis Function networks.

ReLU - Rectified Linear Unit.

RF - Random Forest.

RFr - Random Forest Regressor.

RMSE - Root Mean Square Error.

RNN - Recurent Neural Network.

ROI - Region Of Interest.

SSP - Social Signal Processing.

SVM - Support Vector Machine.

Universitat d'Alacant
Universidad de Alicante

SVR - Support Vector Regression.

TAR - Threshold Autorregressive model.

TF-HUB - Tensor Flow Hub.

TPU - Tensor Processing Unit.

t-SNE - t-Distributed Stochastic Neighbor Embedding.

UKPLab - Ubiquitous Knowledge Processing Lab.

VAR - Vector Autorregressive model.

Universitat d'Alacant
Universidad de Alicante