UCAM

UNIVERSIDAD CATÓLICA DE MURCIA

ESCUELA INTERNACIONAL DE DOCTORADO

Programa de Doctorado en Tecnologías de la Computación e Ingeniería Ambiental

Construcción de un modelo eficiente de predicción de heladas en entornos locales mediante técnicas del análisis inteligente en contextos IoT

Autor:
D. Miguel Ángel Guillén Navarro

Directores:
Dra. Dña. Belén López Ayuso
Dra. Dña. Raquel Martínez España

Murcia, mayo de 2021

ESCUELA INTERNACIONAL DE DOCTORADO

Programa de Doctorado en Tecnologías de la Computación e Ingeniería Ambiental

Construcción de un modelo eficiente de predicción de heladas en entornos locales mediante técnicas del análisis inteligente en contextos IoT

Autor:
D. Miguel Ángel Guillén Navarro

Directores:
Dra. Dña. Belén López Ayuso
Dra. Dña. Raquel Martínez España

Murcia, mayo de 2021

EIDUCAM
Escuela Internacional
de Doctorado

UCAM
Universidad Católica
de Murcia

AUTORIZACIÓN DEL DIRECTOR DE LA TESIS
PARA SU PRESENTACIÓN

La Dra. Dña. Belén López Ayuso y la Dra. Dña. Raquel Martínez España como Directoras(1) de la Tesis Doctoral titulada "Construcción de un modelo eficiente de predicción de heladas en entornos locales mediante técnicas del análisis inteligente en contextos IoT" realizada por D. Miguel Ángel Guillén Navarro en el Programa de Doctorado Tecnologías de la Computación e Ingeniería Ambiental, **autoriza su presentación a trámite** dado que reúne las condiciones necesarias para su defensa.

Lo que firmo, para dar cumplimiento al Real Decreto 99/2011 de 28 de enero, en Murcia a 18 de mayo de 2021.

MARIA BELEN|LOPEZ|AYUSO
Firmado digitalmente por MARIA BELEN|LOPEZ|AYUSO
Fecha: 2021.06.23 16:07:29 +02'00'

MARTINEZ ESPAÑA RAQUEL - 77716639F
Firmado digitalmente por MARTINEZ ESPAÑA RAQUEL - 77716639F
Nombre de reconocimiento (DN): c=ES, serialNumber=IDCES-77716639F, givenName=RAQUEL, sn=MARTINEZ ESPAÑA, cn=MARTINEZ ESPAÑA RAQUEL - 77716639F
Fecha: 2021.06.23 11:07:56 +02'00'

(1) Si la Tesis está dirigida por más de un Director tienen que constar y firmar ambos.

A Elena, la mujer de mi vida

A nuestros hijos, por su amor y alegría

Agradecimientos

Es normal que los agradecimientos se escriban en último lugar. También lo es pensar que es una tarea sencilla, pero me doy cuenta de que no es así. No quiero dejar atrás a nadie y si así lo hago, pido disculpas por adelantado. Creo firmemente que cada persona con la que te cruzas en esta vida, para bien o para mal, siempre aporta algo.

Doy gracias a Dios por lo que permite en mi vida, “porque todas las cosas provienen de él, y son por él y para él"(Rm 11, 35).

Mi primer pensamiento es para mi mujer, Elena. Hace 20 años que nos conocemos y 14 que compartimos nuestras vidas. Nunca me ha objetado nada cuando le decía “me tengo que poner con la tesis”, todo lo contrario, ha sido mi gran apoyo. Sin ella no habría sido posible.

Junto a ella, un profundo agradecimiento a todos nuestros hijos. Cada minuto que le he dedicado en casa a la tesis ha sido tiempo que ellos me han dado a mi. Cada uno de vosotros es una bendición de Dios.

Muchas gracias a mis dos directoras de tesis. A Belén, que siempre tiene abiertas las puertas de su despacho para darte el consejo y dirección adecuados. Gracias por hacernos crecer personal y profesionalmente a todos los que formamos el claustro del Grado en Ingeniería Informática. A Raquel, que sin ella no habría sido posible realizar esta tesis. Trabajadora incansable. Ha sabido dirigir por el camino correcto y conseguir que llegue a buen puerto. Sé que ha sufrido más que yo cuando había algún contratiempo y que mi alegría de ahora es también la suya.

Llevo 14 años trabajando en la UCAM junto a personas que más que compañeros considero amigos. Luz, Onia, Jesús, Antonio, Paco, Andrés Bueno, Mada, Fernando, Antonio Serrano, Andrés Muñoz, Baldo, José Luis, Manuel, Pedro, Joaquín y Javier: no tengo palabras para agradeceros todo lo que hacéis por mi. Doy gracias a Dios por cada

uno de vosotros.

Un recuerdo muy especial también para Chema. Ya no está físicamente en nuestro departamento pero como si lo estuviera. Gracias por todo lo compartido estos años, gracias por tu ayuda para que esta tesis sea realidad y gracias por hacerme partícipe del proyecto "Planificación y gestión de recursos hídricos a partir de análisis de datos IoT (WATERoT, RTC-2017-6389-5)"del que esta tesis forma parte.

También un agradecimiento a todos los compañeros de la Dirección TIC. En especial a Samuel, por confiar en mi para gestionar a un equipo increíble de personas que hacen sencillo algo tan complicado como dirigir todos los proyectos informáticos de la Universidad.

Y me gustaría terminar estos agradecimientos acordándome de mi familia. Los que están, mi madre y mis hermanos, y los que se fueron, mi padre y mis abuelos. Gran parte de lo que soy se lo debo a ellos. Hemos pasado momentos difíciles y también otros entrañables. Espero que sigamos siempre unidos.

Índice

Índice de figuras

Capítulo 1

Introducción

1.1 La agricultura y la tecnología

El ser humano siempre ha buscado la forma de incorporar nuevas tecnologías para hacer más productivos sus cultivos. Desde el uso de arados tirados por bueyes hasta los más modernos tractores guiados por GPS, todo es una continua incorporación de nuevas técnicas. En este ámbito, la agricultura de precisión se encarga de proporcionar herramientas y técnicas para automatizar y mejorar las labores agrícolas [19], [11].

La producción agrícola depende de muchos factores. Algunos pueden ser controlados por los agricultores y otros no, como las heladas [8]. Es evidente que la helada no se puede evitar, pero existen técnicas para que no afecte al cultivo o al menos mitigue las pérdidas producidas [10], [6]. No obstante, estos sistemas suelen ser costosos en términos de consumo energético y agua y deben activarse sólo cuando se tenga la certeza de que se va a producir una helada.

El cambio climático no ha hecho más que agravar la situación. Durante el paso del invierno a la primavera se plantea días de calor antes de lo esperado y el árbol florece. El problema sucede cuando vuelven de manera repentina las temperaturas frías, se produce una helada y se pierde la cosecha. Si a esto se le añade que hay una mayor amplitud térmica, con días más cálidos y noches frías, la situación se complica [1], [13].

La Organización de Naciones Unidas (ONU) ha establecido diecisiete Objetivos de Desarrollo Sostenible (ODS) para conseguir un futuro sostenible para todos, entre los que se encuentran el objetivo 6 (Garantizar la disponibilidad de agua y su gestión sostenible y el saneamiento para todos) y el 7 (Garantizar el acceso a una energía asequible, segura, sostenible y moderna) [5]. Dentro de estos se encuentran a su vez la meta 6.4 (De aquí a 2030, aumentar considerablemente el uso eficiente de los recursos hídricos en todos los sectores y asegurar la sostenibilidad de la extracción y el abastecimiento de agua dulce para hacer frente a la escasez de agua y reducir considerablemente el número de personas que sufren falta de agua) y la 7.4 (De aquí a 2030, duplicar la tasa mundial de mejora de la eficiencia energética). Los resultados de esta tesis están alineados con dichas metas y con lo que el Papa Francisco propone en su encíclica Laudato Sí en cuanto al "cuidado de la casa común" [7].

Como se ha comentado anteriormente, existen diversas técnicas y sistemas antiheladas que protegen el cultivo. El problema es que su coste suele ser elevado, por lo que si se activan antes de tiempo, o incluso sin ser necesario, provoca cuantiosas pérdidas económicas.

En la actualidad existen organismos que a través de modelos principalmente físicos predicen con cierta exactitud la temperatura para una zona [4]. No obstante, es bien conocido por los agricultores que dicha previsión no se ajusta a lo que ocurre en sus parcelas. Es decir, aunque la predicción establezca que no va a helar en una región, la orografía del terreno, la distribución de los cultivos, los humedales u otros factores hacen que la helada pueda producirse.

La hipótesis de partida de esta tesis tiene su génesis en el conocimiento que tienen los agricultores de su explotación respecto a las heladas que se producen en las mismas.

A partir de este conocimiento se ha creado un modelo de predicción de heladas para la parcela. Una red de sensores recoge en tiempo real los valores de las variables meteorológicas utilizadas en el modelo, procesándolas para eliminar ruido y ejecutando un modelo que predice la temperatura. En base al resultado obtenido se activa, o no, el sistema antihelada.

La infraestructura se ha desplegado en dos parcelas de la Región de Murcia. Por su localización geográfica tienen un clima con una gran amplitud térmica, llegando incluso a alcanzar más de 20° de diferencia entre la noche y el día. Es por este motivo por el que se trata de un escenario óptimo para construir y probar el modelo que se describe en esta tesis.

En los siguientes subapartados se desarrollará una breve descripción sobre la infraestructura utilizada y el paradigma del análisis inteligente de datos; ambos, elementos necesarios para abordar la hipótesis planteada.

1.2 Internet de las Cosas

El Internet de las Cosas (IoT) se define como una disciplina capaz de crear sistemas que conectan objetos (cosas) entre sí, a otros servicios a través de Internet. En definitiva, obtiene datos y los transfiere sin la intervención de un humano. La recogida de información se realiza a través de los diferentes sensores que están conectados a los objetos. La transmisión se lleva a cabo utilizando nuevos protocolos de comunicación que permiten un bajo consumo energético y una alta capacidad de transmisión, aún en grandes distancias [17].

El IoT ha revolucionado muchos sectores, entre los que destaca el sector de la agricultura [15]. Mediante el uso de este tipo de infraestructuras, la agricultura ha conseguido alcanzar unos altos niveles tecnológicos, tanto de monitorización como de industrialización. Todo ello con una drástica reducción de costes y con un aumento de producción y calidad de los productos [14].

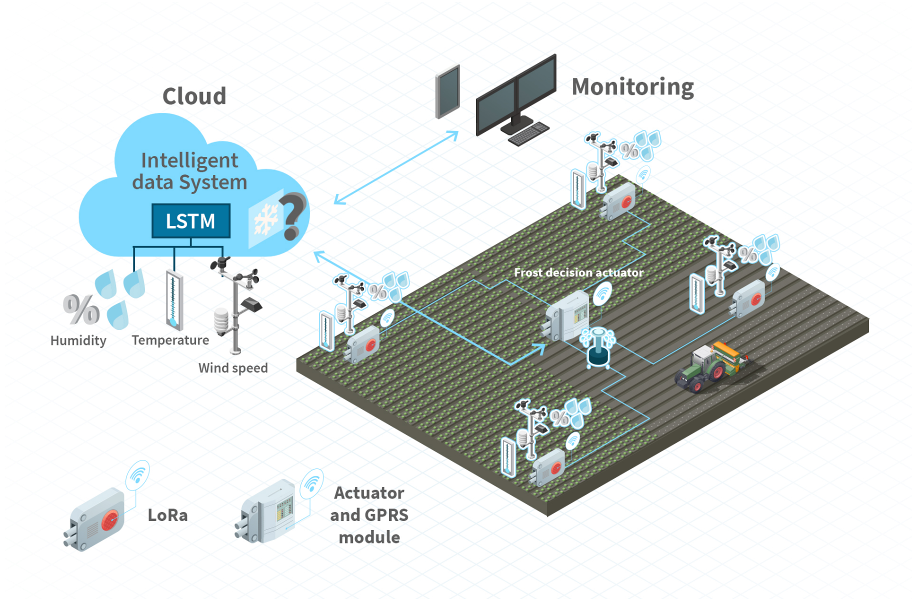


Figura 1.1: Arquitectura del sistema implementado

Uno de los elementos fundamentales de esta tesis, que se muestra en la Figura1.1, es la arquitectura diseñada para la recogida de datos. A través de dicha arquitectura se obtienen los datos de los sensores, se envían al nodo que hace de concentrador, se procesan para eliminar ruidos y finalmente se envían al servidor para que, en base a los resultados obtenidos en la ejecución del modelo, se proceda con la activación o no del sistema antihelada.

A la hora de implementar un sistema IoT es importante seleccionar la tecnología de comunicación adecuada al entorno donde se despliegue. Existen multitud de protocolos, cada uno con sus ventajas e inconvenientes. Estudiada la fisonomía de la parcela donde se ha desplegado la red se ha seleccionado LoRa [3]. Entre sus características principales destacan su bajo consumo, radio de alcance superior a un kilómetro y transferencia de datos mínima. No obstante, también se estudiaron alternativas como SigFox, descartada por su alto coste, y ZigBee [9]. Con esta última se diseñó e implementó un escenario piloto pero los resultados no fueron satisfactorios, al tratarse de una tecnología pensada para comunicaciones de corto alcance y principalmente en interiores [16].

Además de la decisión sobre la tecnología de comunicación, también hubo que evaluar diferentes soluciones del mercado que pudieran dar soporte a los nodos donde se colocan los sensores, se procesan los datos obtenidos y se envían al servidor. Inicialmente se implementaron sistemas de bajo coste que funcionaban correctamente en interior, pero no cuando eran desplegados en las parcelas. Por este motivo, al final se adquirió una solución propietaria que permitía la conexión de sensores a un nodo y este a su vez con otros mediante LoRa. Tal y como se muestra en la Figura 1.1, uno de los nodos hace de concentrador principal, por lo que recibe los datos del resto de nodos y los envía al servidor mediante mediante General Packet Radio Service (GPRS).

En cuanto a los sensores utilizados, destacar que todos los nodos tienen un sensor de temperatura y otro de humedad, variables que se han constatado como fundamentales a la hora de predecir la helada. Otras como la dirección y velocidad del viento también son recogidas, pero sólo en uno de los nodos. Ya que en ausencia de accidentes geográficos el valor es el mismo en toda la parcela.

De esta forma se ha construido una infraestructura que permite crear un modelo de predicción específico para la parcela, en base a los datos recabados.

1.3 El Análisis Inteligente de Datos

Uno de los elementos fundamentales de cara a alcanzar el objetivo principal de esta tesis es el Análisis Inteligente de los Datos (AID) [2]. Este paradigma nos ofrece un conjunto de tareas y técnicas que nos van a permitir obtener una inferencia en las parcelas para abordar si se producirá o no una helada. En cierto modo, y con un nivel de abstracción, se puede decir que el AID proporciona sentido a los datos coleccionados mediante el desarrollo de métodos y técnicas adecuados al campo donde se realiza el estudio.

El problema clave que intenta abordar el proceso del AID es la transformación de datos sin relevancia en información válida que permita la obtención de modelos de conocimiento. Por tanto, se puede definir el AID como un proceso, no trivial, que persigue la generación de conocimiento mediante la identificación de información válida, así como de los patrones que en ésta y en los datos pudieran haber [2]. El AID es un proceso complejo que en general sigue una serie de fases interactivas e iterativas que son descritas a continuación y que se pueden visualizar gráficamente en la Figura 1.2:

- Integración y recopilación de datos: determinar el dominio de aplicación del problema que se presenta para identificar el objetivo a cumplir y las posibles fuentes de información.

- Preprocesamiento de datos: procesar los datos para mejorar la calidad de los mismos, eliminando outliers y completando aquellos que no se hayan podido obtener por algún problema hardware o software.

- Minería de datos: construir uno o varios modelos que permitan alcanzar el objetivo planteado.

- Evaluación e interpretación: analizar, mediante la participación de expertos, la robustez y adecuación del modelo obtenido como solución al problema planteado.

- Difusión y uso de modelos: utilizar el modelo para solucionar el problema planteado en la hipótesis de partida.

Aunque esta tesis aborda todas las fases del proceso de AID, principalmente se va a trabajar en las fases de preprocesamiento y minería de datos desde dos puntos de vista, la computación en la nube y la propia parcela. Esta segunda opción se implementará mediante Edge Computing, el cual permite ejecutar ciertas tareas, incluido el modelo, sin necesidad de conectar con el servidor [20].

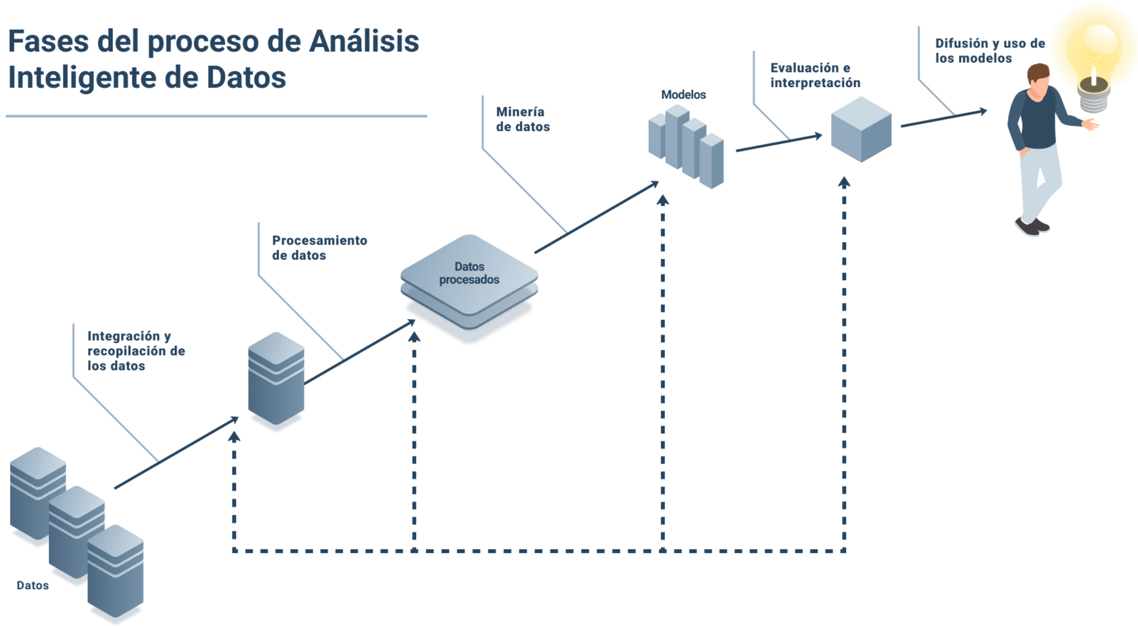


Figura 1.2: Fases del Análisis Inteligente de Datos

En cuanto a la fase de preprocesamiento, en esta tesis se aplican dos técnicas: detección de outliers basada en el vecino más cercano y clustering mediante k-means [12]. Ambas permiten la detección de datos erróneos, también llamados outliers, y por tanto la eliminación del ruido que esos datos pudieran producir a la construcción y ejecución del modelo.

Para la fase de minería de datos se ha hecho uso de redes neuronales profundas basadas en series temporales recurrentes, más conocidas como Long Short-Term Memory (LSTM) [18]. La aplicación de las mismas dentro del ámbito de la detección de heladas es muy adecuada, dadas las características del problema a resolver.

1.4 Objetivos

Esta tesis doctoral se asienta sobre una hipótesis inicial dividida en un conjunto de objetivos, que describimos y detallamos a continuación. Dichos objetivos son las bases sobre la que se construye la tesis doctoral y sobre los que se consigue alcanzar y validar la hipótesis planteada.

La hipótesis fundamental de esta tesis doctoral es la modelización del comportamiento de las temperaturas dentro una parcela agrícola. A partir de ahí predecir las heladas para poder activar con tiempo las técnicas antiheladas y así evitar la pérdida de la cosecha con las consiguientes pérdidas económicas. Dicha hipótesis se deriva en los siguientes objetivos:

- **Objetivo 1. Implementar una arquitectura IoT de captura de datos**. Inicialmente se propone la implementación y el despliegue de un sistema IoT de recogida de información mediante la tecnología LoRa, con sensores de medición de temperatura y humedad del aire, así como velocidad del viento.

- **Objetivo 2. Diseñar e implementar el pre-procesamiento de datos**. En la recogida de datos se pueden producir errores en los mismos, que posteriormente pueden provocar modelos incorrectos o con un rendimiento inferior al esperado. Para evitar esta problemática se aborda en esta tesis la detección y corrección de outliers para crear conjuntos de datos válidos.

- **Objetivo 3.Diseñar e implementar un modelo de predicción de heladas**. Con los conjuntos de datos validados y corregidos tras el pre-procesamiento de los mismos, diseñar e implementar un modelo de predicción de la temperatura del aire que permita inferir si se producirá una helada.

- **Objetivo 4. Validar el modelo de predicción tanto en entornos Cloud como en**

el Edge. Las condiciones de aislamiento de muchas parcelas agrícolas no permiten tener una conectividad suficiente para estar en contacto directo con un servidor en Internet, por tanto, es necesario validar que el modelo predictivo puede ser ejecutado, tanto en entornos Cloud como en entornos Edge. En este último escenario, mediante un uso eficiente de la energía.

- **Objetivo 5. Evaluación de un modelo de predicción de heladas univariable frente a uno multivariable**. La aparición de una helada, además de por la temperatura, está condicionada por otras variables. Diseñar e implementar un modelo multivariable para compararlo con el univariante y verificar que se obtiene una predicción más exacta.

1.5 Estructura del documento de tesis

En esta sección de la introducción se detalla de forma resumida el contenido de cada uno de los capítulos que componen la tesis doctoral:

- *Capítulo 1: Introducción*. En este capítulo se realiza la contextualización de la tesis doctoral, describiendo las disciplinas y campos que se abordan, así como la hipótesis de la tesis y los objetivos plateados.

- *Capítulo 2: Publicaciones que componen el compendio de la tesis doctoral*. En este capítulo se incluye la fundamentación de la tesis doctoral y la consecución de los objetivos en cada una de las publicaciones descritos en el Capítulo 1. Después se adjunta una copia de cada una de las publicaciones científicas que forman parte del compendio.

- *Capítulo 3: Conclusiones y Vías futuras*. Las publicaciones científicas del compendio de esta tesis doctoral tienen asociadas un conjunto de conclusiones que demuestran la consecución de los objetivos planteados. En este capítulo se recogen el total de conclusiones obtenidas en cada publicación, así como las posibles vías futuras de investigación que se derivan de dicho compendio.

- *Capítulo 4: Calidad de las publicaciones*. Por último, en este capítulo se detallan los indicadores de calidad de las revistas científicas de cada una de las publicaciones que integran el compendio. Además, también se indican otras publicaciones científicas derivadas de la tesis doctoral y no incluidas en el compendio.
- *Bibliografía*. En este apartado se aportan un conjunto de referencias para profundizar y obtener más información de los aspectos abarcados en la tesis doctoral.

Capítulo 2

Artículos que componen la tesis doctoral

2.1 Fundamentación de la tesis doctoral

Los objetivos planteados en el capítulo 1 para lograr la hipótesis inicial se han ido abarcando y analizando a lo largo de las cuatro publicaciones científicas que componen el compendio de esta tesis doctoral. A continuación se describe de forma general dicha investigación.

En la primera publicación se describe la situación de partida que ha motivado la hipótesis general y que ha guiado el desarrollo de la tesis. Se plantea el entorno de recogida de datos; se proponen dos mecanismos de detección y eliminación de outliers; y se analiza el uso de Edge Computing como posible solución, de cara a utilizar un sistema eficiente y totalmente autónomo.

La agricultura es un sector básico en el desarrollo de la humanidad. La optimización de los procesos agrícolas es y ha sido siempre un elemento fundamental. En la actualidad, gracias a la disciplina de la agricultura de precisión, se han incorporado elementos de monitorización y control que permiten ahorrar costes y aumentar la producción.

En este ámbito hay una situación que produce pérdidas millonarias: las heladas de los cultivos. Es verdad que existen sistemas antiheladas que reducen los riesgos pero éstos son caros y tienen un consumo de agua elevado. Por ello, se hace necesario pronosticar cuándo helará para decidir si activar o no dichos sistemas.

Se propone el despliegue de una infraestructura IoT que utiliza la tecnología LoRa para comunicarse, en una parcela de Cieza donde hay una plantación de frutales de hueso. En todos los nodos hay un sensor que registra temperatura y en uno de ellos además registra humedad. Los sensores de temperatura se han puesto a diferente altura ya que, dependiendo del tipo de helada, puede haber diferencia de temperatura entre la copa y el pie de los árboles, influyendo en la medida obtenida. Los tres nodos se han desplegado en la parcela en base a las recomendaciones de los agricultores, buscando zonas con diferente temperatura en un mismo instante de tiempo.

En todos los sistemas de estas características se producen errores en la toma de valores, por lo que es necesario detectar y eliminar los valores erróneos u outliers. En la publicación se analizan y comparan dos técnicas de detección de outlier: k-vecinos más cercanos y k-means, siendo esta última la que mejor funciona para los datos proporcionados por la infraestructura IoT.

Tras el preprocesamiento se comparan los valores obtenidos por sensores respecto a los que están disponibles en estaciones meteorológicas de la Agencia Estatal de Meteorología (AEMET) y del Instituto Murciano de Investigación y Desarrollo Agrario y Alimentario (IMIDA). Se constata que existe una diferencia sustancial entre los valores de los sensores y los de dichas estaciones; lo que valida el uso de un sistema de recogida de datos local y la implementación de un modelo de predicción específico para cada parcela.

La publicación finaliza con la comparación entre el uso de un sistema cloud de detección de outliers y un sistema Edge computing. La idea que subyace es la de implementar un sistema de predicción totalmente autónomo que no haría necesario la conexión a Internet. Evidentemente, el rendimiento a nivel computacional es mejor en la nube pero, en base a los resultados obtenidos, no es descartable el uso de sistemas Edge en entornos donde no sea factible una conexión continua a Internet, como quedará demostrado en el tercer artículo.

En este punto de la investigación quedan cubiertos los dos primeros objetivos de la tesis: diseño y despliegue de una infraestructura IoT para capturar los datos; y diseño e implementación de técnicas de preprocesados de datos, concretamente técnicas de detección y corrección de outliers.

A continuación se aborda la idoneidad de utilizar datos globales para construir el modelo, en contraposición al uso de datos locales específicos de un parcela. Para ello, además del despliegue existente en Cieza, se produce el despliegue de una infraestrctura IoT más reducida en la población de Moratalla (Murcia).

Para lograr un modelo ajustado se han comparado el modelo global, que incluyen los datos de ambas parcelas, con modelos locales que incluyen sólo los datos de una parcela.

Como los datos recogidos forman una serie temporal la red neuronal, construida es de tipo LSTM. Dicha elección se ha tomado en base a la comparativa realizada entre el modelo de LSTM con los modelos Autorregresivo Integrado de Promedio Móvil (ARIMA) y Gaussian Process.

De cara a comprobar la robustez del modelo LSTM se han realizado dos tipos de experimentos: una validación cruzada y una validación train/test, siendo 90 % para train y 10 % para test. Los conjuntos de datos utilizados han sido tres: los datos de Cieza; los datos de Moratalla; y un conjunto de datos global con los datos de Cieza y Moratalla juntos.

Con los datos de Cieza se ha conseguido el mejor resultado mientras que el global es el que peor resultado ofrece. Esto certifica que los modelos locales son mejores que los globales.

En esta misma línea, si se analizan las tendencias de las diferentes soluciones, se puede observar que cuando se predice la temperatura durante la noche, que es cuando más frío hace, los datos predichos se ajustan mucho más al valor real. Esto se debe a que los valores de temperatura son mucho más fluctuantes y están influidos por otras variables durante las horas de sol.

Una vez que los modelos han sido validados, comprobándose su robustez y fiabilidad, se realiza una predicción de las siguientes 24 horas. Como resultado, se obtiene un RMSE de 0,6524 en el conjunto de datos que representa una parcela en Cieza, en contraposición del 1,2058 del conjunto de datos globales. Algo similar ocurre con el coeficiente de determinación R^2, con un 0,9820 en Cieza y 0,9648 en el conjunto global. Por consiguiente, queda comprobado que el desarrollo de un modelo local mejora los resultados respecto a un modelo global.

Una última e importante conclusión a raíz de los datos obtenidos, es el hecho de que los casos en los que predice que no helará y finalmente si lo hace (falso negativo) representan sólo el 0,78 %. Esta es la situación más complicada ya que implicaría no activar el sistema antihelada y se perdería la cosecha. Los casos en los que se predice helada pero finalmente no hiela (falso positivo) representan un 0,61 %. Aquí la pérdida se refiere al gasto del sistema antihelada pero no a toda la cosecha.

Cabe destacar, que de cara a que el sistema sea totalmente autónomo, el modelo se ha ejecutado sobre una Graphics Processing Unit (GPU).

Con el estudio realizado se ha abordado el objetivo número tres de la tesis doctoral: diseño e implementación de un modelo de predicción de heladas, concluyéndose que los resultados óptimos se obtienen con modelos locales.

El siguiente paso realizado toma como base el modelo LSTM proponiendo el uso de tecnologías que permitan llevar a cabo las tareas de computación en la propia parcela. La idea principal es reducir consumo y realizar la predicción en áreas cuya cobertura de red impida una comunicación estable con un servidor.

En los últimos años, el uso de lo que se conoce como "Edge Computing" [20] ha ido aumentando gracias a la mejora en el rendimiento obtenido y a su eficiencia energética. Para hacer la comparativa entre ejecución en el servidor y en la parcela se ha evaluado el rendimiento y consumo en una tarjeta Nvidia Jetson AGX Xavier, tanto en entrenamiento como en inferencia.

El modelo presentado anteriormente se debía ejecutar en un servidor porque el coste computacional era grande. No obstante este escenario, con la red IoT conectada con el servidor a través de Internet, no es real en áreas rurales donde existe un ancho de banda bajo y latencia alta. Aquí aparece lo que se conoce como Edge Computing, acercando la computación, tanto entrenamiento como inferencia, a la parcela. Es evidente que se pierde potencia de cálculo pero se gana en eficiencia energética y en autonomía. En la tercera publicación se evalúa si dicha pérdida de potencia de cálculo es asumible en términos cualitativos respecto a la predicción realizada y el tiempo empleado en obtenerla.

Para realizar este análisis se han ejecutado dos implementaciones del modelo LSTM mediante la librería Keras sobre la GPU antes nombrada. Para el caso de la ejecución en modo GPU se ha adaptado el código de la LSTM para permitir cargarlo en dicha tarjeta. Se han seleccionado varios conjuntos de datos para analizar, no sólo el tiempo de computación, sino también realizar un primer análisis sobre el histórico de datos necesarios para obtener una buena predicción. Los datos utilizados han sido los desplegados desde 2017 con el sistema IoT propuesto en la primera publicación de este compendio. Además, también se ha realizado un ajuste de parámetros para evaluar los más óptimos, tanto en precisión como en tiempo de ejecución.

En cuanto a la calidad de los resultados obtenidos cabe destacar que en la configuración óptima, y para un histórico de datos de tres meses, se obtiene un RMSE de 0,6731 y un R^2 de 0,9924. Por un lado se deduce, que el uso del Edge Computing es totalmente válido para el escenario que esta tesis ha estudiado y por otro, que no se necesita un histórico de datos muy grande para obtener resultados satisfactorios. El sistema puede ser instalado en cualquier lugar ya que con datos de tres meses el modelo consigue unos resultados robustos y un error menor de 0,8 ºC.

Con esta tercera publicación abordamos el cuarto objetivo propuesto en la tesis: validar el modelo en un entorno Edge Computing.

La parte final de la investigación aborda el último objetivo de la tesis: evaluar un modelo multivariante frente al univariante a la hora de predecir heladas.

Para las fases de validación y entrenamiento del modelo se han cogido los datos históricos de una estación meteorológica del IMIDA cercana a la parcela de Cieza. El cambio ha venido motivado por la necesidad de disponer de más cantidad de variables a la hora de construir y validar el modelo.

El conjunto de datos utilizados es una serie temporal con una frecuencia de lectura de una hora. Se ha utilizado el 90 % para entrenamiento y el 10 % restante para test. En ambos casos siguiendo el orden temporal de los mismos.

También se han hecho diferentes conjuntos de datos para conseguir un modelo más ajustado; por ejemplo, eliminando los datos correspondientes a las épocas del año en las que no se producen heladas y que generaban ruido a la hora de crear el modelo.

Tras estudiar los resultados de las diferentes configuraciones se certifica que el modelo multivariable es más robusto que el univariable. Este último es más inestable cuando existe una gran diferencia entre temperaturas. Esto es lógico, ya que es la única variable que utiliza para inferir la temperatura y la posible presencia de helada.

Por tanto, el mejor modelo obtenido es un modelo multivariante que incluye temperatura, humedad y dirección del viento para predecir las heladas de forma precisa y eficiente.

2.2 A high-performance IoT solution to reduce frost damages in stone fruits

Título	*A high-performance IoT solution to reduce frost damages in stone fruits*
Autores	Miguel Ángel Guillén-Navarro, Raquel Martínez-España, Belén López, y José María Cecilia
Revista	Concurrency and Computation: Practice and Experience
Año	2021
Estado	Publicado

Contribución del Doctorando
Miguel Ángel Guillén Navarro, declara ser el principal autor y el principal contribuidor del artículo *A high-performance IoT solution to reduce frost damages in stone fruits*.

Received: 30 November 2018 | Revised: 12 March 2019 | Accepted: 21 March 2019
DOI: 10.1002/cpe.5299

WILEY

SPECIAL ISSUE PAPER

A high-performance IoT solution to reduce frost damages in stone fruits

Miguel A. Guillén-Navarro | Raquel Martínez-España | Belén López | José M. Cecilia

Department of Computer Engineering, Universidad Católica de Murcia, Murcia, Spain

Correspondence
José M. Cecilia, Department of Computer Engineering, Universidad Católica de Murcia, 30107 Murcia, Spain.
Email: jmcecilia@ucam.edu

Present Address
José M. Cecilia, Campus de los Jerónimos, Guadalupe 30107, (Murcia) Spain

Funding information
Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia, Grant/Award Number: 20813/PI/18; Spanish Ministry of Science, Innovation and Universities, Grant/Award Number: TIN2016-78799-P, (AEI/FEDER, UE) and RTC-2017-6389-5

Summary

Agriculture is one of the key sectors where technology is opening new opportunities to break up the market. The Internet of Things (IoT) could reduce the production costs and increase the product quality by providing intelligence services via IoT analytics. However, the hard weather conditions and the lack of connectivity in this field limit the successful deployment of such services as they require both, ie, fully connected infrastructures and highly computational resources. Edge computing has emerged as a solution to bring computing power in close proximity to the sensors, providing energy savings, highly responsive web services, and the ability to mask transient cloud outages. In this paper, we propose an IoT monitoring system to activate anti-frost techniques to avoid crop loss, by defining two intelligent services to detect outliers caused by the sensor errors. The former is a nearest neighbor technique and the latter is the k-means algorithm, which provides better quality results but it increases the computational cost. Cloud versus edge computing approaches are analyzed by targeting two different low-power GPUs. Our experimental results show that cloud-based approaches provides highest performance in general but edge computing is a compelling alternative to mask transient cloud outages and provide highly responsive data analytic services in technologically hostile environments.

KEYWORDS

edge computing, frost crops, GPUs, IoT system, precision agriculture

1 | INTRODUCTION

Mainly motivated by the high competition of the market, the high demands on the quality of products, and the lack of scarce resources such as water,[1] agriculture is constantly evolving toward the adoption of new technologies. Increasingly, more farmers are applying new technologies to obtain products with higher quality and less cost. A new research area has emerged as a result of these concepts called Precision Agriculture (PA).[2] Precision agriculture is an innovative, integrated, and internationally standardized approach to increase the efficiency of resource use and to reduce the uncertainty of decisions required to manage variability on farms.[3] One of the greatest dangers in agriculture is arising from inclement weather. Floods or constant temperature changes are only some examples that may jeopardize the harvest, having dramatic consequences.[4] Particularly damaging is the case of the Region of Murcia (Southeastern Spain), where crops suffer temperature variations of up to 20 degrees Celsius (°C) on the same day. For example, temperatures at midday could reach values equal to or even higher than 20°C in winter, and the temperature at night could even become negative. These temperature variations make the flowering of trees predictable, particularly in stone fruit trees. When the tree is fully bloomed, the negative temperatures may cause the flower to freeze, which could be translated in losing the harvest, and thus, having significant economic loses. Some techniques are available to protect crops from low temperatures such as protective covers, artificial fog, fans, application of chemical treatments, or sprinkler irrigation.[5,6]

The anti-freeze techniques are often too expensive and must be applied several hours in advance to really prevent frost in the crop. The main problem for farmers is the lack of frost prediction, as these techniques take several hours to protect the crop.[7] Therefore, this challenging scenario requires several ingredients, ie, (1) real-time weather information for the crop in question; (2) analytical models that, from the information gathered, are capable of predicting a frost; and (3) enough time to be able to take appropriate action. Indeed, farmers can take advantage of technological innovations in the areas of computer science, telecommunications, and agriculture to answer these factors. One of the leading drivers of the digital revolution is the Internet of Things (IoT), where devices and humans are fully connected to the global network. The IoT is being widely applied in agriculture, as they are evolving towards continuous monitoring and automatized systems.[8] The IoT revolution is underpinned by two key factors, ie, (1) data, which carries hidden patterns, correlations, and other valuable insights; and (2) real-time data analytics, since knowledge is often time-sensitive, and useful only within a specific time-frame.[9] In this paper, we propose an Internet of Things (IoT) infrastructure to prevent frost to avoid the loss of the harvest. This infrastructure is composed of several modules, in which the main contributions of this paper are developed. These contributions are listed as follows.

1. A *data acquisition* module is developed based on LoRa to provide good ratios of coverage, end-node's power consumption, and scalability.[10]
2. An IoT *intelligent* module to make predictions about when the actuator should be switched on, is included in the infrastructure. This module is composed of several algorithms to identify outliers based on the nearest neighbor technique and the k-means clustering algorithm. The early detection of outliers is an important issue, since they can cause an incorrect operation of the actuators. For instance, sprinklers could be activated resulting in a loss of water and money or not activated, or activated too late, resulting in a loss of the crop.
3. The time and high availability requirements found in this problem demand a high-performance architecture for these algorithms to be valid for frost prediction. Therefore, we propose a *performance evaluation* on low-power GPU-based computing architectures at the edge compared to their cloud-based counterpart versions.
4. The IoT system proposed is able to activate the anti-frost technique of sprinkler irrigation due to its effectiveness and low installation cost compared to other techniques.

The rest of this paper is organized as follows: Section 2 contextualizes this research and identifies the main novelties of this work with respect to previous works. Section 3 describes the components of the proposed IoT intelligent monitoring system and its practical implementation. Section 4 provides evaluation results that demonstrate the accurate operation of our platform. Finally, Section 5 provides some conclusions and directions for future work.

2 | RELATED WORK

The capacities that IoT offers can be applied in agricultural and farming applications. In the work of Ojha et al,[11] some applications that use sensors and IoT technologies were applied to agriculture. Some of these applications are irrigation management system to optimize the water use in farming; control pest, disease, and fertilizers to increase the crop quality; farming systems monitoring to improve management in large agricultural fields; cattle movement monitoring using radio frequency identifier; remote control and diagnosis of farming machinery; and monitoring the ground water quality and the greenhouse gases, just to name a few. In the work of Srbinovska et al,[12] a monitoring system to control the quality of peppers in greenhouse was presented. The topology shown is composed of a wireless sensor network (WSN) to control crops in greenhouses. The main drawback of this work is that their solution is not extrapolated to other environments. Ferentinos et al[13] also presented a study where they assess wireless sensor networks operation, reliability, and accuracy in greenhouses. The authors perform an analysis of the collected data to research possible problematic situations for the growing plants caused by climatic heterogeneity inside the greenhouse.

In the work of Lichtenberg et al,[14] a WSN was proposed to monitor water level in the farm areas for precision agriculture. In the work of Gutiérrez et al,[15] a similar network was implemented, but in this case, it is used to control the waste of water. It is noteworthy to highlight that in 136 days, they obtained a 90% reduction of water consumption compared with traditional irrigation systems. In the work of Lorite et al,[16] the development and use of a wireless radio frequency technology for the performance analysis of an irrigation scheme was presented. The authors study the use of water with different crops by using telemetry, concluding farmers behavior management is extraordinarily complex due to the multiple factor implicated in the process. In the work of Martínez et al,[17] the authors evaluated whether FIWARE platform is suitable for the development of agricultural applications. They propose an IoT scenario in the field of precision agriculture/viticulture for managing a large number of resources associated with sensors and actuators. The sensors detect data related to the environment and the condition of the vineyards (eg, water and nutrients). The actuators are installed in crop irrigation systems.

Another work where the authors propose an improvement for irrigation in agriculture was presented in the work of Cambra et al.[18] These authors implement an intelligent system for the control of bicarbonates in irrigation for precision hydroponic crops in order to improve water quality in hydroponic agriculture. They use different wireless sensors, including a pH sensor to measure nutrients. In the work of Gil-Lebrero et al,[19] the authors presented a wireless monitoring system for honey bee hives. The IoT system architecture enables the easy deployment and scalability in the field. They used commercial nodes adapted to the measurements of the hives. The nodes allow the use of different technologies according to the scope range, including the following: 3G, GPRS, LoRa, Sigfox, ZigBee, or Wi-Fi. The LoRa technology has been used in agriculture[20] to propose an embedded system called MoniSen that is capable of monitoring environmental parameters within a 10 km radius. The proposed hardware architecture is designed for continuous monitoring of five environmental parameters, ie, air temperature and humidity, soil temperature, soil moisture, and light intensity. Cambra et al[21] presented an intelligent IoT communication system manager as a low-cost irrigation controller. The proposal is a irrigation tool that periodically samples real-time data such as the variable irrigation rate, the vegetation index, and irrigation events (flow rate, pressure level, or wind speed.

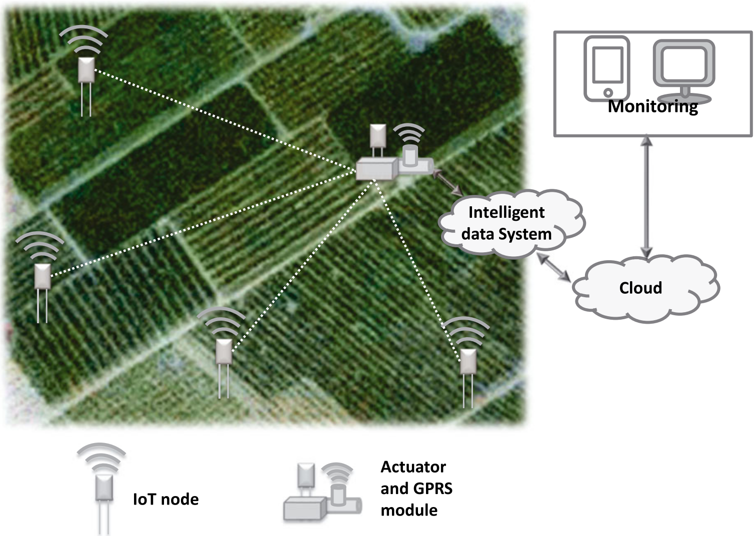


FIGURE 1 The IoT system architecture

However, to the best of our knowledge, we have not found any work that applies an IoT infrastructure to the field of frost prevention or cares about the quality of the data collected on the sensors and their validity. In addition to being in a new field of application, the IoT proposal made in this study applies techniques to correct and evaluate the quality of sensor data to detect outliers. Outlier detection is an important are within data mining and, particularly relevant in precision agriculture.[22] In WSN, outliers are defined as "measures that deviate significantly from the normal pattern of detected sensed data."[23] This definition is based on the fact that sensor nodes are assigned in WSN to monitor the physical world and, therefore, there may be a pattern that represents the normal behavior of the detected data. It is important to consider that outlier detection is a task to be highlighted within the data processing. There are many outlier detections techniques in the literature as applied to WSN. In the work of Zhang et al,[24] a review of these techniques was carried out, showing a categorization of them into statistical-based approaches, *nearest neighbor-based* approaches, classification-based approaches, *clustering-based* approaches, and spectral decomposition-based approaches. In addition, Ayadi et al[22] added to this classification of outliers detection techniques, the artificial intelligence techniques with fuzzy logic, such as artificial neural networks. Both classifications of the outlier techniques agree that KNN and clustering techniques are basic and effective techniques and they are also interpretable so they become appropriate techniques to be used by farmers. In the literature, there are different types of algorithms designed to detect outlier using the KNN algorithm and designing different distance measurements; we refer the reader to other works[25-32] for insights. Some similarity happens with clustering algorithms for outlier detection; some examples include other works.[33-37]

3 | THE IoT INFRASTRUCTURE TO AVOID THE FROST DAMAGE

This section introduces the main components of the proposed IoT solution to minimize frost damage. Figure 1 shows the conceptual architecture of the proposed IoT monitoring system to reduce the frost crops. The system consists of three main components, ie, (1) climate sensors and anti-frost actuators, (2) an intelligent data processing system, and (3) a monitoring component. The intelligent component is halfway between the cloud and the actuator, and in this paper, we analyze where it is most feasible, effective, and reliable for that component to work. These three components are fully connected through a Wireless Sensor Network (WSN) to create an out-of-the-box IoT precision agriculture tool for minimizing damages caused by frost. In summary, the system monitors the climate data of an agricultural plot and, depending on the data, the system acts by activating the anti-frost system and alerting the farmer to make the appropriate decisions if this would be necessary. In what follows, we explain the main modules of the IoT infrastructure.

3.1 | Wireless sensor network (WSN)

One of the most important decisions relies on the selected wireless communication technology. Currently, there exists a myriad of IoT communication protocols that can be used for the interconnection of IoT nodes.[8] There are many technologies available in the market but we have focused on those that achieve long transmission distances, low power consumption, minimal data transfer, and are feasible to implement

FIGURE 2 IoT node

in outdoor facilities. There are several data transmission standards and protocols, including SigFox,[*] LoRa,[†] and ZigBee,[‡] which meet these characteristics.

In the first scenario, we designed an infrastructure with nodes using ZigBee. This choice was made because ZigBee wireless protocol is considered one of the best candidate technologies for the agriculture and farming domains.[18,20] Although according to its specification it reaches 100 meters, in real use environments, ZigBee does not allow transmission distances greater than 60 meters outdoors.[38] This is a big issue for deploying IoT infrastructures in outdoor crops, and thus, we decide to use the LoRa communication protocol. Among the advantages of technology, we may highlight the low power consumption and the great coverage it is able to reach. The choice to use LoRa instead of Sigfox is based on the possibility of deploying your own network, as Sigfox imposes certain restrictions on its use.

3.2 | IoT sensor hardware description

The workflow starts at the sensor level where the climate data information is collected. The hardware architecture of the sensor network is the 4H remote control system of the company Hidroconta.[§] Figure 2 shows the IoT node developed by this company. The IoT node incorporates temperature, humidity, and wind speed sensors. These sensors have been calibrated by the company, so we have not had to perform any calibration phase. The sensors are connected to the analog inputs available on the IoT node, which can be expanded with as many sensors as required, while the actuator is connected to one of the four outputs for 12 V latch solenoid valves. The sensors and the actuator used in the proposed IoT solution are the following.

- Humidity sensor: The humidity sensor also measures temperature, the implanted model is the P3110E from COMET[¶] (see Figure 3A). With a temperature range of -30°C to 80°C with an error of ± 0.6°C and a humidity range of 0 to 100% with an error of ±3%).
- Temperature sensor: The temperature sensor is the model 60P8610 from COMET (see Figure 3B). The temperature range is from -20°C to 60°C with a resolution of 0.1°C.
- Wind speed sensor: This sensor is the model PCE-WS A from the company PCE Instruments[#] (see Figure 3C). It has a measurement range between 3 and 180 Km/h with an error of ± 1 Km/h.

The most important variable to address the frost prediction is the air temperature, but other variables such as humidity or wind speed are also relevant in this domain. Including more variables offer the possibility to create more accurate models and thus, being able to predict the frost earlier. Moreover, sometimes it is also necessary to measure the same variable several times in different places. This is the case with temperature, where height can change the capture value. In our case, the humidity sensor measures both the humidity and air temperature. Having two temperature sensors helps us to detect outliers and/or temperature errors and, on the other hand, it increases the possibility of detecting the lowest temperature to activate the anti-frost technique as soon as possible.

One of the IoT communication nodes also has an actuator connected to it that activates and deactivates the anti-frost sprinkler technique, specifically, a lach-type solenoid (see Figure 3D). Among its characteristics, we may highlight the existence of 3 direct action ways, resistance of up to 80°C and 12 bar of pressure. Beyond sensors and actuators, the IoT node also has hardware dedicated to connectivity where the sensors are connected. An important aspect is that one of the IoT nodes acts as a gateway to send data to the cloud via GPRS connection, while the

[*] https://www.sigfox.com/
[†] https://lora-alliance.org/
[‡] http://www.zigbee.org/
[§] http://www.hidroconta.com/
[¶] https://www.cometsystem.com/
[#] https://www.pce-instruments.com/

(A) (B) (C) (D)

FIGURE 3 Sensors and actuators of the IoT node. A, Humidity sensor; B, Temperature sensor; C, Wind speed sensor; D, Actuator (valve)

others connect to it using LoRa technology. All of them are autonomous since they have a battery 6 volts (V) and 12 amp (A) per hour and a solar panel of 12 V and 5 watts (W). More than sufficient capacities as the consumption in "low" mode is 126μA and 19mA with GPRS connection.

Finally, each IoT node has a micro controller with 256 KB of firmware storage and 96 KB of volatile memory for data. The latter can be expanded with a non-volatile external memory of 244 KB, and thus store up to 20.000 records. Therefore, each node, in addition to sending the information to the gateway, stores the information in its memory so that if there is a failure in the communication it can be forwarded once it is restored.

3.3 | IoT intelligent component

There are many applications where *real-time data mining* of sensor data to make intelligent decisions is essential.[39] One of this application is the approach proposed in this study for monitoring low temperatures in crops. Data measured and collected by WSNs is often unreliable and inaccurate.[24] The quality of data set might be affected by noise and error such as missing values, duplicated data, or inconsistent data. The quality of this data is due to the fact that the low cost and low quality sensor nodes have stringent resource constraints such as energy (battery power), memory, computational capacity, and communication bandwidth. Besides, operations of sensor nodes are frequently susceptible to environmental effects. Thus, it is inevitable that in such environments some sensor nodes do not work properly, which may result in noisy, faulty, missing, and redundant data, being this type of data called outliers.

The prediction of frost temperatures needs reliable and error-free data; therefore, it is necessary to apply a processing technique to these data to avoid creating confusion and errors in the IoT system proposed and implicitly to the farmer. This work proposes two different algorithms for outlier detection within this context; they are as follows: (1) k-nearest neighbors and (2) k-means clustering information. With these techniques, we not only detect outliers, but we also eliminate/transform these outlier in useful information. These two techniques are within the categorization presented in Section 2. Considering the profile of the problem and the rejection that exists in some farmers to the implementation of the new technology and after analyzing the different outliers detection techniques, these two techniques have been selected based on their interpretability and their easy explanation of its operation without going into technical details.

Neighborhood-based outlier detection

The neighborhood-based outlier detection is a lightweight technique which combines the same variables from different IoT nodes. As mentioned above, each IoT node is equipped with two temperature sensors (the humidity sensor also measures the temperature). This provides temperatures at different heights and allows a real-time outlier checking. Moreover, each IoT node stores its closest neighbors to compare its measurements with those taken by its neighbors. Each time a new IoT node is introduced into the network, it notifies all IoT nodes within a radius to be included as a neighbor. This avoids performing the search every time a value has to be checked. The process of verification of the distance can be done offline without penalizing the proposed system. It is noteworthy to highlight that the distance here also takes into account the altitude at which the sensor is located as this is a relevant parameter in temperature. Therefore, the distance between IoT nodes is always displayed taking into account the latitude, longitude, and altitude in the different part of plots and experience of the farmer.

Algorithm 1 shows the general scheme of the outlier detection technique of the nearest neighbor. This technique uses the two nearest neighbors, being $1-KNN$ the air temperature sensor of the IoT node and $2-KNN$ the nearest neighbor in distance that is not deployed in the IoT node. This decision to use only two nearest neighbors is supported by previous experiments, where it was tested that when using information from distant neighbors, the outliers value was distorted, introducing even more noise in the data. Thus, for each temperature, the difference between the temperature of the node T (node under evaluation) with the temperature of the node T_{1-KNN} (outlier1) and with the temperature of the node T_{2-KNN} (outlier2) is calculated. When the temperature T is less than 1.0°C from the nearest neighbor, there is no outlier. If the temperature T is greater than 1.0°C from its nearest neighbor ($1-KNN$) and less than 2.5°C from its second nearest neighbor ($2-KNN$), there is an outlier and the temperature shown by the system is the average of the two neighbors. If these two conditions are not given, then it is an outlier, but its second nearest neighbor ($2-KNN$) can contain another outlier; therefore, the temperature of the nearest neighbor is assigned as the output of the system. The thresholds of 1.0°C and 2.5°C have been established on the basis of experimental evidence and farmers' opinions and assessments.

Algorithm 1 Outlier detection technique based on k-nearest neighbors

procedure KNN
forall *T from the whole of IoT sensors*
    $outlier1 = Diff(T, T_{1-KNN})$
    $outlier2 = Diff(T, T_{2-KNN})$
    if $|outlier1| \leq 1.0$ **then**
        $Out = T$
        else if $(|outlier1| \geq 1.0)\&(|outlier2| \leq 2.5)$
        $Out{=}Average(T_{1-KNN}, T_{2-KNN})$
        else $Out = T_{1-KNN}$
    end if
end for
end procedure

K-means clustering

The k-means[40] is a well-known clustering technique, which is heavyweight, computationally speaking. It is an iterative procedure widely used in pattern recognition and data mining to search for statistical structures in data. It is an unsupervised data mining algorithm to make cluster from high-dimensional data. Given a data matrix composed of observations and variables, the objective is to cluster the observations into groups that are internally homogeneous and heterogeneous from group to group. The *k* of the k-means clustering method indicates the number of groups, which is actually an input parameter in the algorithm. The k-means algorithm establishes prototypes or *centroids*, which are points that represents each cluster. To decide which point belongs to each cluster, the k-means uses the Euclidean distance as a measure of the similarity between observations and clusters. The traditional algorithm uses an iterative refinement.

Given an initial set of k clusters ($c_1, c_2, \dots, c_k$) where k is an input parameter, the algorithm alternates two main steps.

1. **Assignment step** where each point is assigned to the "closest" cluster, ie, the cluster with the least squared Euclidean distance between the cluster prototype and the point.
2. **Update step** calculates the new means to be the prototypes of the new clusters.

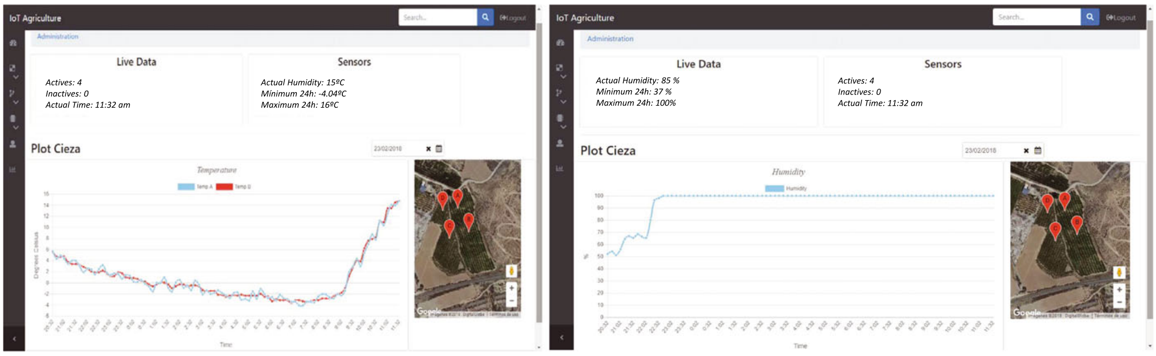
The k-means algorithm has been used in the literature for improving the outlier removal.[41,42] Actually, the term outlier is categorized into two new concepts (Internal and External outlier) when clustering techniques are applied to identify them. *External outlier* is when the outlier is an element of the group which has few data and is located far from other groups. *Internal outlier* is a point that belongs to a particular group but is located far from the centroid. In this work, we focus on the internal outliers, since in this context, the external outliers do not make sense because there is a large amount of data increasingly generated each hour to be analyzed. Once the corresponding clusters have been obtained by means of K-means, the outlier value *x* is detected as follows.

- The cluster *c* corresponding to the value *x* is assigned.
- The maximum distance between each centroid and the furthest value x^f belonging to that cluster is calculated, taking into account the signs. An outlier value can be detected by low or high temperature. This distance is called *Max*.
- If the distance of *x* to the centroid of *c* is greater than Max_c and in addition the distance between *x* and x_c^f is greater than 1°C, then the value is considered outlier.
- The outlier is corrected taking as value the centroid value of the cluster.

This K-means technique is highly parallel as the Euclidean distance calculation between points and clusters can be fully performed in parallel. Some works have proposed parallel implementations of the k-means algorithm in different platforms, including mutlicore CPUs, GPUs, and FPGAs.[43-45] Actually, the RAPIDS library recently announced by Nvidia[46] includes k-means as an accelerated method for clustering observations. In this work, we use three different implementations of k-means for outlier removal; they are sequential implementation in ANSI C to run on single-thread CPUs, multicore implementation using OpenMP[47] and GPU implementation using CUDA.[48]

When a sensor repeatedly obtains outliers during an hour of monitoring, this means that the IoT node is inactive and their values are not used to make decisions. In the test phase, we have implemented the KNN outliers detection technique in some sensors and the K-means technique in others to analyse the efficiency, speed, and quality of the results. Once the data is pre-processed to ensure data reliability, then the IoT actuators are activated if the temperature is less than or equal to 3.0°C. This decision rule is made by the experience and knowledge of experts in the field (agronomists). The rule is based on the need to moisten the entire crop area before the temperature drops from zero in order to make the anti-frost technique more effective. When the actuator is activated, the irrigation valve is opened, which allows the application of the anti-frost sprinkler irrigation technique. The IoT system deactivates the actuators when the following two scenarios occur.

- When negative temperatures have occurred and after these negative temperatures occur two hours with positive temperatures above 7°C.



(A) (B)

FIGURE 4 Monitoring interface for the intelligent system. A, Temperature view; B, Humidity view

- When temperatures do not become negative, but are below 2.5°C, and spend two hours with temperatures above or equal to 4°C. The actuators are activated again if the 2.5°C rule is met.

These temperature thresholds have been established on the basis of the experiments carried out by the farmers and it should be noted that, in the tests carried out with the IoT system proposed in this manuscript, the results have been positive using these thresholds.

3.4 | Monitoring interface

The monitoring interface provides interpretability and simplicity to reach the end user of the system. It must be appreciated that many farmers may not be familiar with the technology; hence, the development of the monitoring interface must be friendly. For the development of this monitoring interface, there was the possibility to choose between open source IoT platforms as Kaa,[‖] Fiware,[**] and ThingSpeak,[††] between others. However, due to the particularity of the problem presented, the application of outlier detection, the configuration of specific alerts that farmers demand, and the activation of the irrigation valves, an ad-hoc monitoring interface has been developed. This allows us to add the specific preferences for each farmer and to configure the alarms or conditions that are necessary to avoid damage to crops due to frost.

Figures 4A and 4B show a part of the administration panel of the IoT monitoring system. It shows the data related to a plot containing four IoT nodes active. The location data of the IoT nodes has been altered to avoid the exact location of the nodes for privacy reasons.

In the calendar icon, the user can pick a day and the interface will show the sensor selected in the map (Figure 4A, air temperature sensors, and Figure 4B, humidity sensor). Each user has his/her own personalized administration panel. These figures show the farmer has available the number of active nodes, the minimum and maximum value of the sensor, and the current value. This monitoring interface has been developed using NodeJS and AngularJS technologies. As a database manager, we work with MySQL 5.7, which allows us to store data of JSON type. This allows us to simplify the process of data manipulation. In addition, we have implemented a MongoDB database to increase system scalability.

3.5 | Deployment of the IoT monitoring system

The IoT system has been deployed in two plots of the Region of Murcia (Spain). The IoT nodes are actively working to date. This deployment was carried out according to farmers' requirements due to the temperature difference between the canopies and the trunk of the trees. The IoT nodes are implanted in a 2-meter high stainless steel support. The wind speed and temperature sensors are located in the upper part (see Figure 5A). The humidity sensor is integrated in the part of the IoT node (see Figure 5B) that is protected from rainfall with its corresponding insulators to prevent it from getting wet and sending erroneous data. In addition, the display has one of the IoT nodes connected to the valve (the IoT actuator), which opens and closes to activate or deactivate the anti-frost technique (see Figure 5C). Each plot has a actuator deployed, and three IoT nodes each with with two temperature sensors, ie, one for humidity and wind speed. The two temperature sensors of the same IoT node are at different heights in the support, and all the IoT nodes have the same configuration, although it is possible to define different heights for each sensor.

The proposed IoT system was put into operation at the beginning of October of 2017 and the data used in these assessment experiments are related to the month of November and December 2017 and January, February, March, November, and December 2018; and January and February 2019. We have to take into account for the frost domain that we are interesting to analyzed and tested low temperatures. The IoT nodes pick up temperatures 24 hours a day with a frequency between 5 and 10 minutes. However, to assess the behavior, we rely on the

[‖] https://www.kaaproject.org/
[**] https://www.fiware.org/
[††] https://thingspeak.com/

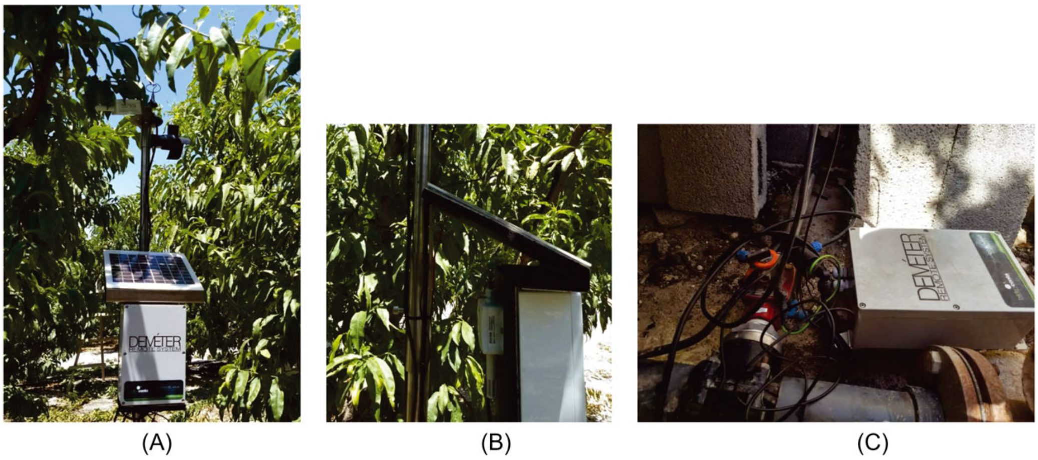


FIGURE 5 A, Deployment of an IoT node with the three sensors in a real environment; B, Detail of the displayed humidity sensor; C, Deployment of the actuating IoT node in a real environment

temperatures collected between 5 pm a day and 12 am hours a day following. Thus, analyzing the temperatures in this range, we avoid problems with high temperatures at midday since these temperatures can be influenced by the position of the IoT nodes, ie, whether these nodes are in the sun or not. It is worth mentioning that this IoT infrastructure is designed to be scalable. The user can include as IoT nodes as desired, and it can include more sensors per IoT node using expansion slots. This offers the possibility to apply this infrastructure in different agricutlural context.

4 | RESULTS AND DISCUSSION

This section shows the results of the proposed IoT solution to address the problem of reducing frost damage to crops. It is structured as follows. First of all, we motivate the need of deploying an IoT system to deal with the frost problem instead of relying on coarse-grained forecast systems, where the predictions expand to a wide geographical area. The IoT infrastructure is then evaluated in several ways, ie, (1) the most appropriate frequency and position (height) for collecting temperatures are analyzed to determine the best sensor configuration and (2) the intelligent component of IoT is evaluated from the quality's (outlier's removal) and performance's (edge vs cloud approach) point of views.

4.1 | Global vs local temperature

General weather forecasting has several limitations. Among them, we may highlight the following: (1) the measurements are generally provided hourly and (2) it is a coarse-grained prediction based on weather station at a given location. The prevention of frost requires real-time information for the particular area where the crops are located. Figure 6A shows a comparison between the temperature measurements made by the proposed IoT infrastructure and the predictions made from the Spanish Meteorological Agency (AEMET) and the OpenWeather website[‡‡] between 21.00 hours on January 10, 2019 and 11.00 hours on January 13, 2019. Although the trend is quite similar, there are significant differences between them. Openweather's prediction is less accurate than AEMET's; AEMET's prediction still differs greatly from in situ IoT infrastructure. Particularly detrimental to us are the differences in low temperatures, as such a deviation could mean the loss of the crop. Moreover, our IoT temperature information is also compared to weather stations that provide real-time information. These stations belong to the Murcian institute for agricultural and food research and development (IMIDA).[§§] In particular, the two IMIDA stations closest to the crop are selected, ie, CI22 and JU81. These stations are located at a maximum distance of 10 km from the IoT infrastructure. Figure 6B shows that there are still big differences with the actual temperature collected by our IoT infrastructure.

4.2 | Difference of temperature according to height and frequency capture

Figure 7 shows temperature variation when it is measured at different frequencies, ie, 10 and 5 minutes. This is commonly used frequencies for collecting temperatures in different IoT scenarios such as IMIDA. As the frequency of data collection increases, more information becomes available and, therefore, additional patterns could be detected. Whenever the temperature begins to drop, the 5-minute temperature appears to indicate the trend to the next temperature, although this does not have to happen in all cases. For example, the temperature at 3:04 am was

[‡‡]www.openweather.com
[§§]http://www.imida.es/

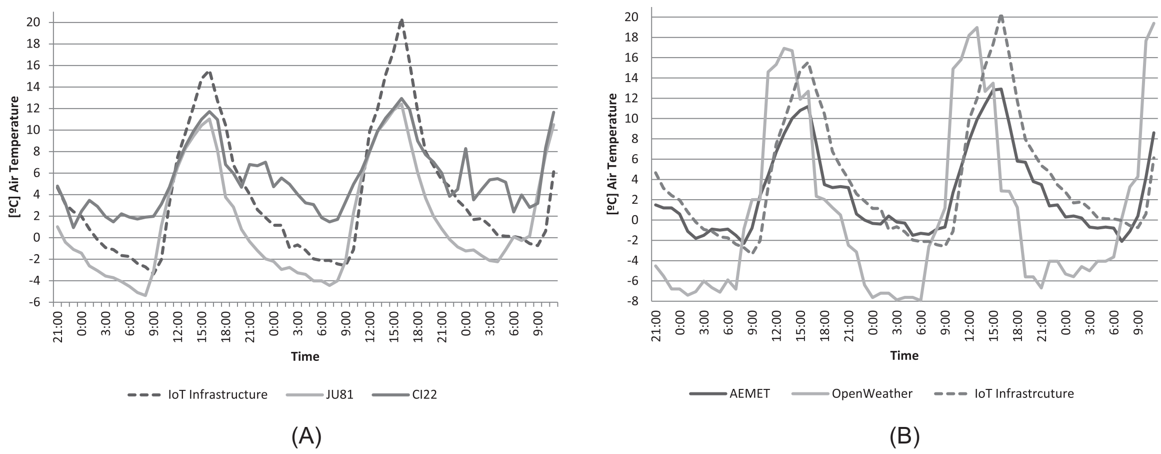


FIGURE 6 Comparative of global and local temperatures. A, Global temperature prediction vs local real temperature; B, Global real temperature vs local real temperature

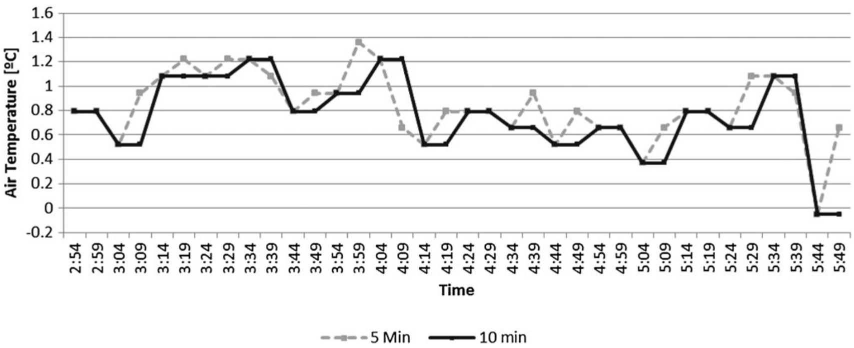


FIGURE 7 Comparative of IoT nodes collect temperature frequencies every 5 to 10 minutes

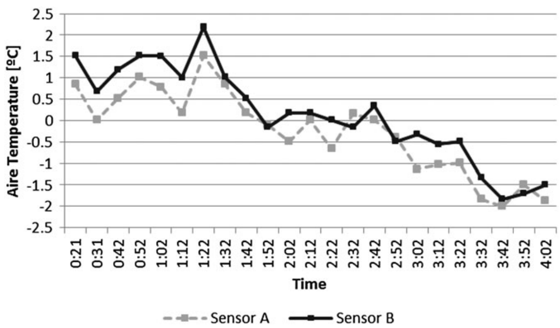


FIGURE 8 Temperature comparison of sensors at different heights

0.79°C; 5 minutes later, the temperature was 0.94°C; and at 3:14 am, it was 1.08°C, so the intermediate reading at 5 minutes indicates that the temperature was rising. Therefore, the frequency has been set every 5 minutes to provide greater information, an accuracy in displaying temperatures and warning farmers of possible frost early, and to activate the anti-frost technique.

Figure 8 shows the air temperatures for sensors A and B where sensor A is located higher than sensor. As explained above, the proposed IoT infrastructure is composed of up to two IoT nodes at different heights (see Figure 5). Temperature differences of these sensors never exceed one degree Celsius. For instance, from 2:42 am to 3:52 am, the sensor A had a higher temperature than the sensor B. This is not a single event, but it occurs several times. Moreover, it is worth highlighting that the biggest temperature differences occur at 1:12 am and 3:02 am. In addition, it should be noted that the greatest temperature differences occur at 1:12 am and 3:02 am. Having information at several heights enriches the

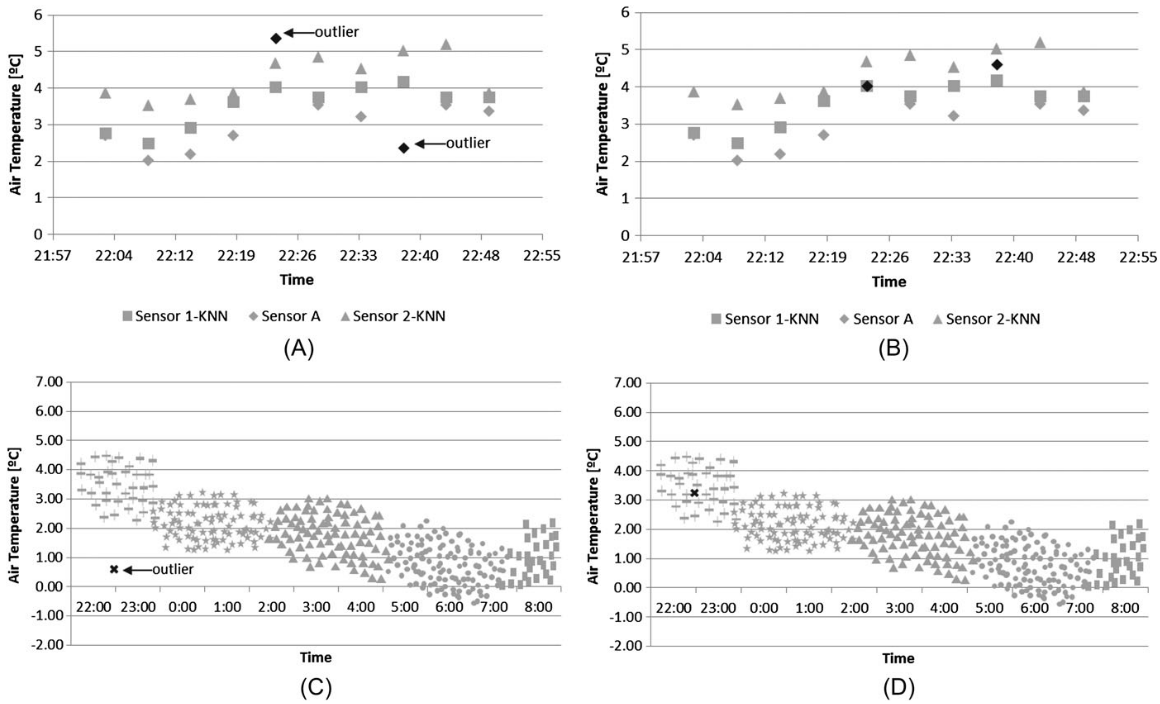


FIGURE 9 A, Air temperature detecting outlier with KNN technique; B, Air temperature with the outlier corrected with KNN technique; C, Air temperature detecting outlier with K-means technique; D, Air temperature with the outlier corrected with K-means technique

information available for decision-making. A degree of temperature difference can mean the loss of the crop, and thus, it is important to activate the antifreeze techniques in the worst-case scenario.

4.3 | Assessing the IoT intelligent component

As described in Section 3.3, the IoT intelligent component is composed of two outlier detection techniques based on the KNN and k-means algorithms. The detection of outliers in this system is a fundamental task due to the possible errors that sensors may have. Actually, outliers are a common problem in agriculture due to the harsh weather conditions in which IoT infrastructures are deployed. This section evaluates the intelligent component of the IoT infrastructure proposed. To perform this evaluation in parallel, two sensors have implemented the outlier detection technique based on K-means and the rest of sensors have used the KNN technique because the latter technique requires some neighbors to check its feasibility.

Figures 9A and 9B show ten temperatures relative to January 6, 2018 between 22:20 and 23:55 hours. The dots in bold represent outliers. Figure 9A shows the raw temperatures that includes the outlier values. Figure 9B shows the outlier values corrected by the KNN technique (see Algorithm 1). In Figure 9A, the temperatures marked as outliers are 5.36°C and 2.36°C. The KNN replaces these values by 4.02°C and 4.65°C, respectively. The former is replaced by 1-KNN and the latter is calculated by the average of the temperatures between 1-KNN and 2-KNN. If these outliers are not corrected, the actuators (valves) would be activated incorrectly, resulting in a loss of resources.

Figures 9C and 9D show the five clusters made for one sensor by the k-means algorithm during January 7, 2018 between 10:00 p.m. and 8:00 a.m. of the following day. Clusters are built using data from 1 month prior to that date and are updated if necessary each time a new value is collected. The outlier detected between 22.00 and 23.00 hours corresponds to a temperature of 0.6°C. The outlier is detected since the temperature obtained by the sensor is higher than 1.0°C with respect to the maximum distance between the temperature furthest from the centroid. The value of the corrected outliers is the centroid value of the cluster to which this value belongs to. The K-means algorithm requires much more information than KNN for the detection of outliers, but it provides a higher accuracy.

4.4 | Cloud vs edge computing

The k-means algorithm is a more accurate tool for outlier detection, but it is computationally expensive. In the agriculture application domain, the lack of connectivity, the low-bandwidth connection is big trouble when IoT analytics services are deployed. Therefore, this section analyzes the k-means algorithm from the computational point of view. Computational experiments have been carried out considering the IoT infrastructure under study where cloud and edge computing approaches are evaluated. For the cloud-based approach, the IoT infrastructure is connected to a computational back-end called *Heterolistic*, which includes two hexa-core Intel Xeon E5-2650 at 2.20 GHz, 128 GB of RAM, private L1 and L2 caches of 32 KB and 256 KB per node, and a L3 cache of 32 MB shared by all the cores of a socket. Moreover, it includes two GPUs, ie, Nvidia GTX 1080 Ti (Pascal), with 12 GB and 3584 cores (28 SM and 128 SP per SM), and an Nvidia TITAN X (Maxwell) with 3072 (24 SM and 128 SP per SM) cores and 12 GB of Global Memory. For the edge computing evaluation, the nodes are System on Chip (SoC) based on Nvidia Jetson family. Particularly, Jetson TK1 and TX1 are under evaluation. The NVIDIA Tegra K1 SoC uses an NVIDIA Kepler GPU core (192 CUDA cores) with NVIDIA 4-Quad-Core ARM Cortex-A15 CPU, 2 GB x16 Memory with 64-bit Width, and 16 GB 4.51 eMMC Memory. The NVIDIA Tegra X1 SoC uses an NVIDIA Maxwell GPU core (256 CUDA cores) and 4-Quad-Core ARM A57/2 MB L2, 4 GB 64 bit LPDDR4 (25.6 GB/s), and 16 GB eMMC, SDIO, SATA.

A set of hydrometeorological data with up to 239 108 instances is used for the evaluation, each with a maximum of 13 variables. The data set contains hourly information from January 2012 to April 2018 for the following variables: mean humidity, maximum humidity, minimum humidity, mean radiation, maximum radiation, mean wind speed, maximum wind speed, mean wind direction, precipitation, mean pressure vapour deficit, dew point, difference with the mean temperature of one hour from the previous hour, and minimum temperature.

Figure 10 shows the execution time in seconds of the k-means algorithm for the target dataset. Three different multicore architectures (Jetson TK1, Jetson TX1, and Heterolistic) are analyzed by varying the number of threads. The highest performance is achieved with 8 threads in Jetson TK1, 4 threads in Jetson TX1, and 32 threads in Heterolistic. The speed-up factor between Jetson architectures is up to 1.5x and between

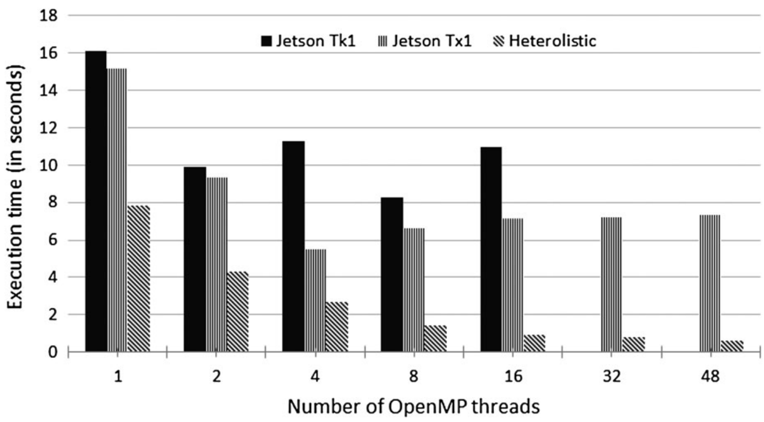


FIGURE 10 Execution time in seconds of the k-means algorithm running 1000 iterations, four clusters, and a hydrometeorological dataset with 13 variables and 239 108 instances

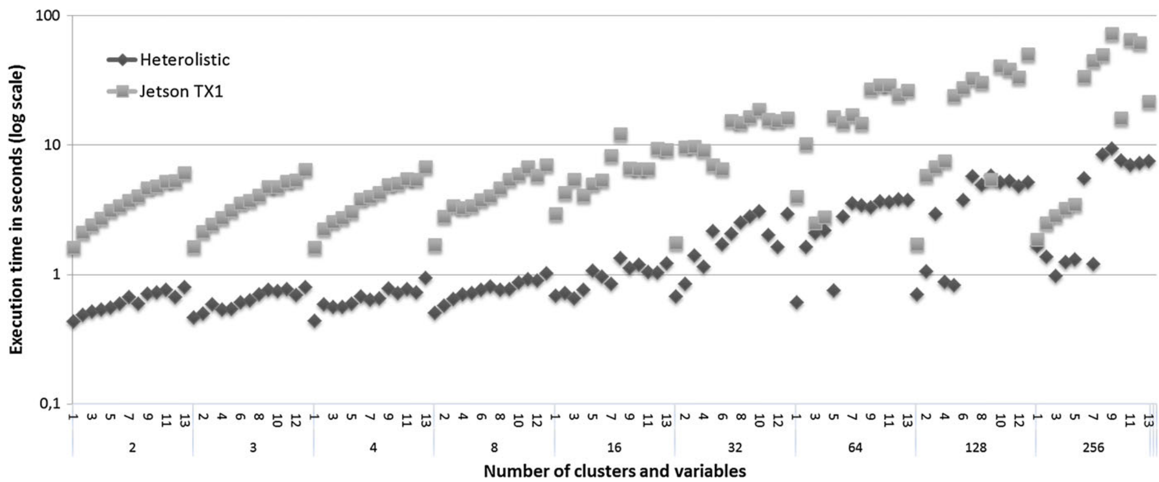


FIGURE 11 Execution time in seconds of k-means algorithm running 1000 iterations for the hydrometeorological dataset with 239 108 instances. The number of variables (from 1 to 13) and clusters (2, 3, 4, 8, 16, 32, 64, 128, and 256) is modified to analyze the behaviour of both Heterolistic and Jetson Tx1 under different workloads

Jetson TX1 and Heterolistic is almost 7x (Heterolistic classifies this dataset in 0,794 seconds and Jetson TX1 in 5,467 seconds). Therefore, the cloud-based approach offers much better performance than Jetson nodes and they are designed to be power-efficiency. From now on, we focus our performance evaluation on Jetson TX1 as it provides better performance than the previous generation.

Figure 11 shows the execution time in seconds (logarithmic scale) of k-means algorithm running 1000 iterations for the hydrometeorological dataset with 239 108 instances. The number of variables goes from 1 to 13 and the number of clusters is in the range of (2, 3, 4, 8, 16, 32, 64, 128, and 256). Notice that the performance of this code is mainly affected by the number of clusters to be performed, the number variables and registers. The cloud-server (Hetorolistic) defeats by a wide margin the low-power consumption architecture Jetson Tx1, generally speaking. The speed-up factor first quartile is 2,3x, the median is 4,3x, and the third quartile is 6,2x for all benchmarking configurations. However, the lack of connectivity and the low-bandwidth connections in agriculture scenarios may justify the possibility to compute on the edge for some cases. Particularly, there is up to 25% of the cases where the performance gap is only 2,3x (Q1) which can be hidden by the network overhead. The lowest differences in performance between both platforms are found when those variables are relatively small. Network latency tests have been carried out using the PingTools program,[49] obtaining rates higher than 2 seconds. Although the device used for them is an Android phone configured to send through the GPRS network, the data obtained is considered better than what can be achieved by measuring directly on the device of the communications node. Therefore, it is not contradicted the hypothesis that the computation in the device is more optimal for the scenario that is described in this article.

5 | CONCLUSIONS AND FUTURE WORK

Climate change is causing trees bloom earlier and the sudden changes in temperature can lead to severe economic damage to farmers, since freezing flowers and spoiling crops. In this article, we have proposed and deployed a high-performance IoT system to reduce frost damages in stone fruit. We design the whole IoT infrastructure in a technological hostile area, which includes sensors/actuators, a wireless sensor network, an intelligent component to detect outliers in real time, and a monitor interface. Regarding the intelligent component, two different algorithms to remove outliers are designed, ie, the KNN and k-menans. The latter provides better quality results, but it is a heavy workload that is well-suited for parallelization. Therefore, a comparison between cloud-edge approaches to perform the k-means algorithm is provided. Different low-power GPU-based computing architectures based on Nvidia Jetson architectures are analyzed at the edge on this IoT infrastructure. Indeed, the cloud-based architecture offers higher performance in general, but edge computing is not too far away from it computationally speaking, and therefore, we really believe that it is a compelling alternative to mask transient cloud outages and provide highly responsive data analytic services in such technologically hostile environments. After the evaluation of the goodness of the IoT system, this is deployed and operating in a real environment giving positive and satisfactory results and covering the needs of farmers facing the problem of frost.

We acknowledge that the classification algorithms pointed out in this work are very basic and other classification algorithms, such as those within the umbrella of deep learning, could obtain better results. However, the computational requirements of these promising algorithms may be too high for this emerging landscape of computation and the power consumption may be a limiting factor. Increasing the computing requirements would require a broader evaluation of edge computing platforms in terms of performance and power consumption. We definitely believe this is an interesting tradeoff to evaluate in future work.

ACKNOWLEDGMENTS

This work was partially supported by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20813/PI/18, and by Spanish Ministry of Science, Innovation and Universities under grants TIN2016-78799-P (AEI/FEDER, UE) and RTC-2017-6389-5. Finally, we thank the farmers for the availability of their resources to be able to asses and improve the IoT monitoring system proposed.

ORCID

José M. Cecilia https://orcid.org/0000-0001-5648-214X

REFERENCES

1. Melgarejo-Moreno J, López-Ortiz M-I, Fernández-Aracil P. Water distribution management in South-East Spain: a guaranteed system in a context of scarce resources. *Sci Total Environ*. 2019;648:1384-1393.
2. Norton GW, Swinton SM. Precision agriculture: global prospects and environmental implications. In: *Tomorrow's Agriculture: Incentives, Institutions, Infrastructure and Innovations-Proceedings of the Twenty-Fourth International Conference of Agricultural Economists*. Routledge: Abingdon, UK; 2018:269.
3. Liaghat S, Balasundram SK. A review: the role of remote sensing in precision agriculture. *Am J Agric Biol Sci*. 2010;5(1):50-55.
4. Nelson GC, van der Mensbrugghe D, Ahammad H, et al. Agriculture and climate change in global scenarios: why don't the models agree. *Agricultural Economics*. 2014;45(1):85-101.

5. Crookston RK. A top 10 list of developments and issues impacting crop management and ecology during the past 50 years. *Crop Science*. 2006;46(5):2253-2262.
6. Guillén-Navarro MÁ, Pereñíguez-García F, Martínez-España R. Iot-based system to forecast crop frost. Paper presented at: 2017 International Conference on Intelligent Environments (IE); 2017; Seoul, South Korea.
7. Guillén-Navarro MÁ, Cadenas JM, Garrido MC, Ayuso B, Martínez-España R. A preliminary study to solve crop frost prediction using an intelligent data analysis process. In: *Intelligent Environments 2018*. Amsterdam, The Netherlands: IOS Press; 2018:97-106. *Ambient Intelligence and Smart Environments*; vol. 23.
8. Tzounis A, Katsoulas N, Bartzanas T, Kittas C. Internet of Things in agriculture, recent advances and future challenges. *Biosystems Engineering*. 2017;164:31-48.
9. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things (IoT): a vision, architectural elements, and future directions. *Futur Gener Comput Syst*. 2013;29(7):1645-1660.
10. Sanchez-Iborra R, Sanchez-Gomez J, Ballesta-Viñas J, Cano M-D, Skarmeta A. Performance evaluation of LoRa considering scenario conditions. *Sensors*. 2018;18(3):772.
11. Ojha T, Misra S, Raghuwanshi NS. Wireless sensor networks for agriculture: the state-of-the-art in practice and future challenges. *Comput Electron Agric*. 2015;118:66-84.
12. Srbinovska M, Gavrovski C, Dimcev V, Krkoleva A, Borozan V. Environmental parameters monitoring in precision agriculture using wireless sensor networks. *J Clean Prod*. 2015;88:297-307.
13. Ferentinos KP, Katsoulas N, Tzounis A, Bartzanas T, Kittas C. Wireless sensor networks for greenhouse climate and plant condition assessment. *Biosystems Engineering*. 2017;153:70-81.
14. Lichtenberg E, Majsztrik J, Saavoss M. Grower demand for sensor-controlled irrigation. *Water Resour Res*. 2015;51(1):341-358.
15. Gutiérrez J, Villa-Medina JF, Nieto-Garibay A, Porta-Gándara MÁ. Automated irrigation system using a wireless sensor network and GPRS module. *IEEE Trans Instrum Meas*. 2014;63(1):166-176.
16. Lorite IJ, Santos C, García-Vila M, Carmona MA, Fereres E. Assessing irrigation scheme water use and farmers' performance using wireless telemetry systems. *Comput Electron Agric*. 2013;98:193-204.
17. Martínez R, Pastor J, Álvarez B, Iborra A. A testbed to evaluate the fiware-based IoT platform in the domain of precision agriculture. *Sensors*. 2016;16(11):1979.
18. Cambra C, Sendra S, Lloret J, Lacuesta R. Smart system for bicarbonate control in irrigation for hydroponic precision farming. *Sensors*. 2018;18(5).
19. Gil-Lebrero S, Quiles-Latorre F, Ortiz-López M, Sánchez-Ruiz V, Gámiz-López V, Luna-Rodríguez J. Honey bee colonies remote monitoring system. *Sensors*. 2016;17(1):55.
20. Ilie-Ablachim D, Pătru GC, Florea I-M, Rosner D. Monitoring device for culture substrate growth parameters for precision agriculture: acronym: MoniSen. Paper presented at: 2016 15th RoEduNet Conference: Networking in Education and Research; 2016; Bucharest, Romania.
21. Cambra C, Sendra S, Lloret J, Garcia L. An IoT service-oriented system for agriculture monitoring. Paper presented at: IEEE International Conference on Communications (ICC); 2017; Paris, France.
22. Ayadi A, Ghorbel O, Obeid AM, Abid M. Outlier detection approaches for wireless sensor networks: a survey. *Computer Networks*. 2017;129:319-333.
23. Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. *ACM Comput Surv*. 2009;41(3):15.
24. Zhang Y, Meratnia N, Havinga PJM. Outlier detection techniques for wireless sensor networks: a survey. *IEEE Commun Surv Tutor*. 2010;12(2):159-170.
25. Lijun C, Xiyin L, Tiejun Z, Zhongping Z, Aiyong L. A data stream outlier delection algorithm based on reverse k nearest neighbors. Paper presented at: 2010 International Symposium on Computational Intelligence and Design (ISCID); 2010; Hangzhou, China.
26. Kontaki M, Gounaris A, Papadopoulos AN, Tsichlas K, Manolopoulos Y. Continuous monitoring of distance-based outliers over data streams. In: Proceedings of the 27th International Conference on Data Engineering (ICDE); 2011; Hannover, Germany.
27. Yang D, Rundensteiner EA, Ward MO. Neighbor-based pattern detection for windows over streaming data. In: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology; 2009; Saint Petersburg, Russia.
28. Buthong N, Luangsodsai A, Sinapiromsaran K. Outlier detection score based on ordered distance difference. Paper presented at: 2013 International Computer Science and Engineering Conference (ICSEC); 2013; Bangkok, Thailand.
29. Abid A, Kachouri A, Mahfoudhi A. Anomaly detection through outlier and neighborhood data in wireless sensor networks. Paper presented at: 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP); 2016; Monastir, Tunisia.
30. Wang X, Wang XL, Ma Y, Wilkes DM. A fast MST-inspired kNN-based outlier detection method. *Information Systems*. 2015;48:89-112.
31. Schubert E, Zimek A, Kriegel H-P. Fast and scalable outlier detection with approximate nearest neighbor ensembles. In: *Database Systems for Advanced Applications: 20th International Conference, DASFAA 2015, Hanoi, Vietnam, April 20-23, 2015, Proceedings, Part II*. Cham, Switzerland: Springer International Publishing; 2015:19-36.
32. Yosipof A, Senderowitz H. k-nearest neighbors optimization-based outlier removal. *J Comput Chem*. 2015;36(8):493-506.
33. Yu D, Sheikholeslami G, Zhang A. Findout: finding outliers in very large datasets. *Knowl Inf Syst*. 2002;4(4):387-412.
34. Rahman MA, Ang KL-M, Seng KP. Unique neighborhood set parameter independent density-based clustering with outlier detection. *IEEE Access*. 2018;6:44707-44717.
35. Zhang Z, Zhu M, Qiu J, Liu C, Zhang D, Qi J. Outlier detection based on cluster outlier factor and mutual density. In: *Computational Intelligence and Intelligent Systems: 10th International Symposium, ISICA 2018, Jiujiang, China, October 13-14, 2018, Revised Selected Papers*. Berlin, Germany: Springer; 2018:319-329.
36. Campello RJGB, Moulavi D, Zimek A, Sander J. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Trans Knowl Discov Data*. 2015;10(1):5.
37. Timón I, Soto J, Pérez-Sánchez H, Cecilia JM. Parallel implementation of fuzzy minimals clustering algorithm. *Expert Syst Appl*. 2016;48:35-41.
38. Kuzminykh I, Snihurov A, Carlsson A. Testing of communication range in ZigBee technology. Paper presented at: 2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM); 2017; Lviv-Polyana, Ukraine.

39. Ma X, Yang D, Tang S, Luo Q, Zhang D, Li S. Online mining in sensor networks. In: *Network and Parallel Computing: IFIP International Conference, NPC 2004, Wuhan, China, October 18-20, 2004. Proceedings*. Berlin, Germany: Springer; 2004:544-550.
40. Linde Y, Buzo A, Gray R. An algorithm for vector quantizer design. *IEEE Trans Commun*. 1980;28(1):84-95.
41. Yoon K-A, Kwon O-S, Bae D-H. An approach to outlier detection of software measurement data using the k-means clustering method. In: Proceedings of the First International Symposium on Empirical Software Engineering and Measurement (ESEM); 2007; Madrid, Spain.
42. Chawla S, Gionis A. k-means–: a unified approach to clustering and outlier detection. In: Proceedings of the 2013 SIAM International Conference on Data Mining; 2013; Austin, TX.
43. Li Y, Zhao K, Chu X, Liu J. Speeding up k-means algorithm by GPUs. *J Comput Syst Sci*. 2013;79(2):216-229.
44. Hussain HM, Benkrid K, Erdogan AT, Seker H. Highly parameterized k-means clustering on FPGAS: comparative results with GPPs and GPUs. Paper presented at: 2011 International Conference on Reconfigurable Computing and FPGAs (ReConFig); 2011; Cancún, Mexico.
45. Baydoun M, Ghaziri H, Al-Husseini M. CPU and GPU parallelized kernel k-means. *J Supercomput*. 2018;74(8):3975-3998.
46. Rapids: open GPU data science. https://rapids.ai/
47. OpenMP ARB. Openmp specification. 2017. https://www.openmp.org/
48. NVIDIA Corporation. Cuda C Programming Guide, version 9.1. 2018.
49. PingTools: a useful set of network utilities. https://www.pingtools.org/

How to cite this article: Guillén-Navarro MA, Martínez-España R, López B, Cecilia JM. A high-performance IoT solution to reduce frost damages in stone fruits. *Concurrency Computat Pract Exper*. 2019;e5299. https://doi.org/10.1002/cpe.5299

2.3 A deep learning model to predict lower temperatures in agriculture

Título	*A deep learning model to predict lower temperatures in agriculture*
Autores	Miguel Ángel Guillén-Navarro, Raquel Martínez-España, Antonio Llanes, Andrés Bueno-Crespo, y José María Cecilia
Revista	Journal of Ambient Intelligence and Smart Environments
Año	2020
Estado	Publicado

Contribución del Doctorando
Miguel Ángel Guillén Navarro, declara ser el principal autor y el principal contribuidor del artículo *A deep learning model to predict lower temperatures in agriculture*.

Journal of Ambient Intelligence and Smart Environments 12 (2020) 21–34
DOI 10.3233/AIS-200546
IOS Press

A deep learning model to predict lower temperatures in agriculture

Miguel A. Guillén-Navarro, Raquel Martínez-España *, Antonio Llanes, Andrés Bueno-Crespo and José M. Cecilia
Computer Science Department, Universidad Católica de Murcia, Murcia, Spain
E-mails: maguillen@ucam.edu, rmartinez@ucam.edu, allanes@ucam.edu, abueno@ucam.edu, jmcecilia@ucam.edu

Abstract. Deep learning techniques provide a novel framework for prediction and classification in decision-making procedures that are widely applied in different fields. Precision agriculture is one of these fields where the use of decision-making technologies provides better production with better costs and a greater benefit for farmers. This paper develops an intelligent framework based on a deep learning model for early prediction of crop frost to help farmers activate anti-frost techniques to save the crop. This model is based on a long short-term memory (LSTM) model and it is designed to predict low temperatures. The model is based on information from an IoT infrastructure deployed on two plots in Murcia (Southeast of Spain). Three experiments are performed; a cross validation to validate the model from the most pessimistic point of view, a validation of 24 consecutive hours of temperatures, in order to know 24 hours before the possible temperature drop and a comparison with two traditional time series prediction techniques, namely Auto Regressive Integrated Moving Average and the Gaussian process. The results obtained are satisfactory, being better the results of the LSTM, obtaining an average quadratic error of less than a Celsius degree and a determination coefficient R^2 greater than 0.95.

Keywords: Deep Learning, LSTM, precision agriculture, IoT

1. Introduction

Smart agriculture is extrapolating and applying the concepts, techniques and systems of Industry 4.0 to the agrarian world. With the incorporation of the advantages offered by new technologies, smart agriculture builds approaches to reorganize the entire agricultural system towards sustainable, high-efficiency, low-input agriculture [25,38]. This new approach benefits mainly from the emergence and convergence of new technologies, including the global positioning system, the geographic information system, sensors, automatic control, remote sensing, mobile computing, advanced information processing and telecommunications. All these technologies unified under the paradigm of Cloud Computing and the Internet of Things (IoT) enable the efficient deployment of new technologies to deal with several issues that affect agriculture [27,30,33]. Therefore, smart agriculture provides a real benefit to society in different ways, such as facilitating the stabilization and increase of agricultural production and improving the environmental impact of this activity [36]. Some of the activities involved within precision agriculture are the disease detection [21,26], the prediction of weather conditions [3,13], yield prediction [2,24], water saving through irrigation monitoring [7,15], just to mention a few.

In this study we focus on the paradigm of smart agriculture by dealing with the problem of predicting weather conditions. In particular, we focus on avoiding frost in crops in extensive cultivation. Low temperatures at certain times of the agricultural cycle are a major problem that can result in the loss of millions of

*Corresponding author. E-mail: rmartinez@ucam.edu.

euros.[1] This can be avoided by predicting in advance a significant drop in temperature in a certain area to activate antifreeze techniques such as the connection of windmills and stoves or the connection of heating in a greenhouse. However, if no action is taken early enough, the crop can be lost completely, leading to great economic losses for the farmer.

The issue of frost prevention is not simple, as it depends on several factors, such as temperature, humidity, wind speed, etc., but also on the location of the plot. The global weather forecast provides coarse-grained information that is not sufficient to predict frost at the plot level. Fine grain hydrometeorological information from the particular area or plot should be taken into account to create more accurate models. The different geographical conditions in which the plantations can be found within the same plot determine the climatic conditions of the crops. In order to monitor the climatic conditions of the crops automatically, an IoT system is used to measure the wind speed, temperature and humidity of a given area within a plot [12]. However, the IoT system needs an intelligent component that helps farmers take decisions to prevent damage to crops from low temperatures. This article proposes a deep learning model based on time series to predict the possibility of frost in crops, taking as input data those provided by an IoT system. Thus, a temperature prediction approach proposed allows the farmer to be able to know the possibility of a drop in temperatures and thus will be able to activate and/or prepare all the necessary resources to apply the anti-frost technique. The latter will be possible thanks to the integration of the proposed approach to predict temperature within the IoT system, as well as helping the farmer to make decisions this system can be programmed to automatically activate the anti-frost techniques.

Deep Learning represents a set of machine learning algorithms based on a set of artificial neural networks composed of complex hierarchical levels [10]. Deep learning models are beginning to be used in the world of agriculture to solve complex problems such as the classification of diseases and/or plants through images or yield predictions in crops [19]. In this study a type of recurrent neural network is proposed to create the temperature prediction model, specifically Long short-term memory (LSTM) neural network is used. This type of neural networks obtain very satisfactory results when the data have a temporal tendency, as is the case of the temperature data of a plot [29]. Therefore, the main objective of this work is to perform a preliminary analysis and design of an LSTM neural network to create a temperature prediction model to be integrated into an IoT system deployed in several agricultural plots. To achieve the best model, we compare local temperature models with a global model containing information from several plots to determine which is more accurate.

This study is organized as follow. In Section 2 a brief background review on aspects related to deep learning and precision agriculture is presented. Section 3 describes the data and techniques used for this study of temperature prediction. Finally, Section 4 presents the results and an analysis of them and Section 5 presents the conclusions and future work.

2. Background

Deep learning techniques have begun to be introduced in the field of precision agriculture to help complete and realize the challenges presented by agriculture [19].

Among the fields of application deep learning has been applied to find the classification of plant species, identification of plant diseases, identification of soil cover, classification of crop type, estimation of yields, identification of weeds, predictions on climatology, etc.

For each of these fields of application, different types of deep learning techniques are used such as convolutional neural networks (CNN), which are deep neural networks traditionally used to classify images. For instance, a new approach for detecting plant diseases is described in [31] using a trained CNN and adjusted to fit accurately into the database of the leaves of a plant that was collected independently for various plant diseases. Another work where a CNN is used is presented in [1]. The authors introduce a new approach to classify and recognize the health status of the plantation and immediately generate treatment solutions on the fly. They use a CNN to classify and recognize different kinds of plant images, detect plant diseases, and determine the growth rate of plantations. The authors of [9] propose a novel approach that uses deep learning to count the fruits of a tree. An initial CNN quickly tags large spot-based datasets. Then, a counting algorithm based on a second CNN estimates the number of fruits in each tree. Finally, a linear regression

[1] https://www.laverdad.es/murcia/ultimas-heladas-region-20190404113101-nt.html

model verifies that the estimated number of fruits coincides with the actual number of fruits. In [22] a CNN is proposed to create a system to obtain a high quality classification of field vegetation in valuable crops and weeds. Specifically this system has been applied to sugar beet fields, seeking to accurately identify weeds in the field.

The LSTM are neural networks are applied to problems where there is a dependence of temporality in the data. In the case of agriculture they are applied to multiple areas. Some examples are described below. In [37], the authors design a new LSTM model as an alternative to computationally expensive physical models to predict the long-term depth of groundwater in agricultural areas. The proposed model consists of a LSTM layer with another layer completely connected above it, with a drop method applied to the first LSTM layer. In this study, the proposed model was applied and evaluated in five sub-areas of the Hetao Irrigation District in the arid northwest of China using 14-year data. In [14] an accurate prediction model of wheat production for agriculture in Pakistan is presented. The model uses a data pre-processing smoothing mechanism, together with an LSTM-based model, to further improve the accuracy of the predictions. In [32], the authors design a high-precision identification model for haploid maize seeds from diploid seeds by applying optimal data from hyperspectral images selected by the combination of an algorithm based on LSTM and CNN.

Focusing on the problem of the prediction of weather conditions, several works have designed LSTM-based models to predict the climatic variables using different climatic variables as input, and they have shown positive results. In [28], the construction of a robust statistical model is proposed for predicting meteorological visibility based on other intermediate variables (temperature, pressure, humidity and dew point). Two single-layer and four-layer LSTM networks are used. The data have been preprocessed by means of normalization, rescaling to the range [0, 1] and using a moving average. The multilayered LSTM model proves to be the most effective. Another work that uses an LSTM to predict climate variables is presented in [35]. The variables used are temperature, humidity and wind speed. In this case the network architecture consists of two LSTM layers. The activation function chosen for the output of the dense layer is the RELU. The optimizer used is RMSProp. The data have been normalized and rescaled to the range $[-1, 1]$ and the results obtained have been satisfactory. In [20], the authors intend to model rain and runoff using LSTM network, which predict discharge for a variety of watersheds. The authors aim to demonstrate the potential of this method. Some variables used are day length, rainfall, temperature or humidity. The network is composed of 2 LSTM layers and between them a Dropout layer to avoid overtraining of the network. The difference between the existing techniques and the proposal made in this work is that here we try to predict the temperature value in order to be able to incorporate an intelligent component into an IoT system and also that only the temperature value is used.

Analyzing conventional machine learning techniques that predict climatic variables, in [23] a extreme learning machine technique is used to predict daily dew point temperature. This study proposes an algorithm based on Extreme-Machine learning and compares it against machine-learning techniques to those that it surpasses in quality. Similar results were to be expected, as they are comparing against algorithms that do not maintain temporality in data such as SVN (Support Vector Machine) and ANN (Artificial Neural Networks). In [4], a study is presented where they use a distributed algorithm based on MapReduce to predict temperature and precipitation variables using a linear regression-based algorithm. Other related paper is [18], where authors estimate and map daily mean air temperature using daytime and nighttime land surface temperatures. In this paper authors use a linear regression to estimate mean temperature, we differ with the authors that they estimate an average value of the day, however we work with values taken every ten minutes to forecast 24 hours in a row. In comparison Deep Learning techniques provide better performance when dealing with temporal data, obtaining models with a better fit and more satisfactory results.

3. Material and method

3.1. Data collecting

The data used to train and validate the model are real data obtained from a IoT system that has deployed 3 nodes in the towns of Cieza and Moratalla (Region of Murcia, Spain). A map with the location of both cities is shown in Fig. 1. The distance between the two cities is approximately 50 km. The goal is to analyze whether a global model is better than individual local models for each area.

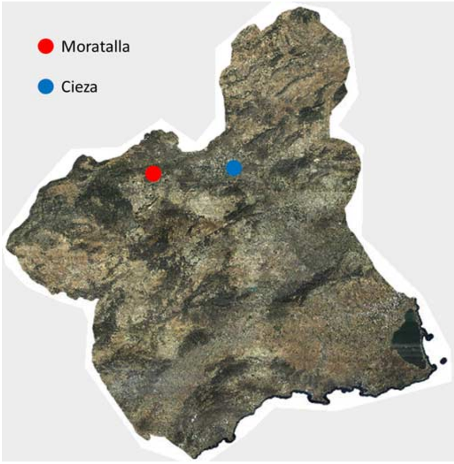


Fig. 1. Map of the region of Murcia with the location of the cities of Cieza and Moratalla.

Table 1
The format of each row collected every 10 minutes

Date (Cieza)	Temperature	Wind speed	Humidity
14/11/2018 7:10	7.76	4.21	3.25

The deployed IoT system consists of several nodes located in different areas of several plots with sensors of temperature, humidity and wind speed. A sample of each row collected is shown in Table 1. Each day will have 144 rows of values, 1 value every 10 minutes, 6 measurements per hour, during 24 hours. We can collect the *date* in dd/mm/aaaa format, *hour* in hh:mm format, and values obtained by sensors for temperature *Celsius degrees*, wind speed (m/s) and relative air humidity (%). In this work, the temperature values are used to analyse the behaviour of the IoT system considering the need for a temperature sensor, in order to reduce costs for farmers.

The IoT system, employed to collected data, consists on a 4H Demeter control unit and a wind speed, humidity and temperature sensor, configured as follows: Demeter 4H is a remote management equipment designed and implemented by Hidroconta.[2] It is a modular equipment and adaptable to the most of installations that is capable of transmitting information through different communication interfaces. These can be with other Demeter through LoRa or with a server in the cloud through GPRS. One of the most outstanding features is that it can work uninterruptedly

[2]https://www.hidroconta.com/

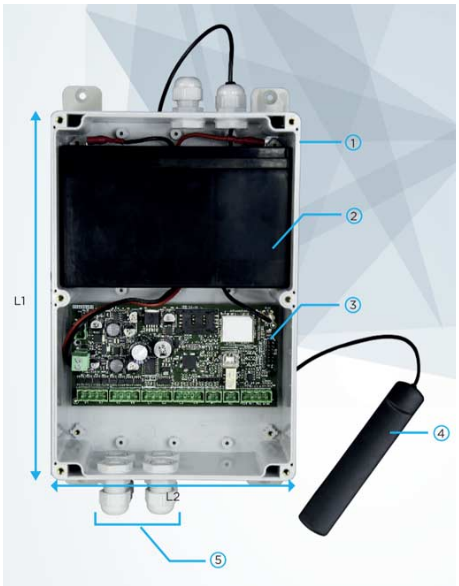


Fig. 2. Demeter 4H by Hidroconta. 1. Outer case. 2. Battery. 3. Main board. 4. Antenna. 5. Closures. 6. L1: length (26 cm). 7. L2: width (21 cm). 8. Depth: 10.5 cm.

for 4 months without communication and without loss of information. Being a totally autonomous equipment that is powered by batteries, which are recharged by a small solar panel. Different sensors can be connected to the equipment through its digital input or through its two analogicals (0–20/4–20 mA of 10 bits of resolution). This connectivity can be extended with expansion cards. Demeter 4H is operated by a microcontroller that has 256 KB of storage for the firmware developed by the same company. It also has 96 KB of volatile memory for program data. It also has a non-volatile external memory with 244 KB for storing history and configuration. Enough to store more than 20,000 records. Having 144 records per day in our work, give a total of the 4 months mentioned above. A picture of this equipment is shown in Fig. 2. Power consumption: 126 uA in low power mode (without communications), 42 uA additional for each expansion and 19 mA with GPRS connection.

The temperature and humidity sensor is a low-cost transmitter of ambient temperature and relative humidity of the Comet brand, specifically the model P3110E. Among its most outstanding features are: Relative humidity range: 0–100%. Accuracy of relative humidity measurement: ±3%. Accuracy of temperature output: ±0.6°C. Temperature range: −30 to 80°C. The trans-

Table 2
Description of the datasets

Datasets	Cieza	Moratalla	All
N.Instances	16,739	17,070	33,809

mitter contains a microprocessor based control circuit included in a plastic case with connection terminals and sensors in a stainless steel mesh filter. The humidity transmitters are equipped with two isolated 4–20 mA outputs. Speed wind sensor is the model PCE-WS. This sensor has a range between 3 and 180 km/h with an error of ±1 km/h.

The IoT nodes use Lora technology to communicate and send the data via GPRS to a data visualization application. In addition, this data is preprocessed to avoid the errors that can be proved by the sensors. The errors have been eliminated using the Kmeans algorithm for outlier detection and correction presented in [12]. The data used correspond to the period from 1/11/2018 to 28/02/2019, having temperature values (measured in degrees Celsius [°C]) with a frequency of 10 minutes. Table 2 shows the number of instances of the datasets used for the creation and validation of the LSTM model. The datasets "All" is formed by the Cieza and Moratalla data in order to analyze the global behavior of the model using both locations. The difference between the number of instances of Cieza and Moratalla within the same period is due to problems with the information collection instruments, as sometimes the data were not recorded correctly. This deficiency was later corrected by the introduction of an outliers correction technique [12].

3.2. Deep learning LSTM

LSTM networks were introduced by [16] as a model of Recurrent Neural Networks (RNN) series capable of learning long-term dependencies [16,17]. While the RNNs have a structure of chains of neural networks, the LSTM follow a similar structure, however, each part of the chain, instead of being a layer of neural network, are multiple layers, which interact between them. LSTM networks overcome the previously inherent problems and memorize temporal patterns over a long period of time. This is a problem that it presents the RNN, since it works very well with short-term dependencies, however, if RNN have to recognize a long-term dependency, they are not as efficient. These drawbacks, presented by RNN, are overcome by LSTM using memory cells and door units [16]. The door units regulate the information that can be added or removed

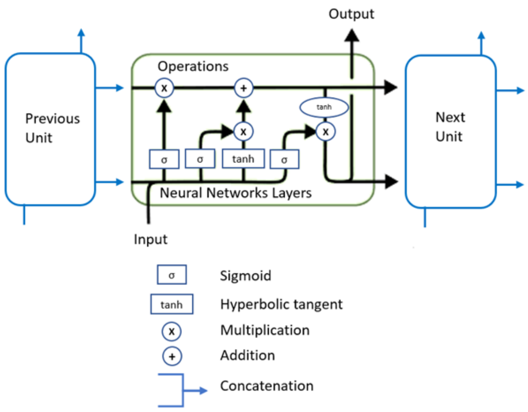


Fig. 3. A memory cell for LSTM model. Unlike RNNs, LSTMs use different door units to recognize long-term dependency.

from the memory cells. A memory cell with its different door units is shown in Fig. 3. In this figure, it can be seen that a LSTM blocks contains four layers of neural networks that interact with each other, through operations of additions, concatenations and multiplications. These four neuronal layers are made up of three sigmoids (σ) and a hyperbolic tangent (tanh). First sigmoid layer is in charge of deciding which information will be thrown away from the cell state, which is a way of forgetting unnecessary information. The next sigmoid, is responsible for deciding what information is to be stored in the cell state. To do this, it also uses a tanh layer. Finally, the last sigmoid decides which parts of the cell state will be passed to the output. In this way, and thanks to this combination of neural networks layers, it makes the LSTM block decide what information it will store in the state of the cell.

This combination of neural networks makes the LSTM block decide what information it will store in the state of the cell.

This makes the LSTMs very efficient in univariate time series models for forecasting problems. These types of problems are based on learning from the series of past observations to predict the next value in the sequence. Therefore, LSTMs are designed to remember information for long periods of time. It has been verified that these networks are especially useful in solving problems based on learning sequences of past observations, with the aim of predicting their next value [17]. The LSTM used for this study is designed with TensowFlow and Keras, and a detailed explanation of the input parameters used for the experiments are shown in the next section.

3.3. Parameters configuration

In this section parameters configuration is described, showing different tests made over all the parameters of the algorithm. This tests are made to discriminate the best parameters in the LSTM algorithm. The specific values of the different parameters are shown in Table 3, in Section 3.4, and here we show several tests previously run that justify the value selection for those parameters. In next lines, we will show the experiments for:

– Optimizer - Fig. 4
– Learning rate - Fig. 5
– Delay sequence - Fig. 6
– Activation function - Fig. 7
– Number of neurons - Fig. 8
– Batch size - Fig. 9
– Loss function - Fig. 10
– Number of epochs - Fig. 11

In Fig. 4, the results obtained with different optimizers are shown. The optimizer with more accuracy was Adam *(Adaptive Moment Estimation)* optimizer, so it was the optimizer selected. This optimizer is just an extension to stochastic gradient descent and is being used as benchmarks in deep learning problems, i.e. K. Xu et al. [34] and K. Gregor et al. [11]. Other optimizers studied are:

– SGD. Stochastic gradient descent, this is the classic algorithm used in deep learning.
– RMSProp. Root Mean Square Propagation, this algorithm does usually well on non-stationary problems.
– Adagrad. Adaptive Gradient Algorithm, that usually fits better in problems like computer vision or natural language.
– Adadelta. Adadelta seeks to reduce AdaGrad's aggressiveness by monotonously slowing down the rate of learning.
– Adamax. Adamax is a variant of Adam, but in this case is based on the infinity norm.
– Nadam. Nesterov Adam optimizer. Basically, this method is Adam RMSProp with Nesterov momentum.

In Fig. 5, Learning Rate is studied. This is a factor to control the learning variation adjusting the weights of the model. Very low values of learning rate make the learning slow, taking more time to converge. In the model was selected the learning rate with value of 0.001.

In Fig. 6, we study the number of temporal series stored in each sequence will allow the LSTM to remember more or fewer series. This hyperparameter is in problems predicting temporal series, not being common in the rest of the areas. The algorithm is configured to remember 6 delay sequences.

In Fig. 7 we show different activation function studied, in neurons, weighted sums are calculated from inputs passing through activation functions where nonlinear deformations are introduced. In this way, the computation of several neurons can be effectively

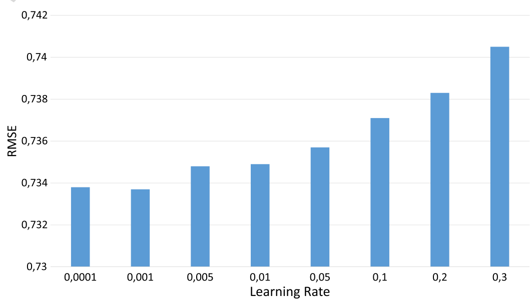


Fig. 5. Selecting the best LSTM learning rate parameter.

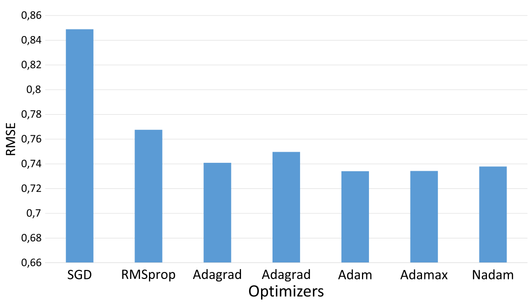


Fig. 4. Selecting the best LSTM optimizer parameter.

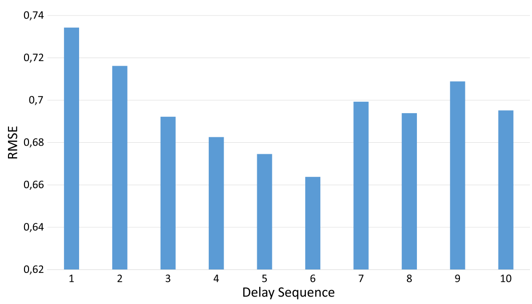


Fig. 6. Selecting the best LSTM delay sequence parameter.

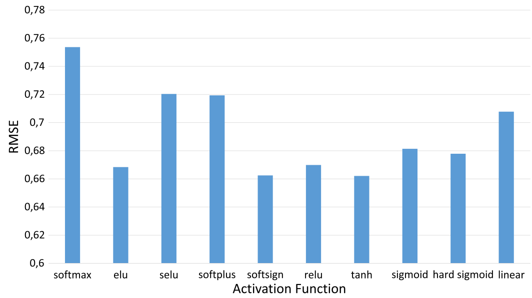


Fig. 7. Selecting the best LSTM activation function parameter.

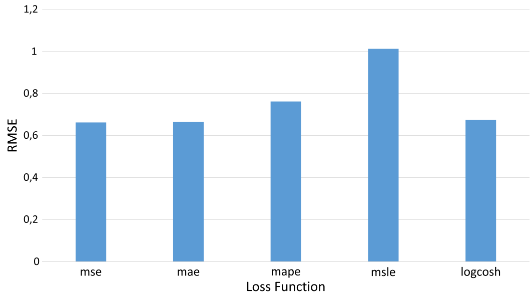


Fig. 10. Selecting the best LSTM loss function parameter.

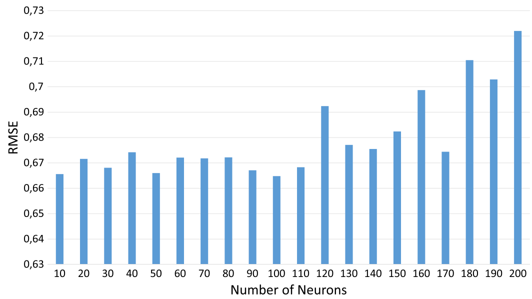


Fig. 8. Selecting the best LSTM number of neurons parameter.

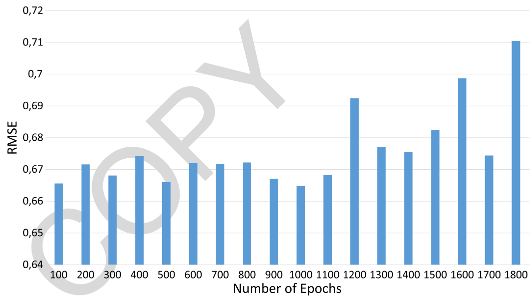


Fig. 11. Selecting the best LSTM number of epochs parameter.

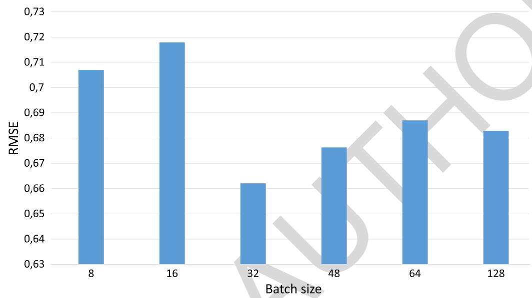


Fig. 9. Selecting the best LSTM batch size parameter.

chained. The hyperbolic tangent *tanh*, is selected as activation function in our model.

In Fig. 8 tests performed to find the optimal number of neurons that the LSTM network should have are shown. The number of neurons is established at 100 for the application.

Training data can be partitioned into small batches to train the network. Less memory is required when training the network with fewer samples and, generally, the batch size is inversely proportional to the number of times needed to train the network, since the weights of the neurons are updated after each forwardpropagation. As Fig. 9 show, different values have been tested and a size of 32 is chosen for the algorithm.

After the forwardpropagation phase, an error is calculated in order to measure the effectiveness of the prediction according to the result to be expected. There are several functions to calculate the error and to be able to adjust the weights of the neurons in the backpropagation phase. After analysing the most relevant ones in the field of linear regression, results shown in Fig. 10 are obtained. The loss function of the quadratic mean error is chosen.

Finally, epochs is the last parameter studied (Fig. 11). It indicates the number of times the training data are passed through the neural network, i.e., the iterations performed during the training process. If the number of times is very large, the model can be overtrained (overfitting), as it learns very specific characteristics of the training set without being able to generalize well. It is also possible not to train the model too much (underfitting), without converging on a point. 1000 Epochs is selected to our model. As we mentioned at the beginning of this section, the summarized values of these parameters are listed in Section 3.4, in Table 3.

Table 3
Optimal parameters for LSTM execution experiments

Parameter	Value
Number of input neurons	100
Batch size	32
Number of epochs	1000
Learning factor	0.001
Optimizer	Adam
Activation function	Hyperbolic tangent
Loss function	Quadratic mean error
Delay sequence	6

3.4. *Experiment configuration*

The objective of this work, as already indicated, is to carry out a temperature prediction model, analysing whether the best option is a global model for the two areas or a local model for each of them. To evaluate the models created by the LSTM, we will follow a conservative methodology. First we will perform a 3-cross validation for each of the indicated datasets. With this evaluation we intend to analyze the performance of the models from the point of view of robustness, without analyzing the type of error produced [5]. This first evaluation will indicate the quality and robustness of the models created, both local and global (using the "All" dataset) considering time data series. Then a second experiment is performed, specifically 90% of the data fro train models and the remaining 10% has been used as a test, choosing whole days of 24 hours. In this second experiment we analyze not only the robustness and prediction efficiency of the models but also the type of error made by the model. It must be taken into account that the same as for training the models, for the test the prediction of the temperature values is made in degree Celsius and every 10 minutes. Therefore the farmer will have updated information and predictions every 10 minutes. Finally a comparison with two traditional techniques, ARIMA and the Gaussian process is carried out, to validate and check the bias of LSTM with other methods of time series prediction.

The results of this experiment are shown and discussed in the Section 4.2. The two experiments are executed for the 3 datasets previously described. The optimal parameters are shown in the Table 3. In order to achieve the final configuration of the network, an empirical experimentation of the different parameters involved has been carried out. A brief summary of this parameterization study is shown in Section 3.3.

The quality evaluation of the model proposed is performed by measuring the goodness of the prediction by the following metrics:

- the Root Mean Square Error (RMSE)
- the Mean Absolute Error (MAE)
- the Pearson Correlation Coefficient (PCC)
- Determination coefficient (R^2)

To predict temperature, we perform a classification task to evaluate false positives and false negatives. A prediction where the real temperature is negative and the prediction is positive (false positive) is much more serious than the opposite, where the prediction indicates that there is no risk of frost and, in reality, it freezes, being able to lose the harvest (false negative). In what follows, false positives are categorized into Error type 1 (Error_1) and false negatives are categorized as Error type 2 (Error_2).

Experiments have been carried out in a GPU-based platform. This platform is composed of:

- 2 hexa-core Intel Xeon E5-2650 at 2.20 GHz.
- 128 GB of RAM.
- Private L1 and L2 caches of 32 KB and 256 KB per node, and a L3 cache of 32 MB shared by all the cores of a socket.
- It includes an NVidia GTX 1080 Ti(Pascal), with 12 GB and 3584 cores (28 SM and 128 SP per SM).

The software environment is based on:

- gcc 7.4.0.
- Nvidia cuda 10.
- Python 3.6.5.
- The design of our LSTM model is based on Tensorflow 1.12.0 and Keras 2.2.4.

4. Results and discussion

This section shows the results obtained from the LSTM model designed to predict the temperature from IoT information. First, we show and analyze the results of the 3-cross validation. Then, we show the results of the test of 90% for train and 10% for test (considering 24 consecutive hours) for the LSTM technique. Finally, a comparison with two techniques for time series (ARIMA and Gaussian Process) is then made to evaluate and compare the performance of LSTM with other techniques.

Table 4

Mean results obtained after the execution of experiment of 3-fold cross validation. Being *RMSE* the Root Mean Square Error, *MAE* the Mean Absolute Error, *PCC* the Pearson Correlation Coefficient, and R^2, the determination coefficient

Dataset	Cieza	Moratalla	All
RMSE	0.9782	0.8133	1.0283
MAE	0.4067	0.4528	0.5287
PCC	0.9863	0.9912	0.9869
R^2	0.9725	0.9824	0.9740

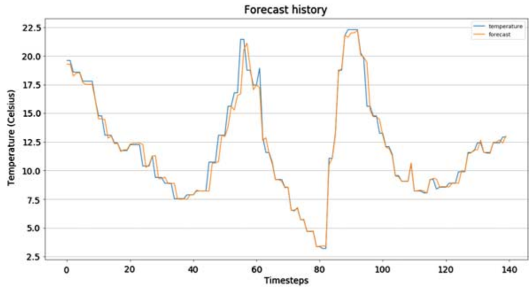
4.1. Cross validation test

In this experiment the adjustment of the LSTM is shown by means of a 3-fold cross validation. Thus, the 16,739 registers in Cieza, 17,070 registers in Moratalla, and the 33,809 registers in Cieza + Moratalla Dataset ("All Datasets") are randomly divided into 3 subsets of which two of them are trained and the third is performed the test, this is repeated until you have tested with the 3 subsets. Table 4 shows the mean results of the 3-fold cross validation experiment for the 3 Datasets. The results in terms of fit of the models are satisfactory, the value of R^2 indicates that the model created for each dataset, fits the real behavior of the data. Analyzing the models from the point of view of the RMSE and the MAE, we obtain an RMSE lower than a Celsius degree for the local models of Cieza and Moratalla. However, the global model obtains an RMSE higher than 1 degree Celsius. Analyzing the MAE, the results are similar to the RMSE, obtaining a better error the results created by the individual models, than by the global model. At a general level the results of all the models are satisfactory and we can affirm that the technique obtains a good behaviour for the prediction of the temperature.

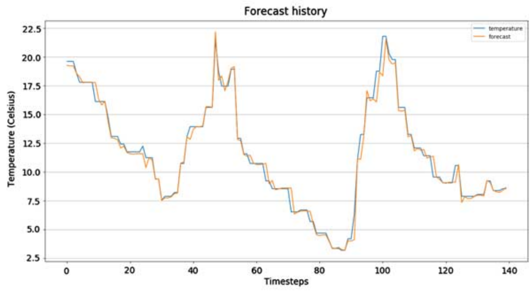
If in addition to analyzing the error, the trend of the models is analyzed, the goodness and fit of the model can be appreciated. Specifically, Figs 12(a), 12(b) and 12(c), show the first 200 instances of the test for each fold of the Cieza dataset. Only 200 instances are shown to be able to appreciate, the small errors that occur. The blue line is the actual temperature and the orange line is predicted by the LSTM model. It stands out as the higher the temperature the more errors occur, however the lower the temperature is, the better the prediction of the model is. This behavior for the frost problem is beneficial because the error in the estimation of low temperatures is more adjusted.

Same way, Figs 13(a), 13(b) and 13(c), show the first 200 instances of the test for each fold of the Moratalla dataset. The behavior is much more uniform in this dataset than in the Cieza datasets. Hence RMSE shown in Table 4 is lower. However, this translates into a greater error of the MAE, since it has a greater deviation in all temperature bands.

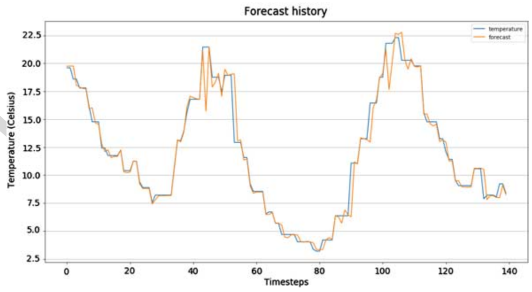
The global model of the "All" dataset is not shown graphically because it has a larger error and it is not interesting in terms of results.



a) Fold 1- Cieza cross validation experiments



b) Fold 2- Cieza cross validation experiments

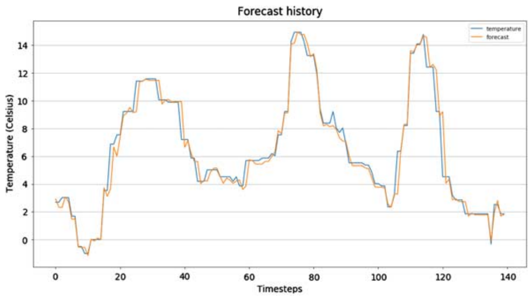


c) Fold 3- Cieza cross validation experiments

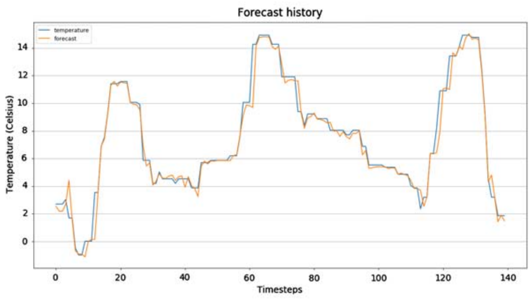
Fig. 12. Comparison between the prediction of the LSTM and the actual temperature of the first 200 instances for the Cieza and Moratalla datasets.

4.2. 24-hours temperature prediction

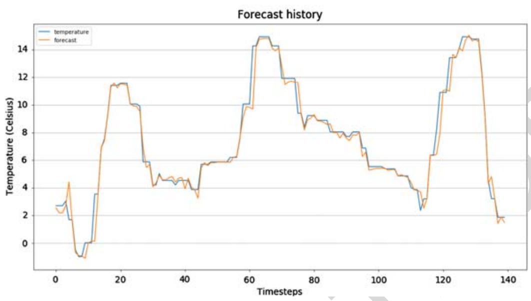
Once the robustness and reliability of the model have been analysed by means of cross validation, the model's capacity to predict 24 consecutive hours is analysed in this experiment. Specifically, to make the



a) Fold 1- Moratalla cross validation experiments



b) Fold 2- Moratalla cross validation experiments



c) Fold 3- Moratalla cross validation experiments

Fig. 13. Comparison between the prediction of the LSTM and the actual temperature of the first 200 instances for the Cieza and Moratalla datasets.

prediction we use full days of 24 hours, predicting values every 10 minutes, selecting the last available days of each dataset. Thus, the aim of this experiment is to measure the quality of the LSTM prediction whenever it tries to predict the temperature in a long period (i.e. 24 hours). To assess the LSTM model to predict 24 consecutive hours we use the RMSE, MAE, PCC and R^2. The values of this experiment are shown in the Table 5.

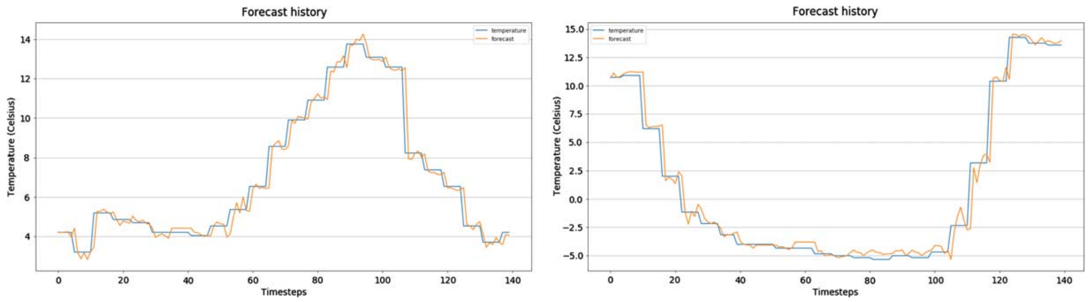
The results obtained for this experiment are similar to the previous general level. The RMSE obtains an average error of 0.64 Celsius degrees in Cieza dataset, and a little above a grade for Moratalla and "All" datasets. The fit of the model (R^2) is similar to the previous experiment. It is worth mentioning that the best results have been obtained with the models created by local temperatures. Although the differences are not too large, but in the Cieza area, the error difference is almost double, while having a local model the prediction is much more accurate. Figures 14 show the prediction of a full day with 144 records for the Cieza and Moratalla datasets, specifically the records correspond to January 3, 2019.

Table 5

Results obtained for the experiments of predicting 24 hours for different days. In the table *RMSE* is the Root Mean Square Error, *MAE* the Mean Absolute Error, *PCC* the Pearson Correlation Coefficient, and R^2, the determination coefficient

Datasets	Cieza	Moratalla	All
RMSE	0.6524	1.1147	1.2058
MAE	0.4089	0.6321	0.6703
PCC	0.9911	0.9869	0.9827
R^2	0.9820	0.9732	0.9648

In addition, this experiment carries out an in-depth analysis of the different types of errors produced. From the point of view of the frost problem, a system error predicting that it will not freeze and then that if the thermometer is actually lowered from zero, is a very serious error, as the farmer loses his/her entire crop. On the contrary, an error in the system indicating that it freezes and then in reality there is no such frost is less serious, because the farmer does not lose the crop simply loses the resources to apply the anti-freeze technique that economically these resources cost much less than losing the crop. To perform this in-depth analysis of the different types of errors, we will perform the classification of values to analyze false positives and false negatives. In the classification we consider cold ('C') when the temperature is lower than or equal to 0 and No-Cold ('NC') when the temperature is higher than 0. Using the prediction values made by the LSTM, we make the comparison between the real value and the predicted value and the classification comparison. Table 6 shows the classification results for the 3 datasets under study. The columns of Cieza, Moratalla and All show the percentage with respect to the total of the data, taking into account the comparison between the predicted by the model and the real value. Prediction column refers to the predictions made by our model, and the Real column correspond to the real data. For these last two columns, the different combinations have the following meaning:



a) Cieza Dataset

b) Moratalla Dataset

Fig. 14. Comparison between the LSTM prediction and the actual temperature of the 144 temperature measurements obtained on 3 January 2019 in Cieza (a) and in Moratalla (b), this corresponds to a full 24-hour day.

Table 6

Results obtained for the experiments of predicting 24 hours for different days, for the three datasets

Dataset		Cieza	Moratalla	All
Prediction	Real			
NC	NC	91.33%	52.41%	70.57%
NC	C	0.78%	1.80%	2.39%
C	NC	0.61%	1.13%	1.13%
C	C	7.26%	44.64%	25.89%
Successes		98.60%	97.06%	96.46%
Error		1.39%	2.93%	3.53%

1. *NC&NC* means that the prediction was *No Cold*, and the real data was *No Cold* too, this is considered as a success/hit in the prediction.
2. *NC&C* means that the prediction was *No Cold*, but the real value was *Cold*, therefore the model failed, and the type of error as defined at the beginning of this section is error type 2, *"$Error_2$"*.
3. *C&NC* means that the prediction was *Cold*, but the real value was *No Cold*, thus, the prediction was wrong. This is the *"$Error_1$"* kind of error.
4. *C&C* means that the prediction was *Cold* and the real data was the same, this is also considered as a success in the model.
5. *Success* row contains the sum of rows considered as success in the model, "NC&NC" and "C&C" rows where the forecast matches with the actual weather.
6. *Errors* contains the sum of rows considered as failures in the model, "C&NC" and "NC&C", that is, "$Error_1$" and "$Error_2$" respectively.

The results of Table 6 show the two types of errors for the Cieza dataset are very low (less than 1%). For the Moratalla dataset the error rises slightly. However, it is worth mentioning the $Error_2$ percentage, considered NC&C of the dataset "All". This type of error is very serious and the percentage of errors should be as low as possible since it causes the farmer to lose his/her crop, since the system indicates that there is no frost and finally there is a frost, so the farmer has not activated the anti-freeze techniques and the crop is lost. To analyze more in detail the types of errors, we show the confusion matrices in the Tables 7, 8, and 9. Table 7 shows the result for Cieza datasets, Table 8 shows the Moratalla and Table 9 shows "All" dataset respectively.

Table 7

Confusion matrix for Cieza dataset. Tested with 1143 registers

Cieza	C	NC
C	83	9
NC	7	1044

Table 8

Confusion matrix for Moratalla dataset. Tested with 1055 registers

Moratalla	C	NC
C	471	19
NC	12	553

Table 9

Confusion matrix for Cieza + Moratalla dataset. Tested with 2294 registers

Cieza + Moratalla	C	NC
C	594	26
NC	55	1619

Table 6, in Cieza datasets, successes are measured with 98.60%, this is the sum of "NC&NC" and "C&C", $91.33\% + 7.26\% = 98.60\%$. This value is detailed in Table 7, with 1044 (from NC&NC cell), and

83 (from C&C cell), $(1044+83)/1143 = 98.60\%$ too. This is important because with Table 7 now we can know the kind of success, in other words, we know that almost all the hits in Cieza were from the "NC&NC" side, maybe due to a warmer weather, 1044 against 83. For Moratalla dataset, the successes are 97.06%, this is the sum of "NC" and "C&C", 52.41% + 44.64%. The successes are more distributed between C&C and NC&NC, since it is a colder area and there are more instances with values equal to or less than zero. It should be noted that Error_2 percentage is twice as high as for Cieza datasets, but less than 2%. Finally, the datasets All has a very high percentage of errors and especially in Error_2. In addition to the Error percentage, we are going to analyze each of the confusion matrices of the 3 datasets, this matrices are shown in Tables 7, 8 and 9. In the first two tables Error_1 and Error_2, has almost the same probability, and in any case, the one that appears a little more is type one, which, as we already know is a bit less dangerous. The difference in the errors between Cieza and Moratalla is that Moratalla is a city with more hours of cold than Cieza hence the best accuracy obtained since the hours of cold are the most difficult to estimate because datasets are unbalanced in terms of number of cold hours. Finally, in the Table 9, we can observe that although there are some differences that we can see in Table 6, the kind of error is much more sensible in this case, because the most common type of error in this case is type 1, 55 cases versus 26, which is much more dangerous than in the previous datasets. Therefore, this global model gets a more negative result for the constraints of the proposed low temperature prediction problem.

As a summary of results, the LSTM technique works well and obtains satisfactory models for both local and global areas. However, taking into account the limitation of the data given by the unbalancing of the cold hours, the local models obtain more positive results, from the economic point of view for the farmer. This is given because the global models get a higher percentage of Error_2 which are the errors, where the model indicates that there is no frost and finally a frost occurs, this causes the farmer to lose his/her crop, while the other type of error, the farmer only loses the resources of the antifrost techniques that are much smaller. Thus, after analyzing the results, it is recommended to deploy an IoT system per area with a single temperature sensor, lowering the cost of that system.

4.3. *Comparing LSTM with other techniques*

This section compares our LSTM approach with two traditional methods that have been widely used in the literature to predict values in a time series; they are Auto Regressive Integrated Moving Average (ARIMA) [8] and Gaussian process [6]. In ARIMA, the autoregressive part indicates that the variable of interest in evolution is regressed based on its previous values. The moving average part shows that the regression error is actually a linear combination of error terms whose values occurred simultaneously and at various times in the past. Integrated means that the data values have been replaced with the difference between their values and the previous values, and this process could be performed more than once. ARIMA is usually applied in cases where the information does not have stationarity. The Gaussian process, however, is a stochastic process based on a collection of random variables indexed by time or space, such that each finite collection of those random variables has a normal multivariate distribution, that is, each finite linear combination of them is normally distributed.

The validation carried out for all the experiments has been a 90% of instances for train and 10% of instances for test was used, as in the previous experiment.

Table 10 the comparison of the results between the proposed LSTM model, the ARIMA and Gaussian Process techniques for air temperature. Three metrics have been used to perform this comparison, specifically root-mean-square error (RMSE), Mean Absolute Error (MAE) and the coefficient of determination (R^2). Analyzing these results from the point of view of model fit, (R^2), LSTM and ARIMA techniques have

Table 10

Comparison of the LSTM technique with the ARIMA time series techniques and the Gaussian Process (G. Process), showing for each value *RMSE* is the Root Mean Square Error, *MAE* the Mean Absolute Error and R^2, the determination coefficient

Techniques	Datasets	Cieza	Moratalla	All
LSTM	RMSE	0.6524	1.1147	1.2
	MAE	0.4089	0.6321	0.6703
	R^2	0.982	0.9732	0.9648
ARIMA	RMSE	1.0352	1.3512	1.5631
	MAE	0.5935	0.6985	0.8521
	R^2	0.9436	0.9423	0.9321
G. Process	RMSE	2.3622	2.6859	3.0123
	MAE	1.2305	1.3625	1.6589
	R^2	0.7236	0.7365	0.7025

satisfactory results and obtain competent models for the 3 datasets. However the Gaussian process has a less stable behaviour and its model is worse than the other two. On the other hand, if we analyze the result from the point of view of the error, both of the RMSE and of the MAE, the best result is obtained by the LSTM model for the 3 datasets, being the difference almost of 0.5°C for some datasets in the RMSE. Therefore, after the comparison, we can conclude that the model that best fits the problem of predicting temperature, using a univariate model, is the LSTM model.

5. Conclusions and future work

The prediction of low temperatures is a latent problem for farmers, as climate change is causing abrupt temperature changes even in warm areas. Knowing in advance whether or not low temperatures will occur helps the farmer to foresee resources and apply anti-frost control techniques early enough to ensure maximum effectiveness. In this work an a predictive model for temperature has been analyzed and constructed using a Long short-term memory neural network. The LSTM has been tested using actual temperature data provided by an IoT system deployed on two areas of Region of Murcia (Spain), also a global model has been tested. The models obtained by the LSTM for temperature prediction will be part of the intelligent component of the IoT system distinguishing the type of zone, as it has been found that local models are more accurate than the global model. The two experiments carried out to test the proposed model have obtained similar results and with quite satisfactory goodness obtaining about 96-98% model fit and an RMSE of less than 1 Celsius degrees. However, from the point of view of the errors made, specifically the Error_2, which means that the system predicts no frost and in reality there is frost, the global model gets a higher percentage of them and local models less error in general and less Error_2. Therefore, the best model to implement the IoT system prediction is the local model in each area. This model also indicates that the IoT system can be composed only of the temperature sensor, as the results obtained are acceptable. Furthermore, in the comparison with the two traditional techniques, it can be seen that the proposed LSTM model obtains a better performance, followed by the ARIMA technique, although the latter obtains a good adjustment of the model but a higher temperature prediction error.

As future work, new variables will be incorporated into the LSTM to create a multivariate LSTM with our local IoT infrastructure and study the influence of other variables on temperature prediction. Moreover, the computational side of deep learning models here developed will be analyzed in terms of performance and power-consumption. This will enable edge computing platforms to execute our local models. Finally, a new machine learning technique based on K nearest neighbours, that maintains the temporality of the data, will be developed to compare with the current results.

Acknowledgements

This work was partially supported by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20813/PI/18, and by Spanish Ministry of Science, Innovation and Universities under grants TIN2016-78799-P (AEI/FEDER, UE) and RTC-2017-6389-5.

References

[1] H.S. Abdullahi, R. Sheriff and F. Mahieddine, Convolution neural network in precision agriculture for plant image recognition and classification, in: *2017 Seventh International Conference on Innovative Computing Technology (Intech)*, IEEE, 2017, pp. 1–3. doi:10.1109/INTECH.2017.8102436.

[2] K.A. Al-Gaadi, A.A. Hassaballa, E. Tola, A.G. Kayad, R. Madugundu, B. Alblewi and F. Assiri, Prediction of potato crop yield using precision agriculture techniques, *PLoS ONE* **11**(9) (2016), e0162219.

[3] K.M. Barlow, B.P. Christy, G.J. O'Leary, P.A. Riffkin and J.G. Nuttall, Simulating the impact of extreme heat and frost events on wheat crop production: A review, *Field Crops Research* **171** (2015), 109–119. doi:10.1016/j.fcr.2014.11.010.

[4] M.R. Bendre, R.C. Thool and V.R. Thool, Big data in precision agriculture: Weather forecasting for future farming, in: *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, IEEE, 2015, pp. 744–750. doi:10.1109/NGCT.2015.7375220.

[5] C. Bergmeir and J.M. Benítez, On the use of cross-validation for time series predictor evaluation, *Information Sciences* **191** (2012), 192–213. doi:10.1016/j.ins.2011.12.028.

[6] S. Brahim-Belhouari and A. Bermak, Gaussian process for nonstationary time series prediction, *Computational Statistics & Data Analysis* **47**(4) (2004), 705–712. doi:10.1016/j.csda.2004.02.006.

[7] D. Bretreger, J. Quijano, J. Awad et al., Monitoring irrigation volumes using climate data and remote sensing observations, in: *Hydrology and Water Resources Symposium (HWRS 2018): Water and Communities*, Engineers Australia, 2018, pp. 112–123.

[8] P.J. Brockwell and R.A. Davis, *Introduction to Time Series and Forecasting*, Springer, 2016. doi:10.1007/978-3-319-29854-2.

[9] S.W. Chen, S.S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C.J. Taylor and V. Kumar, Counting apples and oranges with deep learning: A data-driven approach, *IEEE Robotics and Automation Letters* **2**(2) (2017), 781–788. doi:10.1109/LRA.2017.2651944.

[10] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016.

[11] K. Gregor, I. Danihelka, A. Graves and D. Wierstra, DRAW: A recurrent neural network for image generation, 2015. arXiv:1502.04623.

[12] M.Á. Guillén-Navarro, R. Martínez-España, B. López and J.M. Cecilia, A high performance IoT solution to reduce frost damages in stone fruits, *Concurrency and Computation: Practice and Experience* (2019). doi:10.1002/cpe.5299.

[13] M.Á. Guillén-Navarro, F. Pereñíguez-García and R. Martínez-España, IoT-based system to forecast crop frost, in: *2017 International Conference on Intelligent Environments (IE)*, IEEE, 2017, pp. 28–35. doi:10.1109/IE.2017.38.

[14] S.A. Haider, S.R. Naqvi, T. Akram, G.A. Umar, A. Shahzad, M.R. Sial, S. Khaliq and M. Kamran, LSTM neural network based forecasting model for wheat production in Pakistan, *Agronomy* **9**(2) (2019), 72. doi:10.3390/agronomy9020072.

[15] A.N. Harun, M.R.M. Kassim, I. Mat and S.S. Ramli, Precision irrigation using wireless sensor network, in: *2015 International Conference on Smart Sensors and Application (ICSSA)*, IEEE, 2015, pp. 71–75. doi:10.1109/ICSSA.2015.7322513.

[16] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation* **9**(8) (1997), 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[17] S. Hochreiter and J. Schmidhuber, LSTM can solve hard long time lag problems, in: *Advances in Neural Information Processing Systems*, 1997, pp. 473–479.

[18] R. Huang, C. Zhang, J. Huang, D. Zhu, L. Wang and J. Liu, Mapping of daily mean air temperature in agricultural regions using daytime and nighttime land surface temperatures derived from TERRA and AQUA MODIS data, *Remote Sensing* **7**(7) (2015), 8728–8756. doi:10.3390/rs70708728.

[19] A. Kamilaris and F.X. Prenafeta-Boldu, Deep learning in agriculture: A survey, *Computers and Electronics in Agriculture* **147** (2018), 70–90. doi:10.1016/j.compag.2018.02.016.

[20] F. Kratzert, D. Klotz, C. Brenner, K. Schulz and M. Herrnegger, Rainfall-runoff modelling using long-short-term-memory (LSTM) networks, *Hydrol. Earth Syst. Sci. Discuss.* (2018). doi:10.5194/hess-2018-247.

[21] A.-K. Mahlein, Plant disease detection by imaging sensors – Parallels and specific demands for precision agriculture and plant phenotyping, *Plant Disease* **100**(2) (2016), 241–251. doi:10.1094/PDIS-03-15-0340-FE.

[22] A. Milioto, P. Lottes and C. Stachniss, Real-time blob-wise sugar beets vs weeds classification for monitoring fields using convolutional neural networks, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **4** (2017), 41–48. doi:10.5194/isprs-annals-IV-2-W3-41-2017.

[23] K. Mohammadi, S. Shamshirband, S. Motamedi, D. Petković, R. Hashim and M. Gocic, Extreme learning machine based prediction of daily dew point temperature, *Computers and Electronics in Agriculture* **117** (2015), 214–225. doi:10.1016/j.compag.2015.08.008.

[24] X.E. Pantazi, D. Moshou, T. Alexandridis, R.L. Whetton and A.M. Mouazen, Wheat yield prediction using machine learning and advanced sensing techniques, *Computers and Electronics in Agriculture* **121** (2016), 57–65. doi:10.1016/j.compag.2015.11.018.

[25] E. Pierpaoli, G. Carli, E. Pignatti and M. Canavari, Drivers of precision agriculture technologies adoption: A literature review, *Procedia Technology* **8** (2013), 61–69. doi:10.1016/j.protcy.2013.11.010.

[26] J.D. Pujari, R. Yakkundimath and A.S. Byadgi, Image processing based detection of fungal diseases in plants, *Procedia Computer Science* **46** (2015), 1802–1808. doi:10.1016/j.procs.2015.02.137.

[27] P.P. Ray, Internet of things for smart agriculture: Technologies, practices and future direction, *Journal of Ambient Intelligence and Smart Environments* **9**(4) (2017), 395–420. doi:10.3233/AIS-170440.

[28] A.G. Salman, Y. Heryadi, E. Abdurahman and W. Suparta, Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting, *Procedia Computer Science* **135** (2018), 89–98. doi:10.1016/j.procs.2018.08.153.

[29] J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Networks* **61** (2015), 85–117. doi:10.1016/j.neunet.2014.09.003.

[30] Z. Shareef and S.R.N. Reddy, Wireless sensor network for aquaculture: Review, survey, and case study of aquaculture practices in western Godavari region, *Journal of Ambient Intelligence and Smart Environments* **10**(5) (2018), 409–423. doi:10.3233/AIS-180498.

[31] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk and D. Stefanovic, Deep neural networks based recognition of plant diseases by leaf image classification, *Computational Intelligence and Neuroscience* **2016** (2016), 3289801. doi:10.1155/2016/3289801.

[32] X.-Y. Wang, W.-X. Liao, D. An and Y.-G. Wei, Maize haploid identification via LSTM-CNN and hyperspectral imaging technology, Preprint, 2018. arXiv:1805.09105.

[33] S. Wolfert, L. Ge, C. Verdouw and M.-J. Bogaardt, Big data in smart farming – A review, *Agricultural Systems* **153** (2017), 69–80. doi:10.1016/j.agsy.2017.01.023.

[34] K. Xu, J. Ba, R. Kiros, K. Cho, A.C. Courville, R. Salakhutdinov, R.S. Zemel and Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, 2015. arXiv:1502.03044.

[35] M.A. Zaytar and C. El Amrani, Sequence to sequence weather forecasting with long short-term memory recurrent neural networks, *International Journal of Computer Applications* **143**(11) (2016), 7–11. doi:10.5120/ijca2016910497.

[36] C. Zhang and J.M. Kovacs, The application of small unmanned aerial systems for precision agriculture: A review, *Precision Agriculture* **13**(6) (2012), 693–712. doi:10.1007/s11119-012-9274-5.

[37] J. Zhang, Y. Zhu, X. Zhang, M. Ye and J. Yang, Developing a long short-term memory (LSTM) based model for predicting water table depth in agricultural areas, *Journal of Hydrology* **561** (2018), 918–929. doi:10.1016/j.jhydrol.2018.04.065.

[38] N. Zhang, M. Wang and N. Wang, Precision agriculture a worldwide overview, *Computers and Electronics in Agriculture* **36**(2–3) (2002), 113–132. doi:10.1016/S0168-1699(02)00096-0.

2.4 Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning

Título	*Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning*
Autores	Miguel Ángel Guillén-Navarro, Antonio Llanes, Baldomero Imbernón, Raquel Martínez-España, Andrés Bueno-Crespo, Juan Carlos Cano y José María Cecilia
Revista	Journal of Supercomputing
Año	2021
Estado	Publicado

Contribución del Doctorando
Miguel Ángel Guillén Navarro, declara ser el principal autor y el principal contribuidor del artículo *Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning*.

The Journal of Supercomputing
https://doi.org/10.1007/s11227-020-03288-w

Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning

Miguel A. Guillén[1] · Antonio Llanes[1] · Baldomero Imbernón[1] · Raquel Martínez-España[1] · Andrés Bueno-Crespo[1] · Juan-Carlos Cano[2] · José M. Cecilia[2]

Abstract
The Internet of Things (IoT) is driving the digital revolution. AlSome palliative measures aremost all economic sectors are becoming "Smart" thanks to the analysis of data generated by IoT. This analysis is carried out by advance artificial intelligence (AI) techniques that provide insights never before imagined. The combination of both IoT and AI is giving rise to an emerging trend, called AIoT, which is opening up new paths to bring digitization into the new era. However, there is still a big gap between AI and IoT, which is basically in the computational power required by the former and the lack of computational resources offered by the latter. This is particularly true in rural IoT environments where the lack of connectivity (or low-bandwidth connections) and power supply forces the search for "efficient" alternatives to provide computational resources to IoT infrastructures without increasing power consumption. In this paper, we explore edge computing as a solution for bridging the gaps between AI and IoT in rural environment. We evaluate the training and inference stages of a deep-learning-based precision agriculture application for frost prediction in modern Nvidia Jetson AGX Xavier in terms of performance and power consumption. Our experimental results reveal that cloud approaches are still a long way off in terms of performance, but the inclusion of GPUs in edge devices offers new opportunities for those scenarios where connectivity is still a challenge.

Keywords Edge computing · LSTM · Deep learning · Precision Agriculture

✉ José M. Cecilia
jmcecilia@disca.upv.es

Extended author information available on the last page of the article

Published online: 29 April 2020

1 Introduction

Smart agriculture is an emerging field where the concepts, techniques and systems of Industry 4.0 are applied to the agrarian world [10]. The combination of both IoT and AI provides sustainable procedures to optimize crops, reduce the use of pesticides, optimize irrigation and, in general, avoid the most egregious problems in agricultural processes [6, 28, 29, 38]. Of particular interest to us is the prevention of frost. In Mediterranean areas, low temperatures at certain times of the agricultural cycle are a major problem that can result in losses of millions of euros.[1] However, it is not an easy task as it depends on several factors, including temperature, humidity, wind speed, etc. [30], but also on the particular location of the plot. Actually, the global weather forecast provides coarse grain information which is not valid to predict frost at the plot level.

Some palliative measures are available to avoid crop loss, such as connecting windmills and stoves or connecting heating in a greenhouse to avoid crop losses. However, these measures need to be activated some time in advance, between 2 and 4 h, to be really effective. Therefore, this is a scenario where both accuracy and performance are equally critical. A false positive, i.e., the application predicts a frost but eventually does not occur, means an economic and environmental impact, as these techniques are often too costly in both terms. A false negative, i.e., the application does not predict a frost but eventually occurs, can mean the loss of the crop which usually has dramatic consequences. Equally important is to predict the frost early enough in order to be able to take palliative actions, because otherwise, the consequences would be the same.

Deep Learning is a area within machine learning which relies on a set of artificial neural networks organized as complex hierarchical levels [11]. Deep learning models are applied in agriculture to deal with problems such as the automatic identification of plant disease through images or yield predictions in crops [19]. In the field of deep learning, Long short-term memory (LSTM) [18] is a recurrent neural network (RNN), first proposed by Hochreiter and Schmidhuber, which has feedback connections. LSTM goes beyond processing single data points such as images, but also entire sequences of data such as speech, video or time-series data in general. Thus, they are well suited to classifying, processing and making predictions based on data capture from IoT infrastructures [26].

In our previous work, an IoT system was proposed to obtain fine-grained information from a particular plot area [15] and some prepossessing tasks were carried out to remove outliers and out-of-range values. Moreover, we develop a deep learning model for the frost prediction based on a long short-term memory (LSTM) with satisfactory accuracy results, i.e., an average quadratic error of less than a Celsius degree and a determination coefficient R^2 greater than 0.95 [14]. However, the computational cost of this model was too high and therefore it should be executed offline. This limits the success of this technique of because several reasons. The IoT

[1] https://www.laverdad.es/murcia/ultimas-heladas-region-20190404113101-nt.html.

infrastructure should be connected to the Internet with high-speed connection or, at least, with a minimum number of cloud outages. This is actually not the scenario in rural areas where IoT infrastructures are often found in inhospitable conditions, where high temperatures, lack of connectivity and security are just a few examples that limit the existence of responsive Web services. With a low-bandwidth connection plus the training and inference time of the LSTM model in the cloud, the maximum prediction window allowed for decision making (i.e., 2–3 h) would be difficult to achieve.

Edge computing [32] is an emerging area where processing in close proximity to mobile devices or sensors may provide energy savings, highly responsive Web services for mobile computing, scalability and privacy-policy enforcement for the Internet of Things, as well as the ability to mask transient cloud outages. Indeed, edge computing platforms are designed to be energy efficient and therefore performance is not their primary objective. Some computing platforms are emerging to enable edge computing with a reduced power budget and ever-increasing performance. Among them, we may highlight Nvidia Jetson family which can run between 7.5 and 10 W of power, offering a good performance ratio [17]. In this article, we provide a performance and energy evaluation of edge vs. cloud computing platforms for a LSTM deep learning model to predict the possibility of frost in crops, taking as input data captured from an IoT system. The main contributions of this paper include the following:

1. The LSTM model for frost prediction previously published in [14] has been adapted to be executed on high-end and low-power GPUs.
2. CPU and GPU code is evaluated in terms of performance and power on the Nvidia Jetson AGX Xavier to find out whether the inclusion of GPUs on the edge devices is a compelling alternative for running heavy workloads like the LSTM model. The GPU-based code of out LSTM model shows 1.6x speedup factor compared to multicore counterpart version.
3. We also compare the edge solution with its cloud-based counterpart for both stages: training and inference, showing that Nvidia Jetson may offer enough computational horsepower to create an autonomous decision support systems for frost prediction. Training could be developed during the warmest part of the day when frost is unlikely to occur and inference can be made from midday.
4. The quality of the results obtained is also evaluated to ensure that our version of the GPU draws similar conclusions to its CPU counterpart.

The rest of the paper is structured as follows. Section 2 shows the necessary background to better understand our proposal. Section 3 introduces the deep learning model based on LSTM to predict temperature. Section 4 includes the empirical results in which both quality and performance evaluations are presented. Section 5 summarizes the conclusions and gives some directions for future work.

2 Background

2.1 Edge versus cloud computing

Since the early days, computing has alternated between centralization and decentralization. At the beginning, batch processing and time-sharing prevailed in a centralized fashion. With the advent of personal computing in the 1980s, we then move to a decentralized approach that was centralized again in the mid-2000s through the cloud. Nowadays, cloud computing is established as the most obvious infrastructure to leverage from a mobile device. However, the optimal cloud infrastructure may be too far away from the mobile device. Li et al. show the average round trip time from 260 global vantage points to their optimal Amazon EC2 instances was 74 ms [24]. In addition, the latency of the wireless first hop should be added. Therefore, in some emergent applications, this latency is not tolerable and some authors pointed out the necessity of moving again toward distribution. Satyanarayanan et al. [32] propose two-level architecture to look for mobile applications' interactive performance. The first level was the unmodified cloud computing architecture, and a second level was a network of dispersed elements called cloudlets with state cached from the first level [33]. Moreover, Bonomi et al, motivated by IoT infrastructure scalability instead, introduced the term fog computing that consists again of a multilevel hierarchy of fog nodes spanning from the cloud to IoT edge devices [4]. As stated in [32], the proximity of cloudlets (or fog nodes) provides different benefits but of particular interest to us are two of them:

1. Highly responsive cloud services, achieving low end-to-end latency, high bandwidth and low jitter to services located on the edge. Edge computing brings, through cloudlets, computational power within one wireless hop of sensors or mobile devices. Indeed, applications that are both latency-sensitive and computation-intensive would not become possible without this technology. Interactive mobile applications such as virtual or augmented are leading examples.
2. Scalability via edge analytics, lowering the bandwidth required by high-data-rate IoT sensors processing applications. Reducing the amount of information to be transferred to the cloud through a data analytics on the edge may avoid network overloading and may provide energy saves. A leading example of this is GigaSight framework [35] where video from mobile devices only goes to the nearby cloudlet. The cloudlet runs computer vision application and sends the results and some metadata to the cloud, reducing drastically the application bandwidth.

Edge computing is a compelling alternative to enable smart environments in agriculture ubiquitously. Actually, agriculture procedures are usually developed in rural areas where several technological challenges often cohabit. Among them, we may highlight technical issues such as bandwidth, connectivity, power supply and scaling sensors but also environmental issues that complicate the deployment of IoT, such as extreme weather conditions. Some recent works have carried out studies considering edge computing for smart agriculture or farming. For instance, Zamora-Izquierdo et al. [36] proposes a platform to deal with soilless culture needs in full recirculation greenhouses using moderately saline water. Edge computing here is in charge of monitoring and managing main PA tasks near the access network to increase system reliability against network access failures. However, they do not introduce computationally heavy workloads such as deep learning models at this level as they require higher computational horsepower. Singh, Chana and Buya propose Agri-Info, a cloud-based autonomic system for delivering agriculture as a service. They do provide intelligent procedures to diagnose the agriculture status automatically using fuzzy logic but they execute so in the cloud, offering a mobile interface to see the output. Indeed, this is a good approach as long as high connectivity is guaranteed. Moreover, they do not use information from the ground; they use expert knowledge taken from their mobile or tablets where connectivity is somehow guaranteed. Finally, a survey on the role of IoT in agriculture for the implementation of smart farming is provided in [9]. They clearly state that IoT-based solutions are at the forefront of automatically maintaining and monitoring agricultural farms with minimal human involvement. Indeed, this requires the involvement of AI models that can provide predictions in real time.

2.2 Deep learning in precision agriculture

Deep learning is made up of a set of techniques that allow the construction of models with great potential and outstanding results [19]. Recurrent neural network (RNN) architectures have been widely applied for regression [20]. RNNs are a type of artificial neural networks (ANNs) in which connections between nodes create a graph directed over a time sequence, allowing to have a temporal dynamic behavior. The main difference between RNNs and other ANNs architectures is that they use their internal state (memory) to process input sequences. RNNs work well in problems with short-term dependencies where the model only needs to look at recent (or close) information to carry out the current task [12]. However, there are some problems that require less recent or more distant information to perform the current task. This is called long-term dependency, and RNNs do not work well in such scenarios. The long short-term memory (LSTM) is type of RNNs, which is designed to deal with long-term dependencies such as those within time series based on learning from previous observations to predict the next value in the sequence [18]. While Standard RNNs are based on chain of very simple ANNs, such as a single tanh layer, LSTMs are also based on chain of ANNs, but the ANNs has a different structure.

Deep learning models have drastically improved the state of the art in many sectors of industry including agriculture [25]. In the field of agriculture, different deep learning techniques (mainly convolutional neural networks for image classification and LSTM for the prediction of temporal variables) are applied to multiple problems such as plant diseases [27], the classification of plant species [13], identification of soil cover and classification of crop type [22], estimation of yields [23], identification of weeds [3], predictions on climatology [31], just to mention a few recent studies within the umbrella of precision agriculture.

Focusing on the problem of climatology, some works have been published for climatic prediction, showing very good results. For instance, Salman et al. [31] propose the construction of a robust statistical model to predict meteorological visibility based on other intermediate variables (temperature, pressure, humidity and dew point). They use two- and four-layer LSTM networks. The data are normalized and rescaled in the range of [0, 1] and using a moving average. Here, the multilayered LSTM model was most effective. However, this work was not applied to precision agriculture but for forecasting visibility based on temperature, pressure, humidity, dew point in airport area context. LSTM was also used to predict climate variables in [37]. The targeted variables used as an input were temperature, humidity and wind speed. In this case, the network architecture consists of two LSTM layers with the activation function being the RELU and the optimizer RMSProp. The data were also normalized and rescaled in the range of [−1, 1]. The results were generated for a coarse-grained prediction in cities. Again, it is not applied to precision agriculture context and therefore data were obtained for historical datasets instead of real-time information like us. Kratzert et al. [21] proposed a model for rain and runoff also using LSTM, which predicts discharge for a variety of watersheds. The authors demonstrate the potential of this method by using some variables such as day length, rainfall, temperature or humidity. Here, the LSTM is composed of two LSTM layers, and between them is a dropout layer to avoid overtraining of the network but in the context of water management scheduling not in precision agriculture.

Finally, we have explored the bibliography of combining deep learning methods with edge computing architectures. Chen et al. [7] propose ThriftyEdge, a resource-efficient edge computing for intelligent IoT applications, and one of the case studies is precision agriculture. However, this work is not tailored to deep learning-based applications. Boubin et al. [5] explore fully autonomous precision agriculture where fully autonomous aerial systems (FAAS) map crop fields frequently. They develop the full-stack architecture, including a software driven by reinforcement learning and ensemble models. The early results presented in this paper do not show any relevant results in the field of edge computing. Actually, they work at simulation level, running their experiments in a Laptop. Finally, we refer the reader to [39] for a review in *Edge Intelligence*, where efforts for bridging the gaps between deep learning and edge computing are summarized. Even though the area of precision agriculture is cited as one of the main domains for edge intelligence, it is worth highlighting that any work is cited for this domain.

3 LSTM model for temperature prediction on IoT infrastructures

This section introduces the proposed LSTM model for the temperature prediction in IoT infrastructures. First of all, we briefly introduce the IoT infrastructure to capture information from a plot that was previously introduced in [15]. Next, the proposed LSTM model for the prediction of temperature is explained in depth.

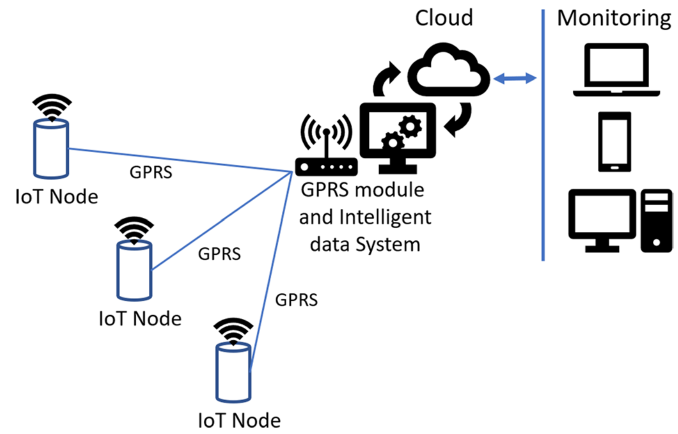


Fig. 1 IoT system architecture

3.1 The IoT infrastructure

Figure 1 shows the IoT architecture previously presented in [15]. This infrastructure is deployed in two different crops at Murcia (South East Spain). The system monitors the hydro-climatological information of a plot, and depending on the data and models developed, the system can alert farmers to take appropriate action if necessary. The infrastructure is based on three main components: (1) hydro-climatological sensors, (2) an intelligent data processing system and (3) a monitoring component. The intelligent data processing system only performs a preprocessing of outliers values to avoid showing incorrect values to the farmer.

In this article, we are evaluating the possibility of introducing the computation of intelligent algorithm at the edge of the network. Therefore, the intelligent data processing system would be physically at the edge or in the cloud, depending on the requirements set by the users. The algorithm will allow farmers to obtain a temperature prediction in real time at their plots.

The workflow begins at the sensor level in which the climate data are captured. The sensor network is the 4H remote control system of Hidroconta.[2] Figure 2 shows the IoT node composed of temperature, humidity and wind speed sensors. They are connected to the analog inputs available on the IoT node. In our case, we only use the information provided by the temperature sensor. Sensors provide information every 10 min.

The IoT node has connectivity capabilities. For this study, there are three IoT nodes where one of them is acting as a gateway to send data to the cloud via GPRS connection. The slave nodes are connected to the gateway using LoRa technology.[3] Moreover, all of them have a 6 volts (V) and 12 amp (A) battery and a 12V and 5 watts solar panel. Finally, each IoT node has a microcontroller with 256 KB of

[2] http://www.hidroconta.com/.

[3] https://lora-alliance.org/.

Fig. 2 IoT node (we refer the reader to [15] for insights)

firmware storage and 96 KB of volatile memory for data, and it is able to store up to 20.000 records. Therefore, the IoT nodes store the information in their internal memory for fault tolerance.

Figure 3 shows the different computing schemes under evaluation for air temperature prediction using LSTM deep learning model. On the one hand, a centralized computing system has been used where the IoT nodes have sent the temperature by means of GPRS to the centralized module, in which an outliers cleaning has been carried out to be able to estimate the temperature using the LSTM. Also, this module allows to visualize the data (3a). The second scheme presents the computational component within the IoT module. In this way, the LSTM is executed locally, which allows temperature prediction without the need to send the information to a centralized module. Finally, the prediction is sent by GPRS to a module that allows data to be displayed (3b).

3.2 The LSTM model for temperature prediction

Analysis of sensor data from IoT infrastructures as the one described above can provide valuable information for creating new applications to deal with the emergent problems that are currently unsolved. We are particularly interested in air temperature modeling for early identification of frost on crops. In our previous work, we designed a deep learning model, which offered very good results in the prediction of this variable. In what follows, we introduce this model and refer the reader to [14] for insights.

In the Mediterranean area, there is a large diurnal temperature variation, which implies that the data time series of air temperature is nonlinear, containing sensitive information, making very difficult to predict with traditional autoregressive (AR) models such as ARIMA [2]. As previously explained in Sect. 2, the LSTM is type of RNNs which is designed to deal with long-term dependencies, and therefore, it has shown very good results for predicting values in time series. LSTM is usually composed of four layers, interacting in several ways, and the key idea of LSTMs is the

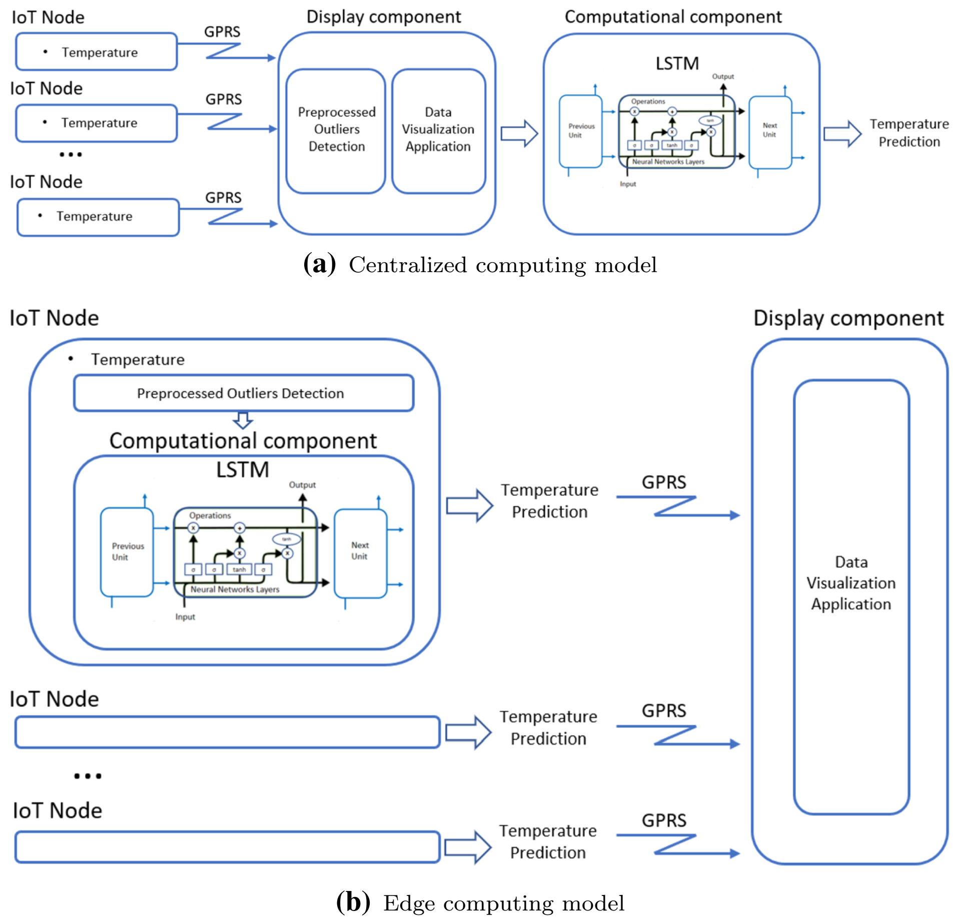


Fig. 3 Two computing models are presented; **a** a cloud-based approach where the computation is carried out in the backend and **b** an edge computing approach where the computation is carried out in the IoT node

cell state which is basically like a conveyor belt. The information runs through the entire chain, with only some minor linear interactions. The LSTM can add or remove information to the cell state through the *Gates*. They are an optional way to let information pass and are based on a *sigmoid neural network* layer and a point multiplication operation. The sigmoid layer produces numbers between zero and one, describing how much of each component must pass. An LSTM has three of these gates in order to protect and control the cell state. Figure 4 shows the interaction of the three gates in detail, as well as how the LSTM modules exchanges information with each other. An LSTM module receives the temperature at time t as input, which is processed together with the temperature prediction at time $t-1$, received from the previous LSTM module. Both temperatures are processed and sent to the next module, which also receives the temperature at $t+1$. As an example, let us assume that the LSTM receives the temperature at the instant t, where $t = 3.41$ °C. After processing this temperature, which is now considered as the previous temperature $(t-1)$, it

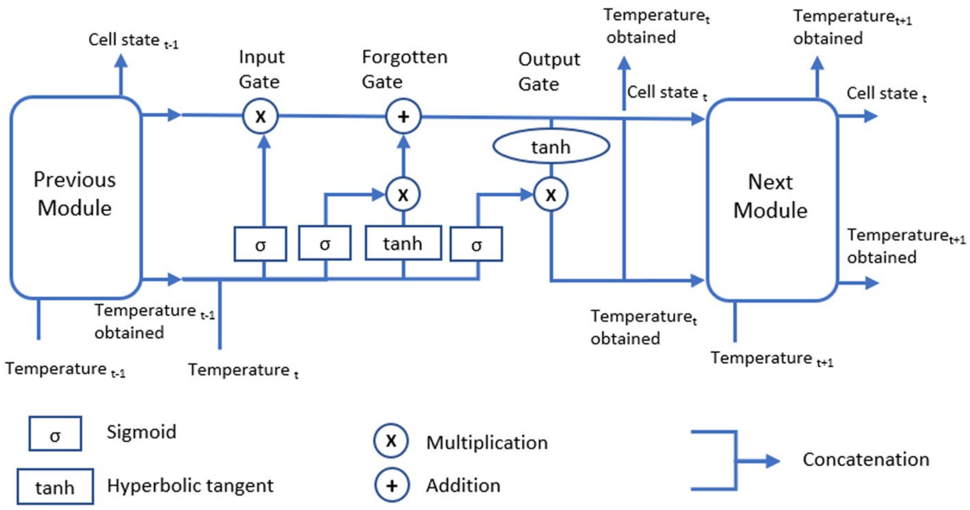


Fig. 4 LSTM scheme developed to deal with the frost prediction. Each LSTM module interacts with the previous and subsequent one. The schema shows the three gates that compose each LSTM module, which allow it to remember or delete the information that passes through them

returns as output the temperature at the instant $t + 1$, let us assume the temperature $t + 1 = 3.05$ °C, that is, the output of the temperature for the next 10 min interval would be 3.05 °C.

The first step in our LSTM is to decide the input layer. This is an importance decision as it decides what information goes through the cell state. In our case, it is a sigmoid layer, which receives as input the air temperature taken from the sensors (see Fig. 3a). Moreover, a hidden stacked LSTM layer that is made by LSTM blocks contains four interacting layers. These four neuronal layers are made up of three sigmoids and a hyperbolic tangent, which have been tested from 50 to 200 neurons. So in each step, the LSTM block has to decide what information will be stored in the state of the cell. To do this, the two types of neural networks work differently: the neural layers with sigmoids decide which values of air temperatures have to be updated, while the neuronal layer of the hyperbolic tangent creates a vector of new values of candidate air temperatures that could be added to the state. The two processes are combined to create a status update. Finally, the output layer gives the predicted air temperature based on previous air temperatures. The parameters used have been optimized, such as learning factor (0.001), optimizer (Adam), activation function (hyperbolic tangent), etc. A more detailed explanation of the parameters (batch size, number of epochs, number of delay sequences...) used for the experiments is shown in Sect. 4.1.1.

3.3 CPU and GPU implementation

Two different implementations of the LSTM model previously described have been developed for this article. Both of them are based on Keras (version 2.2.4) [16]. Keras is a Python-based open-source neural network library, which is capable of running on top of many neural network frameworks such as Tensorflow [1] or CNTK [34], just to mention a few. Keras contains implementations of commonly used neural networks, including convolutional and RNN neural networks. Moreover, it includes the main building blocks such as objectives, optimizers, layers, activation functions.

Algorithm 1 LSTM model implementation with Keras on CPU and GPU

```
//Create a model
model = Sequential()
if gpuMode then
   //GPU
   for i IN range(stack_lstm_layers − 1) do
      model.add(CuDNNLSTM(neuron, return_sequences = True))
   end for
   //Last LSTM layer
   model.add(CuDNNLSTM(neuron)) {Last LSTM layer}
else
   //CPU
   for i IN range(stack_lstm_layers − 1) do
      model.add(LSTM(neuron, return_sequences = True, activation = activation))
   end for
   //Last LSTM layer
   model.add(LSTM(neuron, activation=activation))
end if
//Last layer of the model
model.add(Dense(output))
//Compile the model for training.
model.compile(loss=loss, optimizer=optimizer)
```

Algorithm 1 shows the code baselines of our LSTM implementation. First, we create the model as a linear stack of layers with the *sequential* function from Keras. Then, different layers are added to the model, depending on an input parameter which is actually defined by the user. The influence of this parameter in terms of performance will be studied in Sect. 4.2. Moreover, the code is able to decide wheather the user wants to build the model on GPU through *cuDNNLSTM* or on CPU through *LSTM*. *cuDNN-LSTM* is a Keras function which offers fast LSTM implementation with CuDNN [8]. It can only be run on Nvidia GPU, with the TensorFlow backend. cuDNN is a GPU-accelerated library of primitives for deep neural networks created by Nvidia. It provides GPU counterpart versions for standard routines such as forward and backward convolution, pooling, normalization and activation layers. *LSTM* function is the standard LSTM model defined by Hochreiter [18]. Finally, once the model has defined your model, it is compiled with *compile* function. This creates the structures previously defined by the underlying backend (in our case TensorFlow 10.11.1) in order to efficiently execute your model during training.

Table 1 LSTM parameters used for the different proposed experiments

Parameters	Values
Number of delay sequences	6
Batch size	32
Learning factor	0.001
Optimizer	Adam
Activation function	Hyperbolic tangent
Number of input neurons	50–200
Number of epochs	200–3000

4 Results and discussion

This section introduces the performance, energy and quality evaluation of our LSTM model as applied to predict the air temperature of a particular plot in an IoT infrastructure with edge computing capabilities. First of all, the experimental environment is described, providing the main metrics, hardware and software environment and datasets used in the experiments shown below. Then, the execution time and power consumption of the edge computing platform and the high-performance server are shown and discussed in detail before the quality evaluation of the LSTM to predict air temperature is provided.

4.1 Experimental setup

4.1.1 Metrics

The performance and energy evaluations are carried out by varying the most relevant parameters of the LSTM model, i.e., number of epochs and number of neurons. Moreover, this study analyzes the two main stages of this model, i.e., training and inference, which are carried out in both an edge computing platform (Nvidia Jetson Xavier) and a high-performance computing server. Energy of the system is measured by polling once every second the power supply of the Jetson architecture with the Watts Up Pro power meter, which provides individual energy measurements for the server connected to it, and therefore, energy measurements refer to the entire node executing the LSTM model. As for the quality of the results, the prediction of temperature is a regression task and the metric measures used to assess the quality of the results obtained with the LSTM model are the root mean quadratic error, the mean absolute error (MAE), the Pearson correlation coefficient (PCC) and determination coefficient (R^2). The results are positive when the RMSE and MAE measurements are less than one degree Celsius and PCC, and the closer to one R^2 is, the more acceptable and suitable the model is. For both experiments, 95% of the data are used to train the model and 5% of the data are used to perform the inference task.

Table 2 Each row's format collected every 10 min. 6 per hour. 144 per day

Date	Temperature	Wind speed	Humidity
12/12/2018 7:40	5.72	2.23	3.02

Table 3 Description of the datasets used in the experiments. It shows the number of instances, the start and end date period of the air temperature data

Datasets	No. of instances	Start date	End date
3-month	12,922	01/12/2018 0:09:00	28/02/2019 23:53:00
6-month	25,975	01/11/2018 0:00:00	30/04/2018 23:56:00
12-month	52,018	01/11/2017 0:05:00	31/10/2018 23:55:00
18-month	86,087	01/11/2017 0:05:00	30/06/2019 23:59:00

The optimum parameters used for the experiments are shown in Table 1. Parameters that do not influence the computational performance of the LSTM have been validated in previous experiments.

4.1.2 Datasets

The dataset to evaluate the LSTM model is obtained from the IoT system previously described in Sect. 3.1. Table 2 shows an example of the data collected by this architecture. The IoT infrastructure generates 144 rows per day (i.e., 1 row each 10 min), and the data layout is as follows:

– Date: in *dd/mm/aaaa* format.
– Hour: in *hh:mm* format.
– Air temperature: decimal number in Celsius degrees.
– Wind speed: decimal number in m/s.
– Relative air humidity: decimal number in %.

Our LSTM model only works with air temperature, which is collected every 10 min. For the experimental environment, four different datasets have been created to test scalability of edge and cloud solutions. They include a number of air temperature measurements from different periods, including 3,6,12 and 18 months, respectively (see Table 3).

4.1.3 Hardware and software environment

Experiments have been carried out in two different GPU-based platforms:

– The former is the edge computing like architecture Jetson AGX Xavier Developer Kit. It has four-core ARM v8.0 64-bit CPU, 8MB L2 + 4MB L3, 512-core

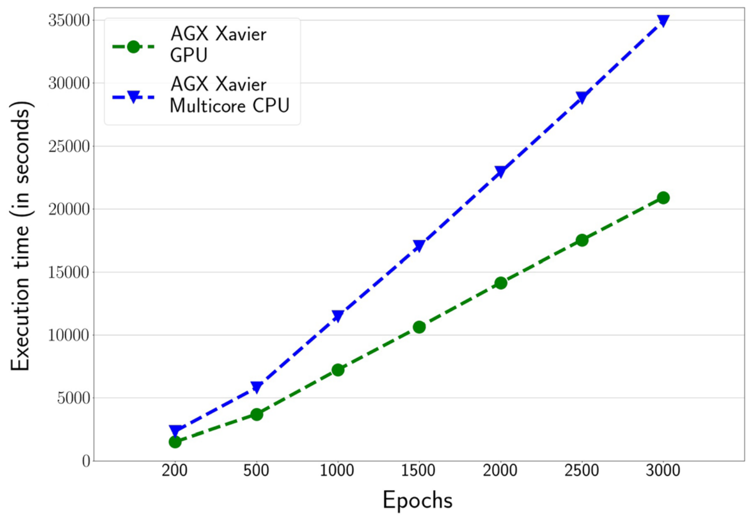


Fig. 5 Execution time (in seconds) of LSTM training stage with multicore CPU and GPU on AGX Xavier, varying the number of epochs for training. The training is carried out with information of 3 months with 150 neurons

Volta GPU with Tensor Cores and 16GB 256-Bit LPDDR4x running at 137GB/sec.

– The latter is called *HETEROLISTIC*, and it is composed of two hexa-core Intel Xeon E5-2650 at 2.20 GHz, 128 GB of RAM, private L1 and L2 caches of 32 KB and 256 KB per node, and a L3 cache of 32 MB shared by all the cores of a socket. It includes an Nvidia GTX 1080 Ti(Pascal), with 12 GB and 3584 cores (28 SM and 128 SP per SM).

The software environment is based on gcc 7.4.0 and cuda 10 and Python 3.6.5. The design of our LSTM model is based on Tensorflow 1.10.1 and Keras 2.2.4.

4.2 Evaluation

One of the main goals of this paper is to validate edge computing for complex tasks such as those under the umbrella of deep learning. Computational devices at the edge are traditionally low power and therefore have limited computational horsepower. Recently, more powerful edge computing devices have emerged, such as those from the Nvidia Jetson family, which include an accelerator to speed up parts of the code. Figure 5 shows the performance difference between a GPU-based code of our LSTM model compared to multithreaded CPU counterpart version, executed in the Nvidia Jetson AGX Xavier for the training stage and varying the number of epochs. Although the GPU architecture included in Xavier only includes a multiprocessor with 512 cores, more than 1.6x speedup factor is reported. Moreover, Fig. 6 shows the energy in KWh of the two implementations. Xavier's power consumption is higher when the GPU is enabled. Actually, the Xavier consumes 10W when the GPU code is running and only 8W when only CPU code does. However, the

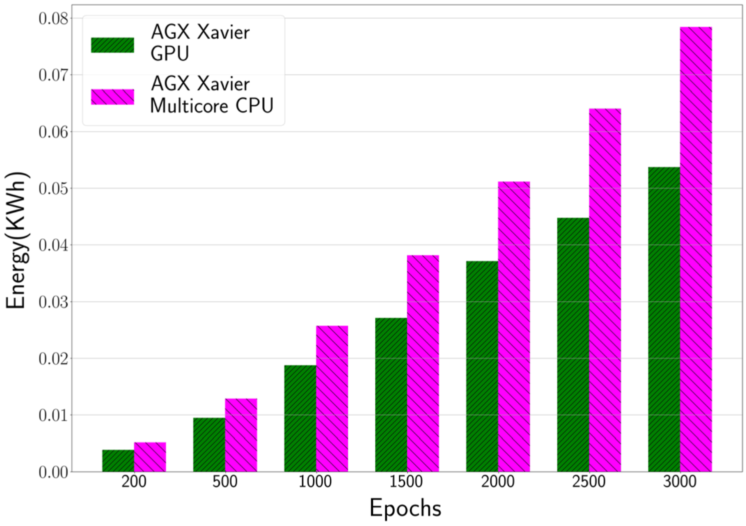


Fig. 6 Energy (in KWh) on AGX Xavier of LSTM training stage with multicore CPU and GPU on AGX Xavier, varying the number of epochs for training. The training is carried out with information of 3 months with 150 neurons

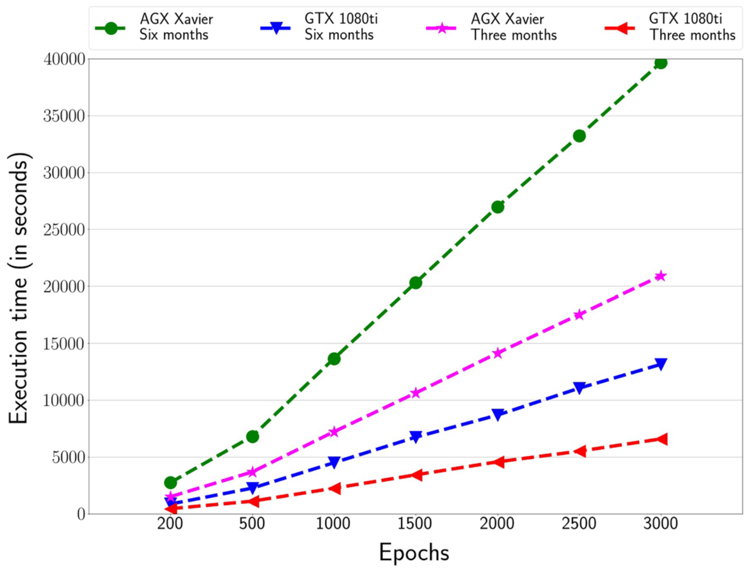


Fig. 7 Execution time (in seconds) of LSTM training stage, varying the number of epochs for training. The training is carried out with information of 3 and 6 months with 150 neurons

performance difference between these codes makes the GPU code more efficient compared to the CPU code in terms of energy consumption.

Figures 7 and 8 show the execution time of the proposed LSTM model to predict the temperature of a plot. Experiments are carried with the GPU version of the LSTM model on the edge computing platform Jetson AGX Xavier and the high-performance computing server *HETEROLISTIC*. Two performance figures are provided

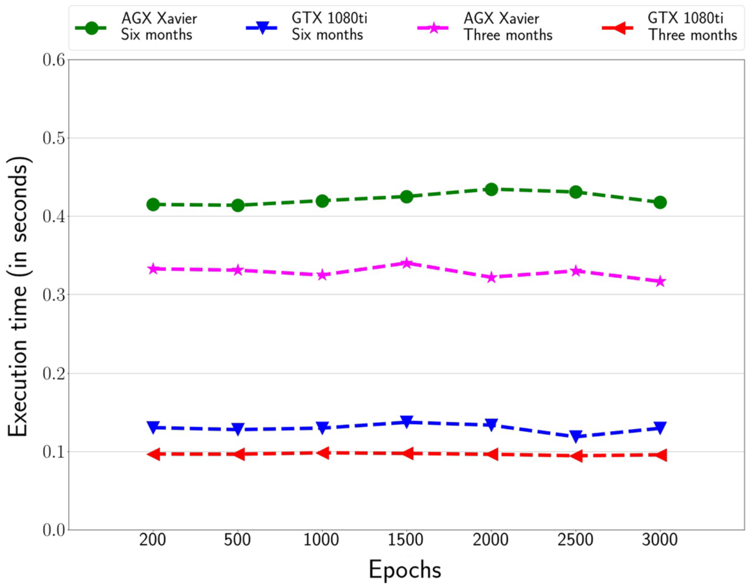


Fig. 8 Execution time (in seconds) of LSTM inference stage, varying the number of epochs for training. The inference is carried out with information of 3 and 6 months with 150 neurons

for main stages of the LSTM model, i.e., training and inference, and in both scenarios, *HETEROLISTIC* is defeated by a wide-margin Jetson Xavier (peak performance difference of 3.5x speed-up factor). However, the magnitude of the execution time for each of these stages is very different. Computationally speaking, the training process is more time-consuming than the inference process; a difference reaches up to five orders of magnitude, and thus different conclusions can be drawn. As previously explained, the LSTM is designed to be a part of intelligent component of an IoT infrastructure. The IoT infrastructure sends information periodically every 10 min, and therefore, this is time limit for making instantaneous predictions. Otherwise, the prediction will become obsolete, and the farmer would not be able to take actions before a temperature drops. Therefore, the execution time obtained for inference in Xavier is valid for performing instantaneous prediction on edge computing.

There are scenarios where the deadline to obtain the prediction is, at least, 12 h or even more. This is the case of frost prediction in places where there is a large diurnal temperature variation as it is the targeted region (South East, Spain). Frost prediction is a scenario where the farmer needs to know the prediction several hours in advance to activate the antifreeze mechanisms. In addition, climate change makes the weather even more unpredictable and changing. Therefore, the deep learning models need to be retrained periodically. Execution time for the training stage for 3000 epochs and 6-month dataset is less than 12 h (i.e., 11.01 h) in the case of Jetson Xavier, which is actually the worst of the cases presented in Figs. 7 and 8. The execution time is drastically reduced when the dataset size and the number of epochs are reduced. For instance, it takes less than 6 h to train with 3-month dataset, running 3000 epochs and less than 1 h if the training is performed with 3-month dataset and 200 epochs. Therefore, Jetson Xavier can offer great performance to enable an

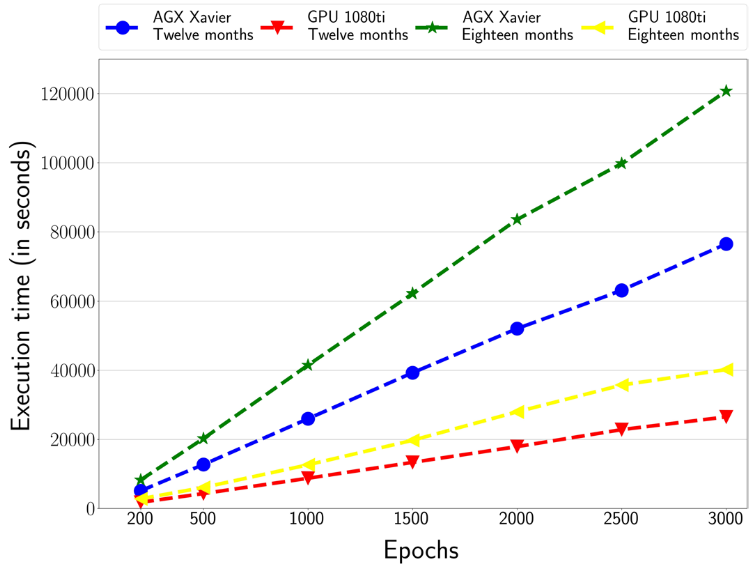


Fig. 9 Execution time (in seconds) of LSTM training stage, varying the number of epochs for training. The training is carried out with information of 12 and 18 months

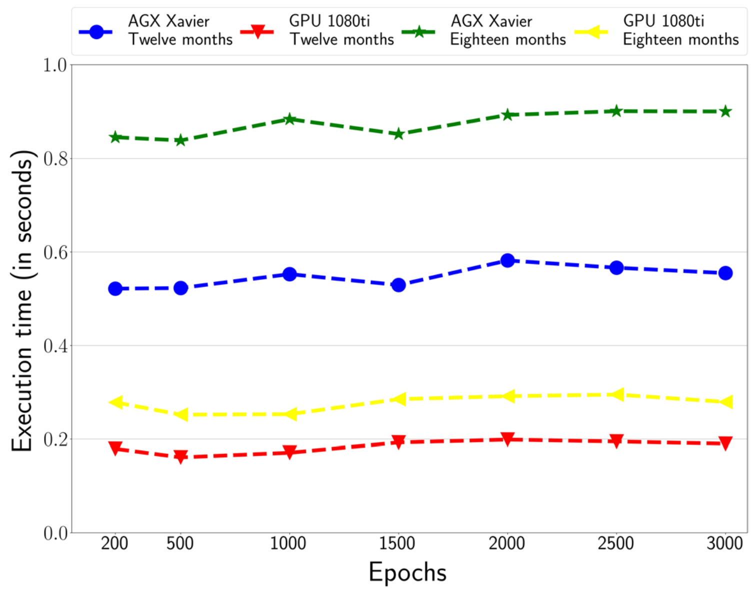


Fig. 10 Execution time (in seconds) of LSTM inference stage, varying the number of epochs for training. The inference is carried out with information of 12 and 18 months

autonomous decision support system for frost prediction, in which both training and inference are performed at the edge.

Figures 9 and 10 show large-scale (12- and 18-month datasets) training and inference on both Jetson Xavier and *HETEROLISTIC*. The training execution time grows exponentially along with the number of epochs, and inference stays flat.

Table 4 Execution time (in seconds) of LSTM training and inference stages, varying the number of neurons for training and setting the number of epochs to 2500

Neurons	Training process			Inference process		
	Jetson Xavier	GTX 1080ti	Speedup (col 2 vs. col 3)	Jetson Xavier	GTX 1080ti	Speedup (col 5 vs. col 6)
50	17660.26	5542.02	3.2	0.3249	0.0930	3.5
100	17523.19	5509.52	3.2	0.3301	0.0945	3.5
150	17693.14	5584.67	3.2	0.3193	0.0963	3.3
200	18502.43	5270.53	3.5	0.3337	0.0979	3.4

The training is carried out with information of 3 months

Finally, Table 4 shows the scalability of the LSTM when the number of neurons increases, setting the number of epochs to 2500. The execution time is not affected too much by increasing the number of neurons. The usage of GPUs allows high-performance data training of the LSTM on both architectures.

4.3 Quality evaluation

This section shows the results obtained from the LSTM model designed to predict the temperature from IoT information. First, we show and analyze the results of the three cross-validations and then we show the results of the test of 95% for training and 5% for testing (considering 24 consecutive hours).

After analyzing the results of the computational performance of the models in their training phase and given the high computational cost with the 18-month datasets, we have only analyzed the quality of the results for the 3-, 6- and 12-month datasets, since the time of 18 months was excessive, even the time for the 12-month datasets is considered excessive for an autonomous system, but we have considered it necessary to study the quality of the results, in case the prediction results were very remarkable.

Table 5 shows the error after model inference by setting the number of neurons to 150 and varying the number of epochs for 3-, 6- and 12-month datasets. The most suitable models are those obtained by the 3-month dataset, whose value of R^2 is 99%. For datasets of 6 and 12 months, the value is also acceptable but it is much better that of the 3-month dataset. Regarding the error, both RMSE and MAE, there is a remarkable fact and it is the increase above a Celsius degree of the 6-month dataset. This error increase is due to the fact that the datasets include summer months, which causes the model to have noise since the objective is to predict low temperatures. As can be seen, this error decreases for the 12-month dataset.

Analyzing in depth the results, we see how the smallest error and the most adjusted model are produced with 3000-epoch configuration. Nevertheless, looking at the difference in time in the experiment of computational performance between the configurations of 2500 and 3000 epochs, we have an hour of difference, being slower the training with 3000 epochs. Studying the difference of error

Table 5 Quality results for 3-month, 6-month and 12-month datasets varying the number of epochs. RMSE (root-mean-square error), MAE (mean absolute error), PCC (Pearson correlation coefficient) and R^2 (determination coefficient)

Datasets	Epochs	RMSE	MAE	PCC	R^2
3-month	200	0.7425	0.5377	0.9958	0.9909
	500	0.7372	0.5241	0.9954	0.9908
	1000	0.7117	0.4953	0.9957	0.9917
	1500	0.6969	0.4749	0.996	0.9919
	2000	0.7015	0.4831	0.996	0.9917
	2500	0.6731	0.451	0.9962	0.9924
	3000	0.6173	0.4136	0.9968	0.9936
6-month	200	1.6575	1.0028	0.9826	0.9654
	500	1.645	0.9851	0.983	0.9659
	1000	1.6136	0.9857	0.9836	0.9672
	1500	1.5631	0.9734	0.9846	0.9692
	2000	1.5418	0.9726	0.9849	0.9701
	2500	1.5332	0.9484	0.9851	0.9704
	3000	1.4435	0.8908	0.9869	0.9738
12-month	200	0.7627	0.5866	0.9598	0.9204
	500	0.7558	0.5803	0.9605	0.9218
	1000	0.7725	0.5946	0.9589	0.9183
	1500	0.7777	0.5890	0.9584	0.9171
	2000	0.8027	0.6133	0.9558	0.9118
	2500	0.8124	0.6228	0.9547	0.9097
	3000	0.8068	0.6152	0.9552	0.9109

Table 6 Quality results for 3-month, 6-month and 12-month datasets, varying the number of neurons. RMSE (root-mean-square error), MAE (mean absolute error), PCC (Pearson correlation coefficient) and R^2 (determination coefficient)

Datasets	Neurons	RMSE	MAE	PCC	R^2
3-month	50	0.7572	0.5106	0.9952	0.9904
	100	0.7425	0.5241	0.9954	0.9908
	150	0.6902	0.4771	0.996	0.992
	200	0.7158	0.4867	0.9958	0.9914
6-month	50	1.6379	0.9997	0.9831	0.9662
	100	1.6498	1.0121	0.9829	0.9657
	150	1.5467	0.9672	0.985	0.9699
	200	1.5772	0.9803	0.9843	0.9687
12-month	50	0.8466	0.6390	0.9515	0.9019
	100	0.8340	0.6335	0.9520	0.9048
	150	0.8124	0.6228	0.9547	0.9097
	200	0.7904	0.5995	0.9571	0.9145

between 2500 and 3000 epochs with the increase of 1 h, we consider more satisfactory considering performance vs. quality the result obtained with 2500 epochs. The two best results are highlighted in Table 5.

Once the quality of the results has been studied after varying the number of epochs, Table 6 analyzes the quality of the model by varying the number of input neurons to the model between 50 and 200. This is done for 3-, 6- and 12-month datasets. The behavior obtained is similar to the variation in the number of epochs. The 6-month dataset obtains a greater error than the 3-month and 12-month datasets. The most suitable model taking into account the determination coefficient R^2 is the 3-month dataset with 150 neurons. Already with 200 neurons, the model starts with overlearning and the error increases slightly.

Therefore, in view of the results, we can conclude that the data indicate that it is possible to run the prediction model of the LSTM at the edge, since with a training of 5 h, there is enough time between frost and frost, even if these occur on consecutive days. The most optimal configuration found is to train the data with the last 3 months collected by the node IoT taking as input 150 neurons and making a maximum of 2500 epochs. With this configuration, the mean quadratic error obtained is less than 0.8 Celsius degrees.

5 Conclusions and future work

Prior knowledge of low temperatures can help the farmer to anticipate resources and apply frost control techniques early enough to ensure maximum efficiency. The overall objective is to create an autonomous decision support system for precision agriculture, and in such hostile environments, connectivity is limited and there are transient clouds that limit the effectiveness of these systems. Edge computing provides a framework in which connectivity and security problems are addressed by computing at the edge of the network, but today's edge computing architectures are not able to handle heavy workloads.

This article evaluates edge computing for frost prediction in crops by estimating low temperatures through LSTM deep learning models. LSTM deep learning models are computationally heavy workloads, but provide very good results for predicting time series. Our results demonstrate that novel edge computing platforms including low-power GPUs such as Nvidia Jetson Xavier provide an excellent framework for driving edge computing as a real alternative to smart applications. Our best LSTM model obtains a deviation less than 1 degree centigrade, being trained with information of 3 months, 2500 epochs and 150 neurons, and its execution time is less than 5 h which allows training before the prediction is required.

As future work, new variables will be incorporated into the LSTM to create a multivariate LSTM and study the influence of other variables on temperature prediction, as well as the network adjustment creating a new architecture with more layers and different activation functions and different learning factors.

Acknowledgements This work was partially supported by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20813/PI/18, and by Spanish Ministry of Science, Innovation and Universities under grants RTI2018-096384-B-I00 (AEI/FEDER, UE) and RTC-2017-6389-5.

References

1. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, et al. (2016) Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), pp 265–283
2. Aliberti A, Bottaccioli L, Macii E, Di Cataldo S, Acquaviva A, Patti E (2019) A non-linear autoregressive model for indoor air-temperature predictions in smart buildings. Electronics 8(9):979
3. Bah MD, Dericquebourg E, Hafiane A, Canals R (2018) Deep learning based classification system for identifying weeds using high-resolution UAV imagery. In: Science and Information Conference. Springer, pp 176–187
4. Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing. ACM, pp 13–16
5. Boubin J, Chumley J, Stewart C, Khanal S (2019) Autonomic computing challenges in fully autonomous precision agriculture. In: 2019 IEEE International Conference on Autonomic Computing (ICAC), pp 11–17. IEEE
6. Cass S (2019) Taking AI to the edge: Google's TPU now comes in a maker-friendly package. IEEE Spectr 56(5):16–17
7. Chen X, Shi Q, Yang L, Xu J (2018) Thriftyedge: Resource-efficient edge computing for intelligent IoT applications. IEEE Netw 32(1):61–65
8. Chetlur S, Woolley C, Vandermersch P, Cohen J, Tran J, Catanzaro B, Shelhamer E (2014) cuDNN: efficient primitives for deep learning. arXiv preprint arXiv:1410.0759
9. Farooq MS, Riaz S, Abid A, Abid K, Naeem MA (2019) A survey on the role of IoT in agriculture for the implementation of smart farming. IEEE Access 7:156237–156271
10. Gondchawar N, Kawitkar R (2016) Iot based smart agriculture. Int J Adv Res Comput Commun Eng 5(6):838–842
11. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press, Cambridge
12. Graves A, Mohamed Ar, Hinton G (2013) Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp 6645–6649. IEEE
13. Grinblat GL, Uzal LC, Larese MG, Granitto PM (2016) Deep learning for plant identification using vein morphological patterns. Comput Electron Agric 127:418–424
14. Guillén-Navarro MA, Martínez-España R, Llanes A, Bueno-Crespo A, Cecilia JM (2020) A deep learning model to predict lower temperatures in agriculture. J Ambient Intell Smart Environ 12(1):21–34
15. Guillén-Navarro MA, Martínez-España R, López B, Cecilia JM (2019) A high-performance IoT solution to reduce frost damages in stone fruits. Concurr Comput Pract Exper. https://doi.org/10.1002/cpe.5299
16. Gulli A, Pal S (2017) Deep learning with Keras. Packt Publishing Ltd, Birmingham
17. Halawa H, Abdelhafez HA, Boktor A, Ripeanu M (2017) Nvidia Jetson platform characterization. In: EuropEan Conference on Parallel Processing. Springer, pp 92–105
18. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
19. Kamilaris A, Prenafeta-Boldu FX (2018) Deep learning in agriculture: a survey. Comput Electron Agric 147:70–90
20. Khosravi A, Koury R, Machado L, Pabon J (2018) Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. Sustain Energy Technol Assessm 25:146–160
21. Kratzert F, Klotz D, Brenner C, Schulz K, Herrnegger M (2018) Rainfall-Runoff modelling using Long-Short-Term-Memory (LSTM) networks. Hydrol. Earth Syst. Sci. Discuss., https://doi.org/10.5194/hess-2018-247
22. Kussul N, Lavreniuk M, Skakun S, Shelestov A (2017) Deep learning classification of land cover and crop types using remote sensing data. IEEE Geosci Remote Sens Lett 14(5):778–782
23. Kuwata K, Shibasaki R (2015) Estimating crop yields with deep learning and remotely sensed data. In: 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). IEEE, pp 858–861
24. Li Y, Zhao K, Chu X, Liu J (2013) Speeding up k-means algorithm by gpus. J Comput Syst Sci 79(2):216–229

25. Liakos K, Busato P, Moshou D, Pearson S, Bochtis D (2018) Machine learning in agriculture: a review. Sensors 18(8):2674
26. Mohammadi M, Al-Fuqaha A, Sorour S, Guizani M (2018) Deep learning for iot big data and streaming analytics: A survey. IEEE Commun Surv Tutor 20(4):2923–2960
27. Mohanty SP, Hughes DP, Salathé M (2016) Using deep learning for image-based plant disease detection. Front Plant Sci 7:1419
28. Pierpaoli E, Carli G, Pignatti E, Canavari M (2013) Drivers of precision agriculture technologies adoption: a literature review. Proc Technol 8:61–69
29. Qi S, Wan L, Fu B (2018) Multisource and multiuser water resources allocation based on genetic algorithm. J Supercomput, pp 1–9
30. Rodriguez SAB, Klein L, Schrott AG, Van Kessel TG (2019) Autonomous mobile platform and variable rate irrigation method for preventing frost damage . US Patent App. 10/219,448
31. Salman AG, Heryadi Y, Abdurahman E, Suparta W (2018) Single layer & multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting. Proc Comput Sci 135:89–98
32. Satyanarayanan M (2017) The emergence of edge computing. Computer 50(1):30–39
33. Satyanarayanan M, Bahl P, Caceres R, Davies N (2009) The case for VM-based cloudlets in mobile computing. IEEE Pervasive Comput 8(4):14–23
34. Seide F, Agarwal A (2016) CNTK: Microsoft's open-source deep-learning toolkit. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 2135–2135
35. Simoens P, Xiao Y, Pillai P, Chen Z, Ha K, Satyanarayanan M (2013) Scalable crowd-sourcing of video from mobile devices. In: Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services, pp 139–152. ACM
36. Zamora-Izquierdo MA, Santa J, Martínez JA, Martínez V, Skarmeta AF (2019) Smart farming iot platform based on edge and cloud computing. Biosyst Eng 177:4–17
37. Zaytar MA, El Amrani C (2016) Sequence to sequence weather forecasting with long short-term memory recurrent neural networks. Int Comput Appl 143(11):7–11
38. Zhang N, Wang M, Wang N (2002) Precision agriculture a worldwide overview. Comput Electron Agric 36(2–3):113–132
39. Zhou Z, Chen X, Li E, Zeng L, Luo K, Zhang J (2019) Edge intelligence: Paving the last mile of artificial intelligence with edge computing. Proc IEEE 107(8):1738–1762

Affiliations

Miguel A. Guillén[1] · Antonio Llanes[1] · Baldomero Imbernón[1] · Raquel Martínez-España[1] · Andrés Bueno-Crespo[1] · Juan-Carlos Cano[2] · José M. Cecilia[2]

Miguel A. Guillén
maguillen@ucam.edu

Antonio Llanes
allanes@ucam.edu

Baldomero Imbernón
bimbernon@ucam.edu

Raquel Martínez-España
rmartinez@ucam.edu

Andrés Bueno-Crespo
abueno@ucam.edu

Juan-Carlos Cano
jucano@disca.upv.es

[1] Universidad Católica San Antonio de Murcia (UCAM), Campus de los Jerónimos S/N, 30107 Murcia, Spain

[2] Universitat Politècnica de València, Camino de Vera S/N, 46022 Valencia, Spain

2.5 A decision support system for water optimization in anti-frost techniques by sprinklers

Título	*A decision support system for water optimization in anti-frost techniques by sprinklers*
Autores	Miguel Ángel Guillén-Navarro, Raquel Martínez-España, Andrés Bueno-Crespo, Juan Morales-García, Belén Ayuso y José María Cecilia
Revista	Sensors
Año	2020
Estado	Publicado

Contribución del Doctorando
Miguel Ángel Guillén Navarro, declara ser el principal autor y el principal contribuidor del artículo *A decision support system for water optimization in anti-frost techniques by sprinklers*.

sensors

MDPI

Article

A Decision Support System for Water Optimization in Anti-Frost Techniques by Sprinklers

Miguel A. Guillén-Navarro [1], Raquel Martínez-España [1,*], Andrés Bueno-Crespo [1] and Juan Morales-García [1], Belén Ayuso [1] and José M. Cecilia [2]

[1] Computer Science Department, Universidad Católica de Murcia (UCAM), 30107 Murcia, Spain; maguillen@ucam.edu (M.A.G.-N.); abueno@ucam.edu (A.B.-C.); jmorales8@alu.ucam.edu (J.M.-G.); bayuso@ucam.edu (B.A.)

[2] Computer Engineering Department (DISCA), Universitat Politècnica de València (UPV), 46022 Valencia, Spain; jmcecilia@disca.upv.es

* Correspondence: rmartinez@ucam.edu

Received: 16 November 2020; Accepted:10 December 2020; Published: 12 December 2020

Abstract: Precision agriculture is a growing sector that improves traditional agricultural processes through the use of new technologies. In southeast Spain, farmers are continuously fighting against harsh conditions caused by the effects of climate change. Among these problems, the great variability of temperatures (up to 20 °C in the same day) stands out. This causes the stone fruit trees to flower prematurely and the low winter temperatures freeze the flower causing the loss of the crop. Farmers use anti-freeze techniques to prevent crop loss and the most widely used techniques are those that use water irrigation as they are cheaper than other techniques. However, these techniques waste too much water and it is a scarce resource, especially in this area. In this article, we propose a novel intelligent Internet of Things (IoT) monitoring system to optimize the use of water in these anti-frost techniques while minimizing crop loss. The intelligent component of the IoT system is designed using an approach based on a multivariate Long Short-Term Memory (LSTM) model, designed to predict low temperatures. We compare the proposed approach of multivariate model with the univariate counterpart version to figure out which model obtains better accuracy to predict low temperatures. An accurate prediction of low temperatures would translate into significant water savings, as anti-frost techniques would not be activated without being necessary. Our experimental results show that the proposed multivariate LSTM approach improves the univariate counterpart version, obtaining an average quadratic error no greater than 0.65 °C and a coefficient of determination R^2 greater than 0.97. The proposed system has been deployed and is currently operating in a real environment obtained satisfactory performance.

Keywords: multivariate LSTM based approach; IoT system; intelligent systems; precision agriculture

1. Introduction

Agriculture is increasingly applying new technologies to increase its market opportunities and improve the use of available resources. Without a doubt, the most necessary natural resource in agriculture is water. However, water is an increasingly scarce resource and must be used rationally in this climate change era [1]. Precision agriculture (PA) is a discipline that offers a set of tools and techniques that enable farmers to increase their production and the quality of their production, while reducing costs and the resources used [2]. PA integrates a set of approaches to manage and help make decisions by reducing the uncertainty caused by the variability of the agricultural field. One of the uncertainties that create the most problems in agriculture is the sudden changes in the weather [3]. Among these abrupt changes are the variability of temperatures or torrential rainfall, causing great economic losses to both farmers and the sectors that depend on them [4]. In particular, this situation is particularly damaging in the southeast of the Mediterranean area, where crops can suffer temperature variations of up to 20 degrees centigrade (°C) within the same day. These ups and downs lead to spring temperatures in late autumn and early winter and low temperatures in early spring. These changes in the weather cause the fruit trees to bloom early and with the low temperatures in early spring the flowers freeze. To avoid crop loss, farmers apply different anti-frost techniques, depending on the type of crop [5]. Anti-frost techniques can be classified as active or passive. Active techniques are temporary and are energy or labor intensive, or both. Passive techniques are related to biological and ecological techniques, including practices carried out before a night of frost to reduce damage. Passive techniques are based on carrying out soil analysis, choosing suitable seeds in the plantations, genetic modifications in the plants, etc. The active techniques use micro-spraying, flood irrigation, stoves, covers, fans, and smoke and fog [6]. This study focuses on active anti-frost techniques, specifically those that use water as an anti-frost agent.

In addition to the meteorological problems, there are also water scarcity issues. Farmers in many areas are restricted in the amount of water available to irrigate their crops, so if they use disproportionate amounts of water for anti-frost techniques, they could save their crops but have no water available to harvest them. Although other anti-frost techniques could be used, most of them require a high installation cost and the amortization is a high burden for the farmers. The techniques that use water are the cheapest in installations, hence it is one of the most used. Therefore, the main objective of this study is to find a model that helps to predict when the temperature in the next hour with an error not exceeding one Celsius degree. This model will be implemented within the intelligent component of an IoT system that will have actuators to activate or deactivate the anti-frost techniques and thus save the maximum amount of water available. However, frost prediction is not an easy task, as it depends on several external environmental factors that are very difficult to control, such as temperature, humidity, wind speed, etc. In addition, there are also factors such as the location of the plot that are involved in this process. General weather forecasting provides information on a large scale which, on many occasions, is not sufficient to predict frost at plot level. At least, not with the precision to determine when to apply anti-frost techniques. Therefore, the particular conditions of each region of the plot must be taken into account in order to obtain more accurate predictive models.

Indeed, IoT systems play a fundamental role to monitor the climatic conditions of the crops automatically in a given area within a plot. IoT alone is not enough to provide an effective solution to this problem. The integration of both Artificial Intelligence (AI) and IoT is mandatory to enable such innovation. AI-enabled IoT (AIoT) brings sensors, machines, cloud-edge computing, analytics, and people together to improve productivity and efficiency, which implies revenue growth and operational efficiency. However, AI techniques, and, particularly, Machine Learning (ML) models involve computationally intensive tasks that also require high-quality data. Moreover, there are several challenges which limit the AIoT implementation in rural areas. Among them, we may highlight technical issues such as bandwidth, connectivity, power supply, and scaling sensors but also environmental issues that complicate the developments, such as extreme weather conditions.

This paper introduces an AIoT system to predict the air temperature based on several input variables. The aim of this AIoT is for farmers to activate their anti-frost techniques based on the use of water only when necessary, thus achieving significant savings of this scarce resource. Thus, we have designed and implemented a model based on a multivariate deep learning model based on Long Short-Term Memory (LSTM) to be integrated into an IoT system deployed in several agricultural plots. In the design of the model, the different alternatives have been evaluated, trying to implement the AIoT system in any plot without the need for significant historical data. In addition, the capacity of the proposed LSTM model to obtain a more precise result depending on the number of climate variables used is also analyzed. Thus, once the model has been assessed and validated, its aim is to activate the actuators and alert the farmer with sufficient precision and time so as not to lose the harvest, taking into account water savings. The LSTM model proposed in this article, deployed on the IoT infrastructure, creates a decision support system that activates and deactivates anti-freeze sprinkler techniques.

In what follows, the main research objectives of this manuscript are addressed:

1. Saving water in anti-frost techniques: One of the main objectives is to design and deploy a hardware and software AIoT infrastructure to monitor and predict temperatures in order to activate the anti-frost techniques based on sprinkler irrigation only when they are strictly necessary, taking care with the use of such a precious good as water.
2. Designing an accurate intelligent component through LSTM models: To achieve the previous goal, a multivariate LSTM model is proposed and evaluated to provide temperature predictions accurately.
3. Data characterization: Analyze the seasonality and amount of historical data needed to establish the system in other environments.
4. Integration and deployment: Integrate the data prediction model together with the IoT system to do the job of activating actuators and saving water.

Each of these objectives is described, detailed and addressed in Section 3 of this manuscript. Thus, this study is organized as follows. First, a brief background is given on aspects related to deep learning and precision agriculture in Section 2. In Section 3, the description and architecture of the AIoT infrastructure is provided, and the prediction model used (LSTM) is explained in depth. It also describes the datasets and variables used to analyze the historical data and the configuration of the experiments carried out to create the prediction model. Then, Section 4 provides the results of an analysis of them and a discussion of the objectives achieved are presented. Finally, some conclusions and orientations for future work are explained.

2. Related Work

This section briefly summarizes the state of the art of precision agriculture techniques in the field of water management. It is well known that the development of intelligent environments for application to precision agriculture is widespread, and this is an appropriate field of action as described in [7]. A first interesting work is [8], where a model composed of an LSTM layer with another fully connected layer on top of it is developed to predict water table depth. They use a dropout method applied in the first LSTM layer. This model is evaluated in China with monthly data during 14 years of water diversion, evaporation, precipitation, and temperature. Moreover, it could help engineers and decision-makers to plan and manage groundwater resources in agricultural areas. The use of such a model in the field of hydrology is very novel and shows that there are areas where different models can be applied to a traditional neural network. The results obtained from the proposed model indicate a value of R^2 of 0.86. Sahoo et al. [9] proposed a model which predicts the level of groundwater in an area of the United States. Since the demand for irrigation water influences the wells in the area from which the water is drawn, farmers can know the level of the groundwater before using this water. The results show a correlation value in the model greater than 0.8.

Another work that demonstrates to what extent Artificial Intelligence can help solve everyday situations but with an important economic impact on small farms is described in [10]. Coopersmith et al. model the process by which the soil becomes wet and dry. This can help farmers decide when is a suitable time to bring in heavy machinery. They have used only publicly-accessible information and classification trees, k-nearest-neighbors and boosted perceptron deliver statistical soil dryness estimates. The authors achieve an accuracy value of 94% with the proposed model. In terms of water management, there are several articles which propose evapotranspiration and evaporation methodologies. For instance, Patil and Deka show an accurate estimation of weekly evapotranspiration in arid regions of India using a method developed by them and based on the ELM model fed with temperature data for the weekly estimation of evapotranspiration for two weather stations [11]; Mohammadi et al. show that the dew point is a very significant element for the identification of meteorological phenomena (evapotranspiration, evaporation, and frost) [12]. They introduce a model for the prediction of the daily dew point temperature based on the LMA. Feng et al. show two scenarios where the estimation of daily evapotranspiration from temperature data collected at six weather stations was carried out [13]. Finally, in [14], a recurrent LSTM neural network for the forecast of the Cimandiri River level in Indonesia is presented, and this model achieves a relative error of less than 10%.

The authors of [15] present an approach to predict temperature and humidity using a two-level sequential decomposition structure. First, the meteorological data were decomposed into four components in series. Then, each of these series is introduced to a gated recurrent unit, and the output information from each of these units is combined to obtain the prediction results. The data used to test the proposed approach are collected using a IoT system. The results indicate a root mean square error (RMSE) greater than 2.4 degrees Celsius for temperature and greater than 13% for humidity prediction. In [16], an intelligent frost management system is created. The connection in real time is guaranteed by the use of a web that allows the interaction of the environmental system with the weather station and the ecological anti-freeze irrigation. The system uses a neural network and a Fuzzy Expert System—the former to optimally predict the temperature inside the greenhouses and the latter to control the activation of a water pump. Obtaining R^2 values for the temperature in the summer of 0.91 and in the winter of 0.95. The authors of [17] deploy the LSTM model for temperature prediction and propose the transductive LSTM (T-LSTM) by altering the cost function in the regression problem. With the T-LSTM model, they obtain a root mean square error greater than 1.5 degrees centigrade in the prediction of temperature.

To the best of our knowledge, although there are some articles that use LSTM to deal with agriculture issues, there is no study that addresses frost prediction through LSTM techniques. Specifically, the LSTM model proposed in this article does not require a great amount of historical data to obtain robust and reliable temperature predictions. This guarantees that it can be applied in different plots for short-term predictions.

3. AIoT System Proposed

As in other economic sectors, IoT technology has revolutionized the agricultural sector [18,19]. This is mainly due to the reduction in size, energy consumption, and cost of hardware components, which has democratized the use of these infrastructures in this sector. The IoT system proposed in this article monitors the climatic data of an agricultural plot and, based on the data and the intelligent system developed, the system acts by activating the anti-frost system and alerting the farmer to make the right decisions if necessary. The main objective of this system is to optimize the use of water in the anti-frost system, effectively predicting the air temperature by means of deep learning techniques included in the intelligent module.

Figure 1 shows the AIoT infrastructure proposed in this article. The system consists of three different components: an infrastructure of agricultural sensors and actuators, an intelligent data processing system, and a monitoring component. The IoT infrastructure has three main sub-modules with three sensors connected to each of them (i.e., humidity, temperature, and wind speed). Each submodule communicates with the central module (actuator) using LoRa technology. LoRa enables long-distance communications, which allows the sensors to be dispersed in the agricultural field, without having to leave the local network. The actuator sends data to the cloud, where the proposed LSTM model will indicate the expected temperature in order to decide whether the actuator activates or deactivates the anti-frost technique, i.e., the sprinklers. In what follows, we introduce each of these components in detail.

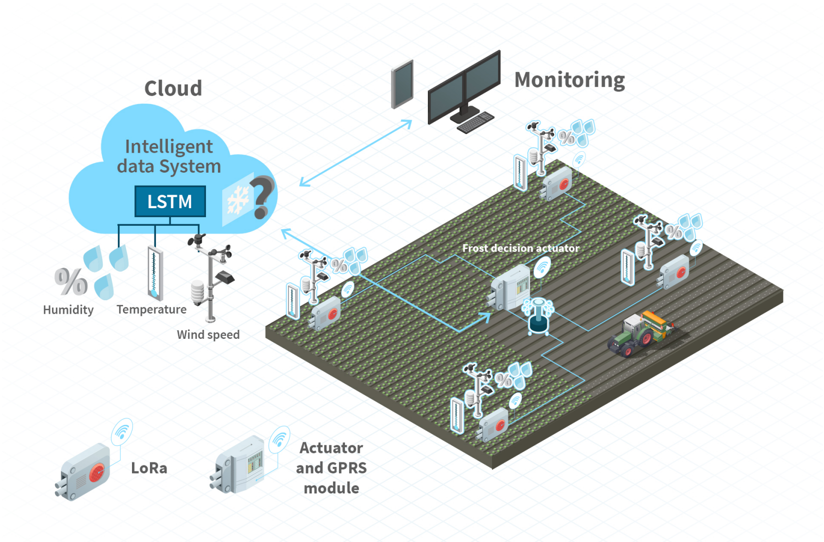


Figure 1. AIoT system architecture.

3.1. Agricultural Sensors and Actuators Infrastructure

Firstly, an IoT infrastructure has been deployed in a real agricultural field, located in Cieza (Region of Murcia, Spain). This infrastructure is based on a wireless sensor network that collects (WSN) data on temperature, humidity, and wind speed. These data are sent via GPRS technology to the intelligent component of the system, to decide if the actuators should start remotely activating the anti-frost technique. One of the most relevant aspects and the one that has given more analysis to design the proposed IoT solution has been the choice of the communication technology between the different nodes.

3.1.1. Analyzing the Wireless Communication Technologies

The wireless communication technology is one of the most important decisions in these scenarios. There are many IoT communication protocols which offer different characteristics for the efficient interconnection of IoT nodes [20]. However, the agriculture scenario requires long transmission distance, has freedom of use, low power consumption, minimal data transfer and is feasible to implement in sensors, actuators, and nodes outdoors. There are several data transmission standards which meet these characteristics such as SigFox (https://www.sigfox.com/), LoRa (https://lora-alliance.org/), or ZigBee (http://www.zigbee.org/). In our previous work, we tested ZigBee for connecting our IoT infrastructure since it is one of the most popular technologies in the field of agriculture [21,22]. Although its technical specifications indicate a communication range of 100 m, our initial tests did not show a communication range of more than 60 m [23].

Thus, a second scenario using the LoRa communication protocol was developed as a means to communicate with the IoT nodes. Among the advantages offered by this standard is the low power consumption and the great distance it reaches, in addition to being a free use protocol. The choice of using LoRa instead of Sigfox is based on the possibility of deploying our own network, since Sigfox imposes certain restrictions on its use. Therefore, the technology selected and implemented in the IoT solution for minimizing frost damage has been the LoRa communication protocol.

3.1.2. IoT Sensor Hardware Description

The hardware architecture of the sensor network was previously explained in [22,24]. It is based on the 4H remote control system of the company Hidroconta (http://www.hidroconta.com/).

The IoT node includes air temperature, humidity and wind speed sensors. These sensors have been calibrated by the company, so we have not had to perform any calibration phase. We briefly described them below:

- IoT nodes: The IoT architecture is composed of three nodes (a master and two slaves), where the sensors are connected to (see LoRa node in Figure 1). The master node is the gateway to send the data to the cloud through the GPRS connection. The connection of the slave nodes with the master node and the actuator is also done by LoRa technology. The other two nodes are connected to the master through LoRa. These nodes are autonomous since they have a 6-volt (V), 12-amp (A) per hour battery and a 12 V, 5-watt (W) solar panel. In addition, each IoT node has 96 KB of volatile data memory and a microcontroller with 256 KB of firmware storage. The former can be extended with an external non-volatile memory of 244 KB, which can store more than 20,000 registers. Therefore, each node stores information in its own memory besides sending the information to the gateway. Thus, if there is a communication failure, the information can be resent. Finally, the sensors and actuators are connected to the analog inputs available at the IoT node.
- Sensors and Actuators: The sensors and actuator components used in the proposed IoT solution are the following. The humidity sensor is COMET P3110E (https://www.cometsystem.com/) which also provides air temperature measurements. The temperature lies between −30 °C to 80 °C with an error of 0.6 °C and the humidity lies between 0 to 100% with an error of 3%. The temperature sensor is COMET 60P8610. The air temperature range lies between −20 °C to 60 °C with a resolution of 0.1 °C. The wind speed sensor is the PCE Instruments (https://www.pce-instruments.com/), model PCE-WS, whose measurement range is between 3 and 180 km/h with an error of km/h.

In this article, more variables than temperature have been included in order to carry out an in-depth analysis of the data and thus analyze the impact of having a multivariate model on the prediction of frost several hours in advance. Having two air temperature sensors (i.e., humidity sensor and air temperature) also provides the detection of outliers and/or temperature errors. Moreover, it also includes the possibility to detect the lowest temperature to activate the anti-frost technique. Actuator: The actuator is a valve that activates and deactivates the anti-freeze spray technique. It is a lach type solenoid which has three different modes of operation, resistance up to 80 °C, and 12 bar pressure.

3.2. IoT Intelligent Component

An IoT system needs an intelligent component to provide support on the farmer's decision. The IoT intelligent component presented in this article includes a deep learning model based on Long Short-Term Memory neural networks (LSTM), specifically designed to work with the input provided by the IoT system developed. These input data are nonlinear times series where deep learning models have shown very good results [25]. The use of recurrent neural networks (RNN) with time series is a very powerful tool because it allows information to persist through a loop in the network diagram, allowing for remembering previous states and using this information to decide which will be the next stage to carry out. However, conventional RNNs usually have problems in their training due to the accumulation of errors that are generated, since it depends not only on the current error but also on past errors, and this makes them not very efficient for the memorization of dependencies in the long term sequences.

To address this problem, the network can remove some of the information that introduces the accumulated error and give more importance to the most recent data such as autoregressive models [26]. Inside an LSTM unit, there is what is known as an internal gate, which is formed by a sigmoid layer and a multiplication operation. Each LSTM memory unit contains three gates that control how the information flows into or out of the unit. The first gate is the input gate which controls when new information can get into the memory. The second gate is known as the forgotten gate which controls when a piece of information is forgotten. This allows less important data to be forgotten, allowing new data to be added. This second gate needs to combine the sigmoid layer with an additional layer formed by a hyperbolic tangent. The last module is the output door which controls that the generated output is saved in the cell state. Each LSTM unit transmits to the next its prediction, which, together with the current input unit, generates the output that is sent as input to the next LSTM unit (see Figure 2).

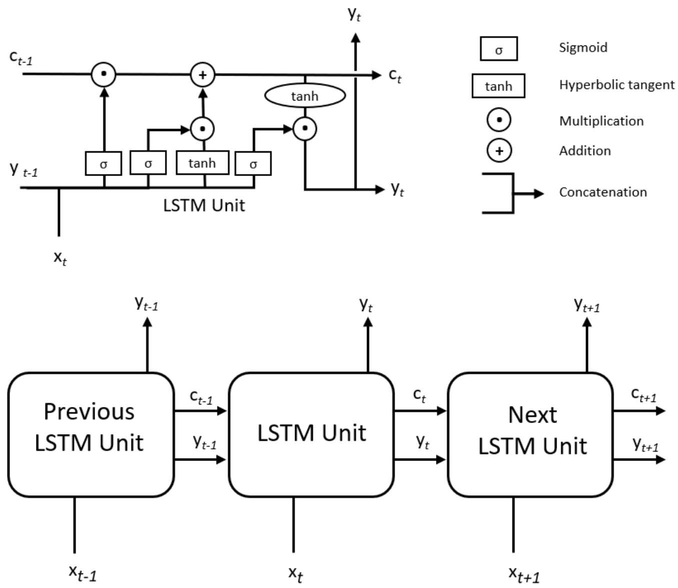


Figure 2. LSTM Univariate. The single input is the value to predict. The output obtained (y_t) and the cell state (c_t) for each LSTM unit is the input (x_t) of the next unit.

In addition to the univariate LSTM scheme, a multivariate LSTM scheme with multiple inputs and a single output is presented (Figure 3). The output used and the input x_1 is the same as the univariate LSTM system presented in the previous section. Therefore, the x_1 input is the target. This model has the advantage of the additional information provided by the other inputs which implies an increase in the dimensionality of the input layer. This increase in dimensionality makes sense due to the relationship between the variable to be predicted and the rest of the input features added to the multivariate model, which makes the model more robust in its forecast than the univariate model previously presented.

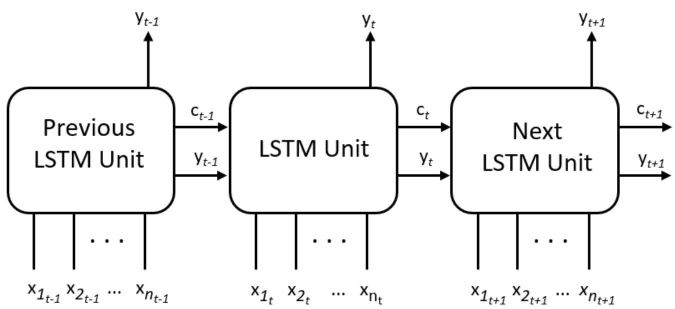


Figure 3. LSTM Multivariate. It is composed of several inputs (n represents the number of inputs) and a single output. The first input is the one predicted in the output, as it happens in the univariate model.

For the multivariate model, several experiments have been performed with different combinations of data, which is explained in detail in Section 3.4.

3.3. Description of the Datasets

The validation and training of the proposed LSTM prediction model have been done using the historical data provided by the Sistema de Información Agrario de Murcia (SIAM) (http://siam.imida.es/). This institution has a set of public data that it collects through a set of meteorological stations with different climatic sensors. Since the deployed AIoT system is in Cieza, we have selected the closest weather station to the place of deployment of the weather station to validate and train the proposed model. The objective of validating the proposed model using the public data of this institute is because this institute has meteorological data with a history of years and one of the objectives of this study is to analyze if it is necessary to have a history of data for the proposed LSTM model to obtain more accurate results. In addition, we also want to show that a multivariate model can produce better results than a univariate model. Therefore, the historical dataset requires including, at least, the variables air temperature, humidity and wind speed that the deployed IoT system will have and with which the inference in the intelligent module will be made. The selected SIAM meteorological station is located in the coordinates 38.2839 latitude, −1.49634 longitude, and 244 altitude in the "La Carrichosa" area in Cieza, Región of Murcia, Spain. The measures available in this station are a weather vane, a pluviometer, a radiometer, a thermometer, and a datalogger, the latter to collect all data and avoid losses in case of connection failures. With these meters, the climatic variables (in brackets are the abbreviations used to refer to them) offered by this station are the following:

- Maximum, average, and minimum temperature (TMAX, TMED, TMIN).
- Maximum, average, and minimum relative humidity (HRMAX, HRMED, HRMIN).
- Maximum and average radiation (RADMAX, RADMED).
- Average and maximum wind speed (VVMED, VVMAX).
- Medium wind direction (DVMED).
- Vapor pressure deficit (DPV).

The data provided at that station are collected every hour and this station has had data since 1996; however, for this study, we have started in the year 2012, as in previous years there was a lot of data missing and the model could be affected. The frosts in the area where the present study is focused occur during the month of January and February since this is the period where the temperature changes and the trees are in bloom. Thus, the months selected to assess and validate the model are December, January, and February. Table 1 shows an overview of the datasets, considering the number of instances of each dataset and the input variables to build the LSTM model. It is important to note that the output variable in our LSTM model is always TMIN.

The information contained in each dataset is detailed below:

- DSU-3-month: This dataset contains instances from 1 December 2017 to 28 February 2018, where the input and output variable is the minimum temperature.
- DSM-3-month: This dataset contains instances from 1 December 2017 to 28 February 2018, being composed of the 18 input variables provided by the weather station and the output variable being the minimum temperature.
- DSU-1-month: This dataset contains only January 2018 instances, where the input and output variable is the minimum temperature.
- DSM-1-month: This dataset contains only January 2018 instances, where there are 18 input variables, which come from the weather station and the output variable is the minimum temperature.
- DSUH-1-month: This data set contains only instances from the months of January 2012 to 2018 that have as input and output the variable minimum temperature.

- DSMH-1-month: This data set contains only the instances for the months of January 2012 to 2018. The input variables are the 18 climate variables provided by the weather station for each of the months of January. The output variable is the minimum temperature.
- DSMS-1-month: This last data set contains only the data of January 2018, considering a selection of input variables that influences the minimum temperature, which is actually the output variable.

Table 1. Description of the datasets used for model validation, indicating number of instances of each dataset (N.Instances) and input variables (Input variables). The acronyms of these variables mean: TMED- mean temperature, TMAX- maximum temperature, TMIN-minimum temperature, HRMED-mean relative humidity, HRMAX-maximum relative humidity, HRMIN- minimum relative humidity, RADMED-mean radiation, RADMAX- maximum radiation, RADACU- accumulated radiation, VVMED-mean wind speed, DVMED- mean wind direction, VVMAX-maximum wind speed, PREC-rainfall, DEWPT- dew point, and DPV-pressure vapor deficit.

Dataset	N.Instances	Input Variables
DSU-3-month	2160	TMIN
DSM-3-month	2160	TMED, TMAX, TMIN, HRMED, HRMAX, RMIN, RADMED, RADMAX, RADACU, VVMED, DVMED, VVMAX, PREC, DEWPT, DPV
DSU-1-month	744	TMIN
DSM-1-month	744	TMED, TMAX, TMIN, HRMED, HRMAX, RMIN, RADMED, RADMAX, RADACU, VVMED, DVMED, VVMAX, PREC, DEWPT, DPV
DSUH-1-month	4464	TMIN
DSMH-1-month	4464	TMED, TMAX, TMIN, HRMED, HRMAX, HRMIN, RADMED, RADMAX, RADACU, VVMED, DVMED, VVMAX, PREC, DEWPT, DPV
DSMS-1-month	744	TMIN, HRMAX, RADMAX, RADACU, VVMED, DVMED,PREC, DEWPT, DPV

3.4. Experiment Configuration for IoT intelligent Components

This section describes the configuration used for a set of experiments, which aim to assess the quality of the univariate and multivariate LSTM models developed in this work. It is worth highlighting that LSTM models are designed to work with time series data. Thus, in the datasets described in Section 3.3, the data are considered in a temporal way taking samples every hour. Different types of data are considered in each dataset in order to find the best combination of input variables to be considered in the model, as well as time history in data needed that will ensure a minimum error when predicting the minimum temperature. Thus, the anti-frost actuators are only activated when it is convenient, saving water resources. The actuators are configured so that, when the temperature for the next hour is 1.5 Celsius degrees (°C) or lower, these actuators will be activated so that the entire surface can be wetted sufficiently in advance before the temperature falls below zero. A series of experiments will be carried out to evaluate the effectiveness of the method in predicting frost. First, 90% of the data in each set will be used to train the LSTM models and 10% to test and evaluate the model, so that the subset selected for testing has not been used for training and corresponds to the last 10% of the dataset. In other words, the data for the evaluation corresponds to approximately three days for the data sets that only use one month of data. However, when several years of data are considered, it would be evaluating with data corresponding to approximately 10 and 15 days, these days being the last ones of the last month because data considered as time series are being evaluated. Each LSTM model used, both univariate and multivariate, has been optimized by performing a sweep for each of the different parameters indicated below, always considering as the best parameters those whose RMSE is lower, and for the same RMSE value, the value of the parameter that obtains the best R2 has been considered. Table 2 shows the optimized parameters for each LSTM model. Some of these parameters are set in an interval pattern since they depend on the type of experiment. In the evaluation of the quality of the models described, we use the metrics of Root Mean Square Error (RMSE), Mean Absolute Error (MAE), the Pearson Correlation Coefficient (PCC) and Determination coefficient (R2). It is necessary to remark that for all the models we made the task of regression trying to predict the minimum temperature in the following hour to the considered one.

Table 2. Optimal parameter for LSTM execution experiments for the univariate and multivariate models.

Parameter	Univariate Model Values	Multivariate Model Values
Number of input neurons	70	60:90
Batch size	32	32
Number of epochs	230	340:750
Learning factor	0.001	0.001
Optimizer	Adam	Adam
Activation function	Hyperbolic Tangent	Hyperbolic Tangent
Loss Function	Quadratic Mean Error	Quadratic Mean Error
Delay Sequence	6	6

Finally, simulations have been carried out in a GPU-based platform. This platform is composed of Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40 GHz, 128 GB of RAM, 1 TB SSD Hard Disk, and a NVIDIA GeForce GTX 780 GPU (Kepler). The software environment where the LSTM models has been designed and implemented is based on Tensorflow 1.7.0, Keras 2.1.5, and Python 3.5.2.

4. Results and Discussion

This section shows the results obtained for the proposed multivariate LSTM model, comparing its results with the univariate counterpart version. The evaluation data set is described in Table 1. The objective is to find the best set of data and input variables that will provide a model that will be the basis of a decision support system for farmers to reduce their water expenditure, since water is a scarce resource worldwide and even more so in the area where this study is focused.

Table 3 shows the results of the different datasets considering the RMSE, MAE, PCC, and R2 metrics. It is worth highlighting that the main objective is to obtain a model with the least possible error, but, if errors exist, it would be desirable to have them for high temperatures, i.e., above 7 °C. In addition, the output for all datasets is the minimum temperature at the following hour.

Table 3. Results obtained applying univariate and multivariate LSTM models to the datasets according to the different datasets and configurations of the input variables. For each dataset, the Root Mean Square Error (RMSE), the Mean Absolute Error (MAE), the Pearson Correlation Coefficient (PCC), and the determination coefficient (R^2) are provided. Results ordered from highest to lowest RMSE.

Dataset	RMSE (°C)	MAE (°C)	PCC	R^2	LSTM Model
DSUH-1-month	1.0641	0.7636	0.9717	0.9442	Univariate
DSU-3-month	0.8860	0.6172	0.9844	0.9691	Univariate
DSM-3-month	0.8637	0.6842	0.9854	0.9706	Multivariate
DSMH-1-month	0.8075	0.6223	0.9761	0.9517	Multivariate
DSU-1-month	0.7227	0.5451	0.9814	0.9613	Univariate
DSM-1-month	0.6495	0.5309	0.9853	0.9687	Multivariate
DSMS-1-month	0.6353	0.5194	0.9856	0.9701	Multivariate

Analyzing the results in a general way, the errors of the minimum temperature prediction obtained are not higher than 1.1 °C and the models created have an adjustment higher than 90%. In addition, the difference between the error values of the RMSE and MAE are not high, which indicates that there are no temperature peaks that can be considered outlier and that can be included negatively in the models. However, if the results are deeply analyzed, the DSUH-1-month dataset, obtains the worst error, with an RMSE above 1 °C and a MAE above 0.75 °C. This result is obtained using the univariate LSTM model, where there is only an input variable, i.e., the minimum temperature. The data considered in this input are the minimum hourly temperatures for the months of January 2012 to 2018. This data set assesses whether there is an egalitarian behavior of the data over the years and, using the trend of the previous years, it is possible to predict the values of the following years. As can be seen in the result, although the result is not conceptually bad, it is not as precise as is necessary for the problem posed.

The DSU-3-month dataset is the following configuration according to the results. In this case, the error, although it is less than 1 °C according to RMSE, is still higher than in other configurations. However, the R2 shows that the model is satisfactory, but, from the point of view of precision, it is not as satisfactory as we would like. This dataset uses the same idea as the DSUH-1-month dataset. The main difference in this case is that three months of data history have been considered, taking as input only the minimum temperature and using the univariate LSTM model. The DSM-3-month dataset is the one that continues obtaining a worse result than the previous ones. In this case, the result is very similar to the one obtained by the DSU-3-month dataset that also uses a three-month history. In this case, the DSM-3-month dataset uses the multivariate LSTM model because it takes as input variables all the climatic variables offered by the weather station studied. In this case, the aim is to analyze the place of the temperature trend for each January, if for each hour, using more climatic variables, these influence the prediction of the minimum temperature for the following hour. The RMSE indicates a value of 0.8637 °C and a model fit of 0.9706, but this is still a high error even though the model fit is the best of all settings. The DSMH-1-month dataset has as input variables the 18 climate variables provided by the studied weather station for each hour of January of the years 2012 and 2018. This dataset has an RMSE of 0.8075 °C. Despite being better than the previous ones, it is still a considerable error. The DSU-1-month dataset is the third best model obtained. In this case, the univariate LSTM model is used and as input variable the minimum temperature is taken only from January 2018. The result is very optimistic because it falls below 0.8 °C and the model adjustment achieved is higher than 96%. However, analyzing the results of the datasets mentioned above, there are indications that the multivariate LSTM model obtains a better performance, hence the experiment with the DSM-1-month dataset, where each hour of the month of January is evaluated taking as inputs the 18 climatic variables obtained from the meteorological station studied. With this last multivariate model, an RMSE of almost 0.65 °C is obtained, which improves the previous result. This result has very little room for improvement since it is just over 0.5 °C. However, by analyzing the 18 input variables, a variable selection is made, eliminating those that, according to the literature, may not directly influence the minimum temperature, such as the maximum temperature of the day, the average relative humidity, and the average wind speed. The selection of variables gives rise to the DSMS-1-month dataset, which obtains a lower RMSE and lower MAE, being 0.6353 and 0.5194, respectively, and a satisfactory model fit. This model created using the multivariate LSTM model is the best and therefore is the candidate to build the basis of the decision support system. In addition, this study has given us the classes and needs to extend the proposed AIoT system with more sensors, in particular, with sensors that measure radiation, wind direction, and precipitation.

Although the differences in the RMSE values are not very large, Figures 4 and 5 depict graphically the trend of the two best models obtained, one for univariate and the other for multivariate, corresponding to the DSU-1-month and DSMS-1-month datasets, respectively.

Figure 4 shows how the model is able to follow temperature trends but has some differences in temperature changes. On the other hand, Figure 5 depicts a greater monitoring of the trend of minimum temperature obtaining a close to perfect prediction in some temperatures. Both models, univariate and multivariate, obtain a greater error in high temperatures, but this does not influence our model, since we seek to be precise in low temperatures to save water and activate anti-frost techniques only when necessary.

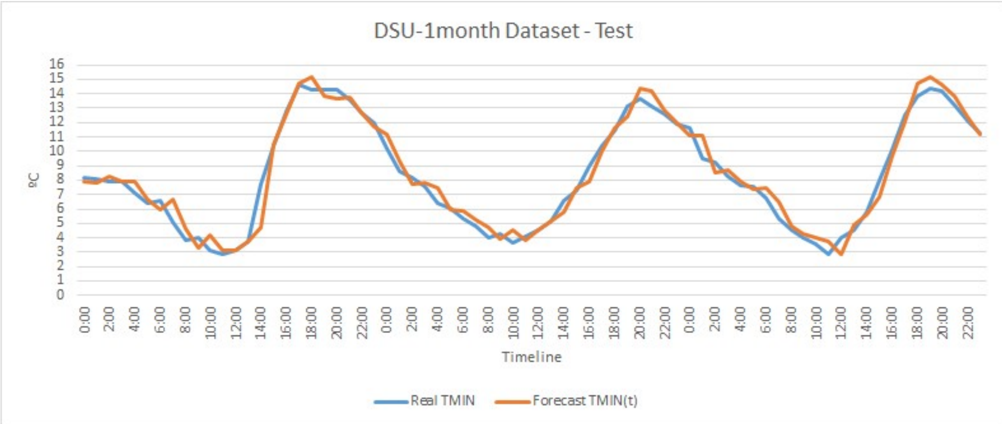


Figure 4. Graphical representation of the actual and predicted minimum temperature values in the test by the univariate LSTM model for the dataset DSU-1-month.

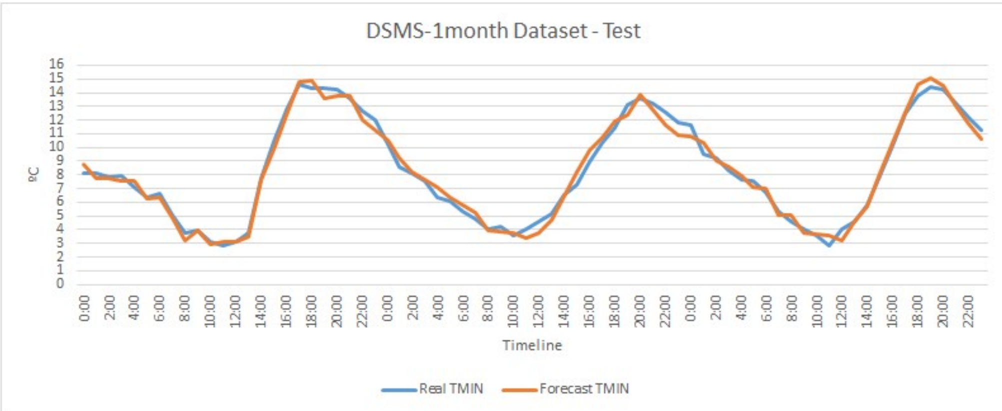


Figure 5. Graphical representation of the actual and predicted minimum temperature values under testing by the multivariate LSTM model for the dataset DSMS-1-month.

Figure 6 shows a comparison between the errors, calculated in absolute value, of the actual temperature and the predicted temperature for the datasets DSMS-1month and DSU-1month. This figure allows us to better appreciate the magnitude of errors obtained by univariate and multivariate models. As can be seen, the multivariate model is much more stable in terms of errors and most of them are below 0.7 °C. However, the univariate model is less stable, since although initially it seems to work the same as the multivariate model, it makes errors with a greater variation in temperature, which makes it less robust and stable than the multivariate model.

The results obtained demonstrate the achievement of the objectives initially proposed. These results are satisfactory because they demonstrate a small error and a good adjustment of the models, in comparison with the results obtained in other works presented in Section 2, and our models show lower RMSE and higher R^2 ratios.

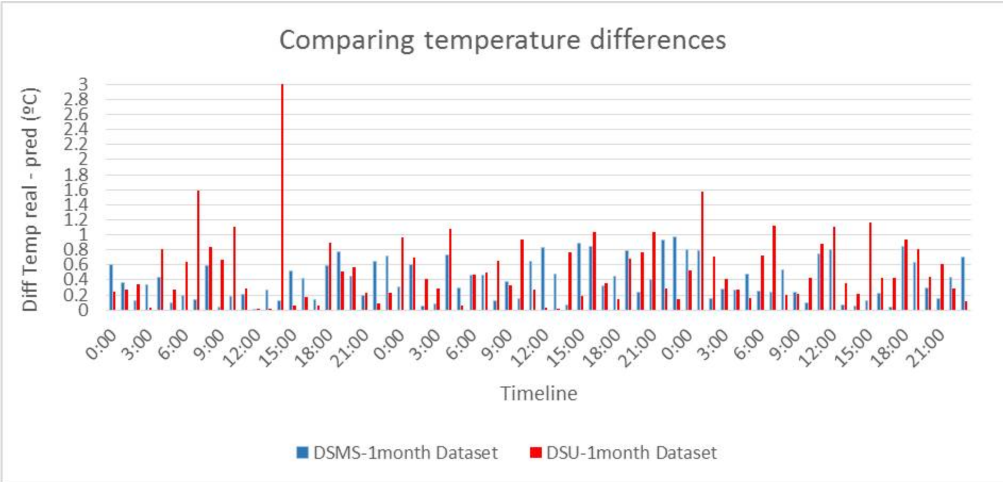


Figure 6. Comparison of the error between the real and the predicted temperature values by the models (under testing) generated with the DSMS-1-month and DSU-1-month datasets.

5. Conclusions and Future Work

Climate change is causing trees to flower earlier and sudden changes in temperature can cause serious economic damage to farmers by freezing flowers and damaging crops. In addition, there is often unnecessary water wastage in areas particularly vulnerable to drought. This article proposes an intelligent IoT system to optimize water use in crops by accurately predicting minimum air temperature at plot level and thus limiting the use of the anti-frost sprinkler irrigation techniques. The intelligent component in the IoT infrastructure proposed is the key point for decision-making. It is designed using an approach based on a multivariate LSTM model which accurately provides air temperature predictions with enough time in advance to take actions to avoid losing the crop. The best configuration of the proposed LSTM multivariate model will be integrated into the IoT system as a decision support component. The objective of the proposed model is to achieve the least possible error so that the farmer can trust the system and the system only activates the irrigation sprinklers to avoid frost when the temperature will be less than or equal to 1.5 °C, this activation being reliable based on the prediction of the temperature in the following hour. In the result analysis of the proposed multivariate LSTM model, we have done comparisons with the univariate LSTM model using different combinations of input variables that can influence the temperature prediction. The best results obtained are satisfactory as they indicate that the model that obtains a lower RMSE is the LSTM multivariate model that uses only data from the same month to train and nine climatic input variables to predict the minimum air temperature. Thus, the results of the proposed multivariate LSTM model obtain an RMSE of 0.64° and a fit value of the R2 model of 0.97. In the future, we will work on analyzing the possibility of implementing the multivariate model at the edge of the IoT infrastructure to prevent transient cloud outages and improve performance. Other intelligent services will be provided such as the prediction of evaporation in rafts to optimize agriculture procedures and other different applications will be deployed such as the climate control in greenhouses.

Author Contributions: Conceptualization, J.M.C. and R.M.-E.; methodology, J.M.C., R.M.-E., and A.B.-C.; software, M.A.G.-N and J.M.-G.; validation, J.M.C., R.M.-E., and A.B.-C.; formal analysis, A.B.-C., J.M.C., and R.M.-E; investigation, J.M.C.; data curation, A.B.-C., M.A.G.-N. and J.M.-G.; writing—original draft preparation, M.A.G.-N., R.M.-E., J.M.C., and A.B.-C.; writing—review and editing, J.M.C.; visualization, B.A. and M.A.G.-N.; supervision, R.M.-E. and B.A.; project administration, J.M.C.; funding acquisition, J.M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially supported by the Spanish Ministry of Science and Innovation, under the Ramon y Cajal Program (Grant No. RYC2018-025580-I) and under grants RTI2018-096384-B-I00, RTC-2017-6389-5 and RTC2019-007159-5, by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20813/PI/18, and by the "Conselleria de Educación, Investigación, Cultura y Deporte, Direcció General de Ciéncia i Investigació, Proyectos AICO/2020", Spain, under Grant AICO/2020/302.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Melgarejo-Moreno, J.; López-Ortiz, M.I.; Fernández-Aracil, P. Water distribution management in South-East Spain: A guaranteed system in a context of scarce resources. *Sci. Total Environ.* **2019**, *648*, 1384–1393. [CrossRef] [PubMed]
2. Ferrández-Pastor, F.J.; García-Chamizo, J.M.; Nieto-Hidalgo, M.; Mora-Martínez, J. Precision agriculture design method using a distributed computing architecture on internet of things context. *Sensors* **2018**, *18*, 1731. [CrossRef] [PubMed]
3. Liaghat, S.; Balasundram, S.K. A review: The role of remote sensing in precision agriculture. *Am. J. Agric. Biol. Sci.* **2010**, *5*, 50–55. [CrossRef]
4. Nelson, G.C.; Van Der Mensbrugghe, D.; Ahammad, H.; Blanc, E.; Calvin, K.; Hasegawa, T.; Havlik, P.; Heyhoe, E.; Kyle, P.; Lotze-Campen, H.; et al. Agriculture and climate change in global scenarios: Why don't the models agree. *Agric. Econ.* **2014**, *45*, 85–101. [CrossRef]
5. Crookston, R.K. A top 10 list of developments and issues impacting crop management and ecology during the past 50 years. *Crop Sci.* **2006**, *46*, 2253–2262. [CrossRef]
6. Snyder, R.L.; Melo-Abreu, J.D. *Frost Protection: Fundamentals, Practice and Economics*; FAO: Roma, Italy, 2005; Volume 1.
7. Dutta, R.; Morshed, A.; Aryal, J.; D'este, C.; Das, A. Development of an intelligent environmental knowledge system for sustainable agricultural decision support. *Environ. Model. Softw.* **2014**, *52*, 264–272. [CrossRef]
8. Zhang, J.; Zhu, Y.; Zhang, X.; Ye, M.; Yang, J. Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas. *J. Hydrol.* **2018**, *561*, 918–929. [CrossRef]
9. Sahoo, S.; Russo, T.A.; Elliott, J.; Foster, I. Machine learning algorithms for modeling groundwater level changes in agricultural regions of the US. *Water Resour. Res.* **2017**, *53*, 3878–3895. [CrossRef]
10. Coopersmith, E.J.; Minsker, B.S.; Wenzel, C.E.; Gilmore, B.J. Machine learning assessments of soil drying for agricultural planning. *Comput. Electron. Agric.* **2014**, *104*, 93–104. [CrossRef]
11. Patil, A.P.; Deka, P.C. An extreme learning machine approach for modeling evapotranspiration using extrinsic inputs. *Comput. Electron. Agric.* **2016**, *121*, 385–392. [CrossRef]
12. Mohammadi, K.; Shamshirband, S.; Motamedi, S.; Petković, D.; Hashim, R.; Gocic, M. Extreme learning machine based prediction of daily dew point temperature. *Comput. Electron. Agric.* **2015**, *117*, 214–225. [CrossRef]
13. Feng, Y.; Peng, Y.; Cui, N.; Gong, D.; Zhang, K. Modeling reference evapotranspiration using extreme learning machine and generalized regression neural network only with temperature data. *Comput. Electron. Agric.* **2017**, *136*, 71–78. [CrossRef]
14. Sudriani, Y.; Ridwansyah, I.; Rustini, H.A. Long short term memory (LSTM) recurrent neural network (RNN) for discharge level prediction and forecast in Cimandiri river, Indonesia. In *IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2019; Volume 299, p. 012037.
15. Jin, X.B.; Yu, X.H.; Wang, X.Y.; Bai, Y.T.; Su, T.L.; Kong, J.L. Deep learning predictor for sustainable precision agriculture based on internet of things system. *Sustainability* **2020**, *12*, 1433. [CrossRef]
16. Castañeda-Miranda, A.; Castaño-Meneses, V.M. Internet of things for smart farming and frost intelligent control in greenhouses. *Comput. Electron. Agric.* **2020**, *176*, 105614. [CrossRef]
17. Karevan, Z.; Suykens, J.A. Transductive LSTM for time-series prediction: An application to weather forecasting. *Neural Netw.* **2020**, *125*, 1–9. [CrossRef]
18. Tzounis, A.; Katsoulas, N.; Bartzanas, T.; Kittas, C. Internet of Things in agriculture, recent advances and future challenges. *Biosyst. Eng.* **2017**, *164*, 31–48. [CrossRef]
19. Shi, X.; An, X.; Zhao, Q.; Liu, H.; Xia, L.; Sun, X.; Guo, Y. State-of-the-art internet of things in protected agriculture. *Sensors* **2019**, *19*, 1833. [CrossRef]

20. Jawad, H.M.; Nordin, R.; Gharghan, S.K.; Jawad, A.M.; Ismail, M. Energy-efficient wireless sensor networks for precision agriculture: A review. *Sensors* **2017**, *17*, 1781. [CrossRef]
21. Cambra, C.; Sendra, S.; Lloret, J.; Garcia, L. An IoT service-oriented system for agriculture monitoring. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6.
22. Guillén-Navarro, M.A.; Martínez-España, R.; López, B.; Cecilia, J.M. A high-performance IoT solution to reduce frost damages in stone fruits. *Concurr. Comput. Pract. Exp.* **2019**, e5299. [CrossRef]
23. Kuzminykh, I.; Snihurov, A.; Carlsson, A. Testing of communication range in ZigBee technology. In Proceedings of the 2017 14th International Conference the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), Lviv, Ukraine, 21–25 February 2017; pp. 133–136.
24. Guillén, M.A.; Llanes, A.; Imbernón, B.; Martínez-España, R.; Bueno-Crespo, A.; Cano, J.C.; Cecilia, J.M. Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning. *J. Supercomput.* **2020**. [CrossRef]
25. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; Volume 1.
26. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long time lag problems. *Adv. Neural Inf. Process. Syst.* **1996**, *9*, 473–479.

Capítulo 3

Conclusiones y vías futuras

La tesis doctoral que se presenta en este compendio de publicaciones tiene como objetivo fundamental la predicción de heladas en áreas específicas y su comparación con las predicciones globales.

En este capítulo se hace un resumen de las conclusiones que se pueden extraer de todo el desarrollo de la tesis, así como las futuras líneas de trabajo.

3.1 Conclusiones

Durante la realización de esta tesis doctoral se han ido obteniendo una serie de resultados de forma gradual que han ido cumpliendo los objetivos propuestos, y que se comentan a continuación.

En el primer artículo publicado se diseñó e implementó una infraestructura para la captura de variables atmosféricas, en diferentes puntos de una parcela situada en Cieza (Murcia). Al ponerla en funcionamiento se detectaron algunos errores en la obtención de datos. Por ello, para localizar los valores que quedaban fuera (outliers) se evaluaron dos alternativas, KNN y K-means. Se obtuvieron mejores resultados con el segundo.

En el caso de k-means para mejorar el rendimiento, y debido a su adecuación para ser ejecutado en paralelo, se ha hecho una comparativa entre entornos Cloud y Edge. El resultado en el entorno Edge no es tan lejano en rendimiento al Cloud, sobretodo si se tiene en cuenta el bajo consumo del primero. Esta fue una conclusión fundamental que permitió, tras una serie de estudios, la publicación del tercer artículo de la tesis.

En el segundo artículo del compendio, tomando como referencia los estudios realizados para el anterior, se propone una red neuronal LSTM. Por la problemática a solucionar este tipo de redes, optimizadas para series temporales, se adaptan mucho mejor para obtener un buen resultado. Así se certifica una vez analizados los datos obtenidos.

Se construyeron dos modelos, uno teniendo en cuenta los datos de una única parcela, Cieza, y otro el conjunto de datos de dos ubicaciones, Cieza y Moratalla. Se verificó que un modelo local se ajusta más a la realidad que uno construido sobre el total de los datos. En este sentido cabe resaltar que la desviación de la predicción respecto de la real fue menor a un grado celsius. En esta línea también es importante indicar que el porcentaje de falsos negativos -se predice que no hiela pero al final hiela- es mucho más bajo con el modelo local que con el global.

Este segundo artículo concluye con una comparativa entre los resultados obtenidos con el modelo construido mediante LSTM y otras técnicas de series temporales, confirmándose la idoneidad de LSTM respecto a las otras.

En el tercer artículo de este compendio, como continuación de una de las conclusiones del primero, se aborda la posibilidad de desplegar un sistema aislado, sin conexión al servidor. El objetivo principal era evaluar la idoneidad de usar Edge computing como alternativa a la ejecución en la nube, ya que muchas de las parcelas se encuentran en entornos rurales aislados, sin conexión a la red o con una conectividad muy reducida.

Aunque en la actualidad la potencia de cálculo de esta tecnología es mucho menor con respecto a la ejecución del mismo modelo en la nube, en este artículo se comprobó que con una GPU se pueden obtener buenos resultados. En concreto, se ha ejecutado el modelo LSTM sobre una plataforma Edge computing de GPUs de baja potencia, Nvidia jetson Xavier.

Los resultados obtenidos se han comparado con los que se obtienen en un entorno cloud. Aunque a priori se observa que en la GPU se necesita de un mayor tiempo de ejecución, dada la forma de trabajar de la arquitectura propuesta, con frecuencia de lectura del sensor establecida en diez minutos, se concluye que es tiempo más que suficiente para inferir si helará o no. Es decir, aunque el servidor tarda mucho menos tiempo en ejecutar y dar una respuesta, el tiempo de lectura de datos es más que suficiente para que la GPU devuelva un resultado. Por tanto, la conclusión final de este artículo es que los entornos de ejecución Edge computing son válidos para el tipo de problema que se aborda ya que permiten, dentro de un entorno aislado y de bajo consumo, ejecutar el modelo y predecir la helada.

En el último artículo de este compendio se pretende mejorar la predicción que realiza la LSTM incorporando más variables al modelo y, de esta forma, activar los sistemas antiheladas con mayor precisión. Se estudiaron diferentes conjuntos de datos y combinaciones de variables hasta conseguir una configuración adecuada. En el mejor de ellos se obtuvo solamente un RMSE de 0,64ºC y un coeficiente de determinación R^2 de 0,97. Datos sin duda muy satisfactorios y que permiten concluir que el objetivo principal de esta tesis ha sido alcanzado.

3.2 Vías futuras

En esta tesis doctoral se ha desarrollado un modelo de predicción de heladas y se ha probado sobre una arquitectura Edge de cara a validar su uso dentro de un entorno aislado. Aunque estas dos líneas han sido trabajadas a lo largo de toda la investigación, suponen en sí un punto de partida para futuros estudios.

Aun cuando en el último artículo ya se describió como se ha implementado un modelo multivariable, una línea de trabajo muy interesante es incorporar nuevas variables a la LSTM, así como estudiar la influencia de otras variables sobre la predicción de temperatura. En este sentido, reajustar la red creando una nueva arquitectura con más capas y diferentes funciones de activación y factores de aprendizaje es una vía futura que aportará buenos resultados a la hora de predecir heladas.

Otro aspecto a mejorar es el diseño y mejora de técnicas de preprocesamiento de datos para la eliminación y corrección de valores outliers provocados por la recogida de datos a través de sensores. En la presente tesis doctoral se ha trabajado sobre la detección de outliers sobre valores de temperatura, pero también es interesante ampliar la detección a otras variables como humedad o velocidad del viento. Además se deben analizar otros algoritmos de detección y corrección de outliers.

Dentro del mismo espacio de trabajo, para la mejora del modelo de predicción, una aproximación puede ser la implementación de un comité de redes LSTM cada una configurada y entrenada de forma diferente, de tal manera que el nuevo sistema tendrá más resultados sobre los que apoyar la decisión de activar el sistema antihelada.

Destacar que el uso de Edge Computing ayudará a la hora de crear un sistema autónomo eficiente y preciso. En este sentido se debe trabajar en analizar la posibilidad de implementar dentro de la infraestructura IoT el modelo multivariable. Otros servicios inteligentes, tales como predicción de evaporación en balsas o pantanos, podrán ser provistos para optimizar los procedimientos agrícolas y otras aplicaciones que permitan el control de clima en invernaderos.

Una línea de trabajo futuro es la integración de la arquitectura propuesta con diferentes sistemas antiheladas, optimizando su uso en base a la predicción y a la técnica de prevención a utilizar. En este ámbito, estudiar el tipo de helada y los mecanismos de detección existentes ayudará a la hora de decidir si activar el sistema antihelada.

Por último, indicar que este modelo de predicción puede ser llevado a otros entornos como turismo inteligente, deporte de precisión o mantenimiento de instalaciones inteligente.

Capítulo 4

Publicaciones, calidad de las revistas y otras publicaciones

La tesis doctoral que se presenta, está formada por un compendio de publicaciones científicas publicadas en revistas del alto impacto e indexadas por JCR. A continuación se muestran sus referencias bibliográficas completas.

Publicación	
Título	*A high-performance IoT solution to reduce frost damages in stone fruits*
Autores	Miguel Ángel Guillén-Navarro, Raquel Martínez-España, Belén López, y José María Cecilia
Revista	Concurrency and Computation: Practice & Experience
Año	2021
DOI	10.1002/cpe.5299
Estado	Publicado

Autores – Afiliación	
Nombre	**D. Miguel Ángel Guillén Navarro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dra. Raquel Martínez España**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dra. Belén López Ayuso**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. José M. Cecilia Canales**
Universidad	Universidad Católica de Murcia (UCAM)

Detalles de la revista Concurrency and Computation: Practice and Experience
Redactor jefe: David W. Walker, Jinjun Chen, Nitin Auluck y Martin Berzins eISSN: 1532-0634 Editorial: John Wiley Factor de impacto (2019): 1.447 (Journal Citation Reports) Categoría: Computer Science, Theory & Methods Ranking: Q3, posición: 55 de 108 Página Web: `https://onlinelibrary.wiley.com/toc/15320634/2021/33/2`

Publicación	
Título	*A deep learning model to predict lower temperatures in agriculture*
Revista	Journal of Ambient Intelligence and Smart Environments
Año	2020
DOI	10.3233/AIS-200546
Estado	Publicado

Autores – Afiliación (Orden Alfabético)	
Nombre	**D. Miguel Ángel Guillén Navarro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dra. Raquel Martínez España**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Antonio Llanes Castro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Andrés Bueno Crespo**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. José M. Cecilia**
Universidad	Universidad Católica de Murcia (UCAM)

Detalles de la revista Journal of Ambient Intelligence and Smart Environments
Redactor jefe: Hamid Aghajan ISSN: 1876-1364 eISSN: 1876-1372 Editorial: IOS Press Factor de impacto (2019): 1.595 (Journal Citation Reports) Categoría: Computer Science, Hardware & Architecture Ranking: Q3, posición: xx de xx Página Web: `https://content.iospress.com/journals/journal-of-ambient-intelligence-and-smart-environments/13/2`

Publicación	
Título	*Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning*
Revista	The Journal of Supercomputing
Año	2021
DOI	10.1007/s11227-020-03288-w
Estado	Publicado

Autores – Afiliación	
Nombre	**D. Miguel Ángel Guillén Navarro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Antonio Llanes Castro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Baldomero Imbernón Tudela**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dra. Raquel Martínez España**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Andrés Bueno Crespo**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Juan Carlos Cano**
Universidad	Universidad Politécnica de Valencia (UPV)
Nombre	**Dr. José M. Cecilia**
Universidad	Universidad Católica de Murcia (UCAM)

Detalles de la revista *The Journal of Supercomputing*
Editor-in-Chief: Hamid Arabnia
ISSN: 0920-8542
eISSN: 1573-0484
Editorial: Elsevier
Factor de impacto (2016): 2.469 (Journal Citation Reports)
Categoría: Computer Science, Theory & Methods
Ranking: Q2, posición: 31 de 108
Página Web: `https://www.springer.com/journal/11227`

Publicación	
Título	*A decision support system for water optimizationin anti-frost techniques by sprinklers*
Revista	Sensors
Año	2020
DOI	10.3390/s20247129
Estado	Publicado

Autores – Afiliación	
Nombre	**D. Miguel Ángel Guillén Navarro**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dña. Raquel Martínez España**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. Andrés Bueno Crespo**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**D. Juan Morales García**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dra. Belén López Ayuso**
Universidad	Universidad Católica de Murcia (UCAM)
Nombre	**Dr. José M. Cecilia**
Universidad	Universidad Politécnica de Valencia (UPV)

Detalles de la revista *Sensors*
Editor-in-Chief: Dr. Raffaele Bruno
eISSN: 1424-8220
Editorial: MDPI
Factor de impacto (2019): 3.275 (Journal Citation Reports)
Categoría: Instruments & Instrumentation
Ranking: Q1, posición: 5 de 64
Página Web: `https://www.mdpi.com/journal/sensors`

4.1 Datos relativos a la calidad de las publicaciones

Las publicaciones que forman parte de la tesis doctoral están todas publicadas en revistas de alto impacto según el índice JCR. Para cada una de las publicaciones, se van a detallar los indicares de calidad según índice JCR, teniendo en cuenta la categoría de la revista, su índice de impacto y el cuartil en el cuál se encuentra la revista.

4.1.1 A high-performance IoT solution to reduce frost damages in stone fruits - Concurrency and Computation: Practice & Experience

En esta sección se indican los datos e indicadores de calidad de la revista *Concurrency and Computation: Practice & Experience* donde se ha publicado el primer artículo que compone el compendio de esta tesis doctoral titulado *A high-performance IoT solution to reduce frost damages in stone fruits*. En las figuras 4.1, 4.2 y 4.3 se muestran los datos de la revista, los indicadores de calidad y el factor de impacto.

CONCURRENCY AND COMPUTATION-PRACTICE & EXPERIENCE

ISSN: 1532-0626
eISSN: 1532-0634
WILEY
111 RIVER ST, HOBOKEN 07030-5774, NJ
ENGLAND

Go to Journal Table of Contents Go to Ulrich's Printable Version

TITLES
ISO: Concurr. Comput.-Pract. Exp.
JCR Abbrev: CONCURR COMP-PRACT E

View Title Changes

CATEGORIES
COMPUTER SCIENCE, SOFTWARE ENGINEERING -- SCIE
COMPUTER SCIENCE, THEORY & METHODS -- SCIE

LANGUAGES
English

PUBLICATION FREQUENCY
18 issues/year

Figura 4.1: Información de la revista donde se ha publicado el artículo *A high-performance IoT solution to reduce frost damages in stone fruits*.

Current Year 2018 2017 All Years

Key Indicators - All Years Export

Customize columns

Year	Total Cites	Journal Impact Factor	Impact Factor without Journal Self Cites	5 Year Impact Factor	Immediacy Index	Citable Items	% Articles in Citable Items	Average JIF Percentile
	✓Trend	Trend	Trend	Trend	Trend	Trend	Trend	Trend
✓2019	2,908	1.447	1.290	1.268	0.769	433	99.77	46.759
2018	2,576	1.167	1.093	1.148	1.102	244	99.59	39.666
2017	2,157	1.114	0.984	1.114	0.905	367	100.00	44.947
2016	2,057	1.133	1.002	1.219	1.065	248	99.60	39.536
2015	1,384	0.942	0.741	0.914	0.709	309	100.00	44.549
2014	1,184	0.997	0.670	0.958	0.461	165	100.00	55.340
2013	1,064	0.784	0.671	0.854	0.865	148	98.65	45.021
2012	904	0.845	0.738	0.790	0.276	152	100.00	49.488
2011	800	0.636	0.476	0.779	0.236	140	100.00	35.477

Figura 4.2: Indicadores clave de los últimos años de la revista *Concurrency and Computation: Practice & Experience*.

Rank

JCR Impact Factor

JCR Year	COMPUTER SCIENCE, SOFTWARE ENGINEERING			COMPUTER SCIENCE, THEORY & METHODS		
	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile
2019	61/108	Q3	43.981	55/108	Q3	49.537
2018	70/107	Q3	35.047	59/105	Q3	44.286
2017	62/104	Q3	40.865	53/103	Q3	49.029
2016	66/106	Q3	38.208	62/104	Q3	40.865
2015	59/106	Q3	44.811	59/105	Q3	44.286

Figura 4.3: Ranking, Cuartil y Factor de Impacto de la revista *Concurrency and Computation: Practice & Experience* en los últimos años según categorías.

4.1.2 A deep learning model to predict lower temperatures in agriculture - Journal of Ambient Intelligence and Smart Environments

En esta sección se indican los datos e indicadores de calidad de la revista *Journal of Ambient Intelligence and Smart Environments* donde se ha publicado el segundo artículo que compone el compendio de esta tesis doctoral titulado *A deep learning model to predict lower temperatures in agriculture*. En las figuras 4.4, 4.5 y 4.6 se muestran los datos de la revista, los indicadores de calidad y el factor de impacto.

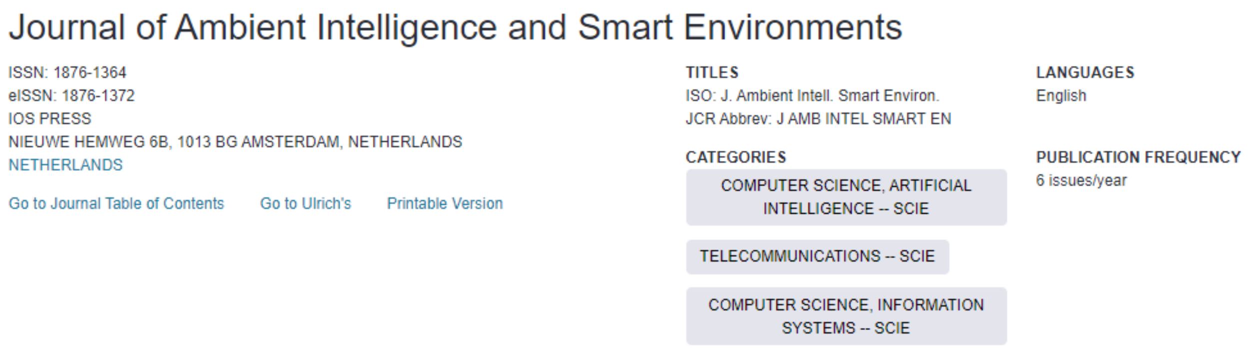


Figura 4.4: Información de la revista donde se ha publicado el artículo *A deep learning model to predict lower temperatures in agriculture*

Current Year 2018 2017 All Years

Key Indicators - All Years

Export

Customize columns

Year	Total Cites	Journal Impact Factor	Impact Factor without Journal Self Cites	5 Year Impact Factor	Immediacy Index	Citable Items	% Articles in Citable Items	Average JIF Percentile
	✓Trend	Trend	Trend	Trend	Trend	Trend	Trend	Trend
✓2019	424	1.595	1.392	1.275	0.750	28	96.43	30.053
2018	306	1.186	1.116	0.900	0.103	29	96.55	20.488
2017	362	0.878	0.744	1.005	0.533	45	100.00	17.027
2016	262	0.809	0.618	1.006	0.268	41	100.00	14.689
2015	208	0.707	0.520	1.020	0.143	49	100.00	23.577
2014	188	1.063	0.714	1.319	0.125	40	100.00	48.000
2013	132	1.082	0.653	1.252	0.086	35	100.00	47.402
2012	128	1.298	0.532	1.640	0.179	28	96.43	62.863
2011	40	0.630	0.370	0.630	0.286	21	95.24	28.664

Figura 4.5: Indicadores clave de los últimos cinco años de la revista *Journal of Ambient Intelligence and Smart Environments*.

Rank

JCR Impact Factor

JCR Year	COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE			COMPUTER SCIENCE, INFORMATION SYSTEMS			TELECOMMUNICATIONS		
	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile
2019	91/137	Q3	33.942	113/156	Q3	27.885	65/90	Q3	28.33
2018	101/134	Q4	25.000	121/155	Q4	22.258	76/88	Q4	14.20
2017	105/132	Q4	20.833	125/148	Q4	15.878	75/87	Q4	14.36
2016	109/133	Q4	18.421	123/146	Q4	16.096	81/89	Q4	9.55
2015	104/130	Q4	20.385	108/144	Q3	25.347	62/82	Q4	25.00

Figura 4.6: Ranking, Cuartil y Factor de Impacto de la revista *Journal of Ambient Intelligence and Smart Environments* en los últimos años según categorías.

4.1.3 Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning - Journal of Supercomputing

En esta sección se indican los datos e indicadores de calidad de la revista *Journal of Supercomputing* donde se ha publicado el segundo artículo que compone el compendio de esta tesis doctoral, titulado *Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning*. En las figuras 4.7, 4.8 y 4.9 se muestran los datos de la revista, los indicadores de calidad y el factor de impacto.

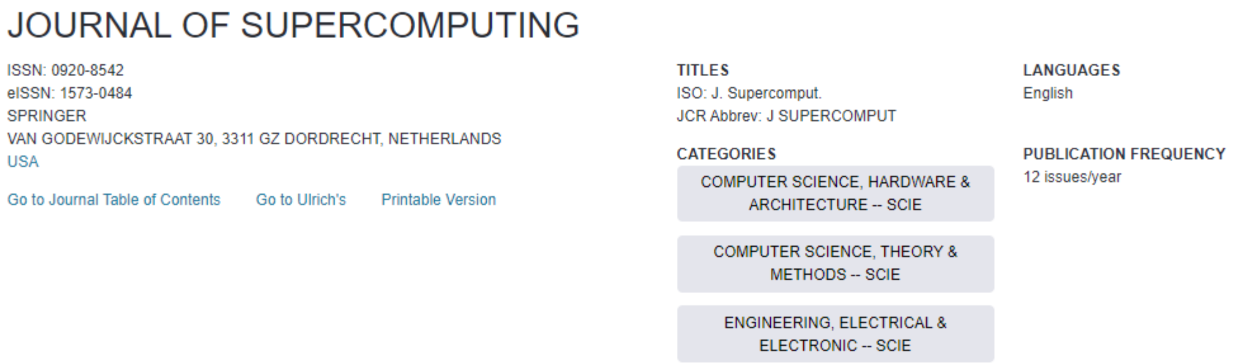


Figura 4.7: Información de la revista donde se ha publicado el artículo *Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning*

Current Year 2018 2017 All Years

Key Indicators - All Years Export

Customize columns

Year	Total Cites	Journal Impact Factor	Impact Factor without Journal Self Cites	5 Year Impact Factor	Immediacy Index	Citable Items	% Articles in Citable Items	Average JIF Percentile
	✓Trend	Trend	Trend	Trend	Trend	Trend	Trend	Trend
✓2019	3,734	2.469	2.045	2.025	0.642	380	99.21	60.456
2018	3,374	2.157	1.871	1.943	0.824	318	99.37	59.047
2017	2,421	1.532	1.295	1.495	0.538	264	100.00	50.666
2016	1,929	1.326	1.197	1.349	0.282	238	100.00	40.704
2015	1,236	1.088	0.890	1.013	0.115	209	99.52	51.531
2014	912	0.858	0.649	0.884	0.168	279	99.64	44.579
2013	694	0.841	0.627	0.870	0.155	277	99.64	44.231
2012	580	0.917	0.728	0.867	0.188	224	100.00	49.286
2011	335	0.578	0.496	0.523	0.155	103	100.00	28.518

Figura 4.8: Indicadores clave de los últimos cinco años de la revista *Journal of Supercomputing*.

Rank

JCR Impact Factor

JCR Year	COMPUTER SCIENCE, HARDWARE & ARCHITECTURE			COMPUTER SCIENCE, THEORY & METHODS			ENGINEERING, ELECTRICAL & EL		
	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF F
2019	24/53	Q2	55.660	31/108	Q2	71.759	123/266	Q2	
2018	22/53	Q2	59.434	35/105	Q2	67.143	132/266	Q2	
2017	25/52	Q2	52.885	44/103	Q2	57.767	153/260	Q3	
2016	35/52	Q3	33.654	52/104	Q2	50.481	163/262	Q3	
2015	23/51	Q2	55.882	47/105	Q2	55.714	147/257	Q3	

Figura 4.9: Ranking, Cuartil y Factor de Impacto de la revista *Journal of Supercomputing* en los últimos años según categorías.

4.1.4 A decision support system for water optimizationin anti-frost techniques by sprinklers - Sensors

En esta sección se indican los datos e indicadores de calidad de la revista *Sensors* donde se ha publicado el segundo artículo que compone el compendio de esta tesis doctoral titulado *A decision support system for water optimizationin anti-frost techniques by sprinklers*. En las figuras 4.10, 4.11 y 4.12 se muestran los datos de la revista, los indicadores de calidad y el factor de impacto.

Figura 4.10: Información de la revista donde se ha publicado el artículo *A decision support system for water optimizationin anti-frost techniques by sprinklers*

Current Year 2018 2017 All Years

Key Indicators - All Years

Export

Customize columns

Year	Total Cites	Journal Impact Factor	Impact Factor without Journal Self Cites	5 Year Impact Factor	Immediacy Index	Citable Items	% Articles in Citable Items	Average JIF Percentile
	✓Trend	Trend	Trend	Trend	Trend	Trend	Trend	Trend
✓2019	63,306	3.275	2.570	3.427	0.744	5,528	94.95	74.528
2018	46,222	3.031	2.295	3.302	0.695	4,481	95.83	68.404
2017	31,941	2.475	1.905	3.014	0.556	2,945	93.51	61.717
2016	23,901	2.677	2.078	2.964	0.412	2,190	95.80	70.576
2015	15,836	2.033	1.571	2.437	0.366	1,649	93.88	58.241
2014	12,379	2.245	1.724	2.474	0.363	1,256	91.08	64.535
2013	9,689	2.048	1.622	2.457	0.348	955	91.62	60.973
2012	7,082	1.953	1.488	2.395	0.321	950	88.21	60.358
2011	4,763	1.739	1.434	2.060	0.375	669	91.03	52.143

Figura 4.11: Indicadores clave de los últimos cinco años de la revista *Sensors*.

Rank

JCR Impact Factor

JCR Year	CHEMISTRY, ANALYTICAL			ENGINEERING, ELECTRICAL & ELECTRONIC			INSTRUMENTS & INSTRUMENTATION			EL
	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile	Rank	Quartile	JIF Percentile	Rank
2019	22/86	Q2	75.000	77/266	Q2	71.241	15/64	Q1	77.344	n/a
2018	23/84	Q2	73.214	n/a	n/a	n/a	15/61	Q1	76.230	12/26
2017	31/81	Q2	62.346	n/a	n/a	n/a	16/61	Q2	74.590	15/28
2016	25/76	Q2	67.763	n/a	n/a	n/a	10/58	Q1	83.621	12/29
2015	36/75	Q2	52.667	n/a	n/a	n/a	12/56	Q1	79.464	16/27

Figura 4.12: Ranking, Cuartil y Factor de Impacto de la revista *Sensors* en los últimos años según categorías.

4.2 Otras publicaciones

Durante los años de realización de la tesis doctoral, además de las publicaciones que componen el compendio de esta tesis, se han realizado otras colaboraciones que han permitido la publicación de otros estudios dentro de la línea que envuelve la tesis. A continuación se indican cada una de las publicaciones adicionales que se han llevado a cabo:

- Guillén-Navarro, M. A., Pereñíguez-García, F., & Martínez-España, R. (2017). IoT-based system to forecast crop frost. In 2017 International Conference on Intelligent Environments (IE) (pp. 28-35). IEEE.

  Esta publicación de un congreso internacional con categoría Core C, fue la primera que impulsó el desarrollo del primer artículo que componen esta tesis doctoral. En este artículo se realiza un estudio donde se analizan propuestas y alternativas para seleccionar los elementos necesarios para componer y desplegar un sistema IoT enfocado a la predicción de heladas en agricultura.

- Guillén-Navarro, M. Á., Cadenas, J. M., Garrido, M. C., Ayuso, B., & Martínez-España, R. (2018). A preliminary study to solve crop frost prediction using an intelligent data analysis process. In Intelligent Environments 2018 (pp. 97-106). IOS Press.

  Esta publicación de un congreso internacional con categoría Core C, realiza el estudio de alternativas para llevar al proceso del análisis inteligente de datos. El estudio realizado en esta publicación es el preámbulo del modelo inteligente de datos diseñado en la presente tesis doctoral.

- Guillén M. A., Cadenas J.M., Garrido M.C., Martínez-España R. (2019). Minimum temperature prediction models in plots to forecast frost in crops. Agriculture and Environment Perspectives in Intelligent Systems, 91-106, IOS Press.

  Esta publicación es un capítulo de libro, proveniente de la ampliación del artículo del congreso internacional de 2018. En este capítulo de libro se completa el estudio y experimentación presentada en el congreso.

- Guillén-Navarro, M. Á., Martínez-España, R., Bueno-Crespo, A., Ayuso, B., Moreno, J. L., & Cecilia, J. M. (2019). An LSTM Deep Learning Scheme for Prediction of Low Temperatures in Agriculture. In Intelligent Environments (Workshops) (pp. 130-138).

  En esta publicación de un congreso internacional categorizado como Core C, se desarrolla un modelo LSTM para predecir temperatura y prevenir heladas. Esta publicación es la base de la segunda publicación que componen el compendio de esta tesis doctoral.

- Guillén-Navarro M. A, Martínez-España R., Bueno-Crespo A., López B., Moreno J.L., Hernández D. and Cecilia J. M. (2019). Evaluation of different edge-computing platforms for the prediction of low temperatures in agriculture. 19th International Conference Computational and Mathematical Methods in Science and Engineering. In Press

  Esta publicación realizada en un congreso internacional abarca y estudia la ejecución en el Edge de los modelos propuestos en las publicaciones anteriores. La evaluación de los diferentes modelos en diferentes configuraciones ha servido como estudio premilimnar del tercer artículo que compone el compendio de esta tesis doctoral.

- Cadenas J.M., Garrido M.C., Martínez- España R., & Guillén-Navarro M.A.. Making decisions for frost prediction in agricultural crops in a soft computing framework. Computers and Electronics in Agriculture, 175:105587, 2020.

  Por último, tras ya disponer de un modelo fiable de predicción de heladas, se realiza una colaboración con otros autores expertos en el ámbito fuzzy para considerar las predicciones de las heladas desde un punto de vista discreto; en lugar de predecir la temperatura, en esta publicación tratamos de clasificar si una helada se va a producir o no, utilizando varias variables climatológicas y realizando un análisis preliminar de las variables más influyentes en la helada. El enfoque innovador realizado en este artículo es la inclusión de valores difusos que nos aportan una mayor cantidad de información y una visión más ampliada de los valores climáticos. Esta publicación se encuentra en *Computers and Electronics in Agriculture, revista indexada en JCR con un índice de impacto de 3.858, siendo la revista número 24 de 109 (Q1) en la categoría de Computer Science, Intercisciplinary Applications.*

Bibliografía

[1] Naveen Kumar Arora. Impact of climate change on agriculture production and its sustainable solutions, 2019.

[2] Michael Berthold and David J Hand. *Intelligent data analysis*, volume 2. Springer, 2003.

[3] Martin Bor, John Edward Vidler, and Utz Roedig. Lora for the internet of things. 2016.

[4] Francisco Javier Calvo Sánchez. Modelos atmosféricos. 2018.

[5] Organización de Naciones Unidas. Objetivos de desarrollo sostenible., 2021. https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/(Web; accedido el 26-05-20219)-.

[6] David Espín Sánchez et al. Riesgo de heladas por inversión térmica en la huerta de murcia: incidencia. 2015.

[7] Santo Padre Francisco. Carta encíclica laudato si, 2015. https://www.vatican.va/content/francesco/es/encyclicals/documents/papa-francesco_20150524_enciclica-laudato-si.html (Web; accedido el 26-05-2021).

[8] Jetse D Kalma, Gregory P Laughlin, Joseph M Caprio, and Paul JC Hamer. The occurrence of frost: types, distribution and prediction. In *The Bioclimatology of Frost*, pages 5–12. Springer, 1992.

[9] Ievgeniia Kuzminykh, Arkadii Snihurov, and Anders Carlsson. Testing of communication range in zigbee technology. In *2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, pages 133–136. IEEE, 2017.

[10] John N Landers and K Witte. Irrigation for frost protection. *Irrigation of Agricultural Lands*, 11:1037–1057, 1967.

[11] Jessica Lindblom, Christina Lundström, Magnus Ljung, and Anders Jonsson. Promoting sustainable intensification in precision agriculture: review of decision support systems development and strategies. *Precision Agriculture*, 18(3):309–331, 2017.

[12] Yoseph Linde, Andres Buzo, and Robert Gray. An algorithm for vector quantizer design. *IEEE Transactions on communications*, 28(1):84–95, 1980.

[13] Anupama Mahato. Climate change and its impact on agriculture. *International Journal of Scientific and Research Publications*, 4(4):1–6, 2014.

[14] Vinayak N Malavade and Pooja K Akulwar. Role of iot in agriculture. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 1(13):56–57, 2016.

[15] Yao Shifeng, Feng Chungui, He Yuanyuan, and Zhu Shiping. Application of iot in agriculture. *Journal of Agricultural Mechanization Research*, 7:190–193, 2011.

[16] Benny Vejlgaard, Mads Lauridsen, Huan Nguyen, István Z Kovács, Preben Mogensen, and Mads Sorensen. Coverage and capacity analysis of sigfox, lora, gprs, and

nb-iot. In *2017 IEEE 85th vehicular technology conference (VTC Spring)*, pages 1–5. IEEE, 2017.

[17] Felix Wortmann and Kristina Flüchter. Internet of things. *Business & Information Systems Engineering*, 57(3):221–224, 2015.

[18] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

[19] Naiqian Zhang, Maohua Wang, and Ning Wang. Precision agriculture—a worldwide overview. *Computers and electronics in agriculture*, 36(2-3):113–132, 2002.

[20] Zhi Zhou, Xu Chen, En Li, Liekang Zeng, Ke Luo, and Junshan Zhang. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*, 107(8):1738–1762, 2019.