

MODELO DE REFERENCIA PARA LA DETECCIÓN DE EVENTOS DE CONTAMINACIÓN INDUSTRIAL BASADO EN UNA RED DE SENSORES

A REFERENCE MODEL FOR DETECTION OF INDUSTRIAL POLLUTION EVENTS BASED ON A SENSORS NETWORK



AUTOR

JULIÁN MIGUEL ACEVEDO MORENO
Magister en Ing. de Sistemas y Computación
*Pontificia Universidad Javeriana
julian.acevedo@javeriana.edu.co
COLOMBIA

AUTOR

EDGAR ENRIQUE RUIZ GARCÍA
Magister en Ing. de Sistemas y Computación
*Pontificia Universidad Javeriana
Jefe de Sección
Depto. de Ingeniería de Sistemas
Profesor de Planta
eruiz@javeriana.edu.co
COLOMBIA

*INSTITUCION

PONTIFICIA UNIVERSIDAD JAVERIANA
PUJ
Universidad Privada
Calle 40 Nro. 5-50
Bogotá, D.C
COLOMBIA

AUTOR

RICARDO HJALMAR GONZALEZ GARCÍA
Dr. en Computación
**Universidad Sergio Arboleda
*Pontificia Universidad Javeriana
Escuela de Ciencias Exactas e Ingeniería
ricardo.gonzalez@usa.edu.co
COLOMBIA

AUTOR

MARIELA JOSEFINA CURIEL HUERFANO
Dra. en Informática
*Pontificia Universidad Javeriana
Directora de Carrera
Depto. de Ing. de Sistemas
Profesor de Planta. Investigador
mcuriel@javeriana.edu.co
COLOMBIA

**INSTITUCION

UNIVERSIDAD SERGIO ARBOLEDA
USA
Universidad Privada
Calle 74 Nro. 14-14
Bogotá, D.C.
COLOMBIA

INFORMACIÓN DE LA INVESTIGACIÓN O DEL PROYECTO: El presente artículo describe el resultado de la investigación realizada por el Ing. Julián Acevedo como parte de los requisitos establecidos para obtener el Título de Magister en Ingeniería de Sistemas y Computación de la Pontificia Universidad Javeriana. El trabajo hace parte del grupo de Investigación SIDRE del Departamento de Ingeniería de Sistemas de la Pontificia Universidad Javeriana. El problema a resolver fue el de diseñar un modelo de referencia para el procesamiento de datos relacionados con la contaminación industrial, provenientes de una red de sensores. El trabajo fue dirigido por la Dra. Mariela Curiel y a lo largo del proyecto se contó también con la colaboración de los profesores Edgar Enrique Ruiz y Ricardo González.

RECEPCIÓN: Mayo 31 de 2016

ACEPTACIÓN: Septiembre 6 de 2016

TEMÁTICA: Sistemas Inalámbricos y Móviles

TIPO DE ARTÍCULO: Artículo de Investigación e Innovación

Forma de citar: Acevedo, J., González, R., Ruiz, E. y Curiel, M. (2016). Modelo de referencia para la detección de eventos de contaminación industrial basado en una red de sensores. En R, Llamosa Villalba (Ed.). Revista Gerencia Tecnológica Informática, 15(43), 19-35. ISSN 1657-8236.

RESUMEN ANALÍTICO

La contaminación industrial es un aspecto que está afectando en forma importante la sostenibilidad del planeta. Para poder manejar la contaminación, el primer paso es detectarla tan pronto como ésta ocurra, de ahí la importancia de la recolección oportuna de datos. Actualmente la tecnología nos da un gran soporte para esta tarea porque se cuenta con redes de sensores que, al estar en el ambiente, permiten recolectar grandes cantidades de datos. Si estos datos son procesados de forma adecuada, pueden ayudar en la detección temprana de focos de contaminación. No obstante, hay que enfrentar el problema de manejar de forma efectiva una gran cantidad de datos provenientes de dispositivos heterogéneos, que incluyen tanto a los sensores tradicionales o motas de las redes inalámbricas de sensores, como a los sensores ubicados en teléfonos inteligentes. El manejo de esta heterogeneidad motivó a diseñar un modelo de referencia que sirviera de guía para realizar las funciones de detección, almacenamiento, procesamiento y visualización/publicación de los datos, usando las tecnologías más adecuadas. La idea es que este modelo de referencia pueda posteriormente ser instanciado con componentes de hardware y software específicos, que respondan a las necesidades de determinadas empresas o entes gubernamentales, interesados en tomar acciones preventivas o correctivas sobre la contaminación ambiental. En este artículo se presenta el modelo de referencia propuesto y se describe un prototipo que instancia este modelo. El prototipo desarrollado está enfocado en la medición de la contaminación auditiva en una empresa del sector automotriz en Bogotá. La información obtenida con el prototipo fue utilizada para tomar medidas preventivas para el control de focos de contaminación auditiva por parte de la empresa.

PALABRAS CLAVES: Redes inalámbricas de sensores, Teléfonos inteligentes, Monitorización, Minería de datos, Contaminación industrial.

ANALYTICAL SUMMARY

Industrial pollution is significantly affecting the sustainability of the planet. A data collection process allows us to review the contamination levels, and take actions in a timely manner. Nowadays wireless sensor networks support the collection of large amounts of environmental data. If these data are properly processed, they can be used in the detection of pollution sources, and in the design of preventive -or corrective- strategies to deal with it. However, this requires to effectively manage large amount of data obtained from heterogeneous devices, including not only data from traditional sensors or motes of wireless sensor networks, but also from sensors within Smartphones. In order to handle this heterogeneity, we design a reference model that serves as a guide to combine the technologies of storage, processing and presentation of the data. This model can then be instantiated with specific hardware and software components that meet the needs of companies or government agencies interested in dealing with environmental pollution. In this article, the proposed reference model is presented, in conjunction with a prototype that can be used to solve problems of noise pollution in an automotive company located in Bogota.

KEYWORDS: Wireless sensor networks, Smartphones, Monitoring, Data mining, Industrial pollution.

INTRODUCCIÓN

La contaminación industrial es un aspecto que está afectando en forma importante la sostenibilidad del planeta [1]. En [2] se define contaminación industrial como *“la incontrolable degradación del medio ambiente por el crecimiento industrial no planeado, el cual está directamente relacionado con la descarga a la atmósfera*

de sustancias contaminantes sin ningún control de la cantidad, densidad y composición química”. Una de las formas de enfrentar los problemas de contaminación es por medio de regulaciones gubernamentales. Por ejemplo, a través del Decreto 948 del 5 de junio de 1995, el Estado colombiano reglamenta las emisiones al medio ambiente en sus diferentes formas como: ruido, olores, contaminación del aire, contaminación del agua,

entre otros. La Resolución 610 del 24 de marzo de 2010, también establece parámetros para la calidad del aire en Bogotá. En estas regulaciones se establecen los niveles mínimos y máximos de los diferentes tipos de emisiones y las penalidades en caso de un incumplimiento por parte de las empresas generadoras de emisiones. Dado que la generación de datos e información permite la revisión de estos niveles, el Estado también ha creado infraestructura para su recolección. Tal es el caso de la Red de Monitoreo de Calidad de Aire de Bogotá (RMCAB), que cuenta con 13 estaciones fijas y una estación móvil, que permiten recolectar información sobre la concentración de contaminantes en la atmósfera bogotana [3]. No obstante, por contar con un número limitado de estaciones, es posible que no se detecten todos los focos de contaminación existentes o que no se detecten con la suficiente anticipación. Por esta razón, es deseable contar con redes que posean un mayor número de sensores, ampliamente distribuidos.

Hoy en día la tecnología nos brinda una solución a este problema. Por un lado, existen las redes inalámbricas de sensores (WSN, por sus siglas en inglés) compuestas por un conjunto de nodos sensores de bajo costo y con capacidades de monitorizar el entorno. Por otro lado, los teléfonos inteligentes, ampliamente utilizados en la actualidad, también poseen sensores capaces de capturar, clasificar y transmitir imágenes, audio, ubicación y otros datos, de forma interactiva o autónoma. Los teléfonos equipados con estos sensores se convierten en equipos perfectos para recoger información sobre el medio ambiente utilizando un enfoque de detección participativa (*participatory sensing*) [4].

Después de la recolección, uno de los desafíos es el manejo eficiente de este gran volumen de datos y la heterogeneidad presente en los dispositivos, en los medios de comunicación y en la naturaleza de la información recolectada. Aparte de la diversidad de tecnologías, en el proceso de recolección de datos, también intervienen factores inherentes a la confiabilidad de la información, cuando por ejemplo se recogen datos generados por sensores defectuosos, mal calibrados, o que incluso puedan llegar a ser manipulados maliciosamente [5].

Para tratar de minimizar la complejidad de estos procesos, se han desarrollado *middlewares*, cuya principal función es ofrecer transparencia respecto a las plataformas base, ocultando las diferencias en las características del tipo hardware empleado (ejemplos: Mate [6], Magnet [7], Cougar [8], SINA [9], DsWare [10] y Mosquito [11]). No obstante, estos sistemas no son compatibles con todos los tipos de nodos existentes en las redes de sensores y la interoperabilidad entre ellos es bastante reducida. Lo anterior pone en evidencia la necesidad de desarrollar un modelo de referencia que permita que los sistemas

middleware existentes y otros componentes de software pertenecientes a las redes de sensores, se comuniquen con independencia de los lenguajes de programación, sistemas operativos y plataformas de hardware. Este modelo debe también sugerir la forma de almacenar y procesar la información para que, tanto usuarios como aplicaciones, puedan extraer conocimiento útil de forma oportuna. El trabajo que se describe en este artículo es un modelo de referencia con las características mencionadas. La idea es que este modelo general pueda ser instanciado con componentes específicos, que respondan a las necesidades de determinadas empresas o entes gubernamentales interesados en resolver problemas de contaminación ambiental.

El artículo tiene la siguiente estructura: La sección 1 describe el marco conceptual propio de la investigación. Luego, en la sección 2 se presenta el modelo de referencia propuesto, en la sección 3 se describe una instancia del modelo de referencia a través del desarrollo de un prototipo para la medición de la contaminación auditiva en una empresa del sector automotriz en Bogotá. En la sección 4 se discuten los resultados obtenidos al implementar una instancia del modelo de referencia planteado en esta investigación. En la sección 5 se describen trabajos relacionados y, finalmente, en la sección 6 se presentan las conclusiones y los trabajos futuros.

1. MARCO TEÓRICO

Para explicar los conceptos más relevantes del trabajo de investigación, la sección se ha dividido en tres subsecciones: En la subsección 1.1 se describe la definición y las principales características de un modelo de referencia. Luego la subsección 1.2 contiene una descripción de las motas y de los teléfonos inteligentes. Finalmente, la subsección 1.3 presenta una breve descripción de las técnicas de minería de datos y texto que se pueden usar para extraer información de los datos recolectados.

1.1 MODELO DE REFERENCIA

Un Modelo de Referencia es un marco de referencia abstracto para entender el significado de las relaciones entre entidades de algún ambiente determinado. El modelo debe ser independiente de estándares específicos, tecnologías e implementaciones [12] y no debe contener los elementos en sí mismos, sino una descripción de su funcionalidad. Finalmente, un modelo de referencia es un mecanismo para entender un determinado problema sin proveer la solución. Es por ello que en este artículo separamos el modelo de una instancia del mismo.

1.2 REDES DE SENSORES Y TELÉFONOS INTELIGENTES

Las Redes Inalámbricas de Sensores o *Wireless Sensor Networks WSN* son un tipo particular de red, compuestas por [13]: un conjunto de nodos sensores encargados de monitorear el entorno, un nodo recolector o *Sink*, que concentra los datos tomados por los nodos sensores y que también puede hacer la labor de un *Gateway* para comunicar la información de los nodos sensores hacia otros sistemas. Los nodos sensores son también llamados motas o *motas*, en referencia al proyecto *Smart Dust* [14], que fue uno de los primeros en usar este tipo de dispositivos. Las motas pueden intercambiar información entre sí, usando redes inalámbricas, y pueden contar con una fuente de energía propia que les permita operar de forma autónoma por periodos relativamente largos de tiempo. Las redes inalámbricas de sensores pueden tener la capacidad de auto-reconfigurarse en caso de la falla o caída de un nodo y se caracterizan por su escalabilidad, flexibilidad, autonomía e interoperabilidad. Por sus características de autonomía, estas redes permiten la medición de variables aún en lugares de difícil acceso.

1.2.1 Características de las motas

Los nodos sensores o motas tienen la capacidad de capturar información del entorno, procesarla y enviarla a los nodos recolectores, para cumplir este propósito, están compuestos por: un procesador, memoria, un grupo de sensores, una interfaz de red inalámbrica y una fuente de alimentación eléctrica [15].

Entre los **procesadores** desarrollados para motas encontramos los microcontroladores Intel 8051, el PIC, el ARM7 y el Atmel, entre otros. Estos dispositivos poseen dos tipos de memoria, la memoria RAM y una memoria flash no volátil, para que no se pierda la información almacenada cuando el equipo se apague, ambos tipos de memoria pueden ser utilizadas para almacenar los datos recolectados por la aplicación, el sistema operativo y la configuración del dispositivo.

Los sensores son dispositivos de hardware que producen una respuesta medible ante un cambio de estado físico en el área que están monitorizando; por ejemplo pueden detectar cambios en la temperatura o la presión [16]. En los nodos sensores la señal analógica continua que detectan los sensores es digitalizada por un convertidor analógico digital (ADC) y enviada a un controlador para ser procesada. Los sensores suelen ser de pequeño tamaño, y se destacan también por su bajo consumo de energía. Diferentes tipos de sensores se pueden integrar en una misma plataforma para el monitoreo de la contaminación, como puede ocurrir en

una placa Arduino [17], donde diversos tipos de sensores se pueden usar para medir la humedad, la temperatura, el monóxido de carbono, el dióxido de carbono, el dióxido de azufre, el ozono y dióxido de nitrógeno.

La fuente de poder brinda la energía para la operación de los diferentes componentes que están presentes en la mota. Esta energía se puede obtener a través de baterías, paneles solares o mediante una conexión directa a una fuente de electricidad externa.

En el proceso de captura de la información y su entrega a un sistema central, intervienen varios elementos. La captura la realiza la mota y luego, con el uso de un protocolo de comunicaciones, se puede entregar la información a un nodo coordinador, quien a su vez puede retransmitirla hasta llegar al sistema central, buscando la ruta más adecuada, mediante el uso de algún protocolo de enrutamiento [18]. A continuación se explican los medios físicos y protocolos involucrados en la comunicación.

1.2.2 Medios para la comunicación

Físicos: Las motas realizan su comunicación vía radio usando normalmente bandas ISM (*Industrial Scientific and Medical*) no licenciadas. Las tecnologías inalámbricas que pueden ser usadas son: Wi-Fi a/b/g/n/ac (IEEE 802.11), *Bluetooth* (IEEE 802.15.1) y otras tecnologías basadas en el estándar IEEE 802.15.4. La más conocida de estas es *Zigbee*, la cual está enfocada en hacer un muy bajo consumo de energía en la transmisión y recepción de datos. La velocidad máxima alcanzada por las motas con *Zigbee* es de 250 Kbps, con *Bluetooth*, dependiendo de la versión utilizada, se pueden lograr velocidades que van desde 1 Mbps y pueden llegar a superar los 24 Mbps, finalmente con Wi-Fi es posible obtener velocidades que van de 1 Mbps a cientos de Mbps en las versiones más recientes.

Protocolos de Enrutamiento: Los protocolos de enrutamiento se dividen en dos grupos: los proactivos y los reactivos. Los protocolos proactivos continuamente envían información de enrutamiento para mantener las rutas actualizadas. Entre los protocolos proactivos se encuentran el OLSR (*Optimized Link-State Routing Algorithm*) y el DSDV (*Destination-Sequenced Distance Vector*). Los protocolos reactivos sólo crean las rutas cuando éstas se requieren para enviar un mensaje. Ejemplos de estos protocolos son el AODV (*Ad hoc On Demand Distance Vector Routing*) y el DSR (*Dynamic Source Routing*). Los dos tipos de protocolos se usan en las WSN según la función que el sensor vaya a cumplir, por ejemplo, si la medición la realiza un sensor que está en movimiento se puede usar un protocolo reactivo ya que la ruta puede cambiar dependiendo del lugar

donde se encuentre; si por el contrario el sensor está estático, se puede usar un protocolo de tipo proactivo que pudiera consumir menos recursos para enviar rápidamente un mensaje por alguna ruta preestablecida previamente.

1.2.3 Arquitectura de software

La capa más baja del software para una WSN la conforma el sistema operativo. Entre los sistemas operativos desarrollados para las motas se encuentran: PalOS, SOS, TinyOS, MANTIS y *BTnut*, tal como se describe en [19]. La capa siguiente de software es el *middleware*, cuya función principal es la de abstraer de los servicios de las capas posteriores toda la problemática asociada a la heterogeneidad que puedan tener los sensores. También se encarga de proveer mecanismos que mantengan los parámetros de calidad de servicio (especialmente la eficiencia, la tolerancia a fallos y el consumo de energía) en los niveles esperados. Algunos ejemplos de *middleware* para WSN son: Mate [6], Magnet [7], Cougar [8], SINA [9], DsWare [10], Impala [20], Zuma [21] y Mosquitto [11].

1.2.4 Teléfonos Inteligentes

Aunque las redes inalámbricas han tenido un éxito importante en los últimos años, otro dispositivo electrónico que ha tenido un desarrollo aún más interesante es el *Smartphone* o *Teléfono Inteligente*, que destaca no sólo por sus capacidades, sino también por su amplio uso. En 2015 se vendieron alrededor de 1.400 millones de teléfonos inteligentes en el mundo [22] según reportes del *Gartner Group*. La incorporación en estos dispositivos de otras tecnologías inalámbricas, adicionales a las redes celulares, como Wi-Fi, *Bluetooth* y NFC (*Near Field Communication*), ha permitido mejorar su capacidad de comunicación como dispositivos autónomos.

Tanto la movilidad de estos dispositivos, como los sensores que poseen los hacen muy atractivos para las tareas de detección de eventos externos, ya que es posible ampliar el área de cobertura con respecto a la que puede brindar una WSN tradicional que use tecnología *Zigbee*. Estos sensores, que se han incluido inicialmente en los teléfonos inteligentes para darle una mejor experiencia al usuario y aumentar la competitividad de los dispositivos en el mercado, pueden tener otros importantes usos en procesos de monitorización ambiental e industrial.

Entre los sensores con los que puede contar un teléfono inteligente se encuentran los acelerómetros, los giroscopios, los micrófonos, los sistemas de

posicionamiento global o *GPS* (*Global Positioning System*) y los sensores fotográficos [23], entre otros, tal como se puede apreciar en la Figura 1. Estos sensores permiten la captura de información de fenómenos físicos directamente por parte del teléfono y brindan la oportunidad de que los teléfonos puedan ser utilizados como nodos sensores en una WSN.

FIGURA 1. Sensores presentes en los *Smartphones*



A pesar de que el consumo de energía no ha sido una característica prioritaria en el diseño de los teléfonos inteligentes actuales, como si lo ha sido para los nodos sensores que usan *Zigbee*, dicho consumo es mucho más bajo que el que realiza un equipo de computación tradicional. A pesar de esto, para los fabricantes de teléfonos, el hecho de que éstos puedan operar por largos periodos de tiempo, ha sido un punto muy importante en su proceso de comercialización, ya que les permite mayor autonomía. Es por ello que muchos de los teléfonos poseen opciones para desactivar temporalmente algunos de sus componentes, como los sensores de posicionamiento (GPS), algunas de sus interfaces de red o incluso la conexión a redes 2G o 3G, para de esta forma ahorrar energía y funcionar de forma ininterrumpida por periodos más largos de tiempo [24].

La idea de usar los teléfonos inteligentes como sensores no es nueva, hay algunos trabajos en los cuales los teléfonos inteligentes han sido empleados formando parte de las WSN sólo de forma parcial y cumpliendo los roles de nodo recolector (*Sink*) [25] o de *Gateways* [24] [26]. En González y Huerta [27] se presenta la utilización de teléfonos inteligentes, en el rol de nodos sensores en una WSN. Otras empresas se han dedicado a mejorar la integración entre los sensores de las WSN y los *Smartphones*, tal es el caso de las compañías Libelium [28] y Sensorcon [29] cuyas motas

pueden medir diferentes aspectos del medio ambiente (temperatura, humedad, aire, presión atmosférica, etc.) y enviar los datos a un teléfono inteligente vía Wi-Fi.

1.3 MINERÍA DE DATOS Y DE TEXTO

La minería de texto es el descubrimiento de información previamente desconocida extrayéndola de forma automática a partir de diferentes fuentes escritas. La minería de datos se define como el proceso de exploración y análisis, usando medios automáticos o semi-automáticos, de grandes volúmenes de datos para descubrir en ellos reglas o patrones significativos [30]. Aunque la minería de datos y la minería de texto comparten algunas características, la minería de datos está diseñada para trabajar con datos estructurados y la minería de texto sobre datos no estructurados o semi-estructurados, como son los datos generados por las motas y la información que pudieran registrar usuarios en teléfonos inteligentes.

Las técnicas más representativas de la minería de datos son:

Algoritmos de clasificación: predicen una o más variables discretas, basándose en otros atributos del conjunto de datos. Ejemplos de estas técnicas son los árboles de decisiones, las redes neuronales, los clasificadores bayesianos y las máquinas de vectores de soporte (*support vector machines*).

Algoritmos de regresión: predicen o aproximan la relación de dependencia entre una variable dependiente y un conjunto de variables independientes. Entre las técnicas de regresión se encuentran la regresión lineal global o clásica y la regresión lineal ponderada localmente. Existen también técnicas de regresión no lineal como la regresión logarítmica y los árboles de regresión.

Algoritmos de segmentación: dividen los datos en grupos, o clústeres, de elementos que tienen propiedades similares. Los algoritmos de *clustering* o agrupamiento se dividen en: a) Jerárquicos (a sus vez divididos en aglomerativos o divisivos) y b) no Jerárquicos, en los que el número de grupos se determina de antemano y las observaciones se van asignando a los grupos en función de su cercanía. Un ejemplo de estrategia de *clustering* jerárquico es el MST (*minimum spanning tree*). Entre los algoritmos no jerárquicos se encuentran el *k-mean* y *k-mediod*.

Algoritmos de asociación: buscan correlaciones entre diferentes atributos de un conjunto de datos. Existen diversos algoritmos que realizan búsquedas de reglas de asociación tales como *Apriori*, *Eclat* y *FP-Growth*.

Algoritmos de análisis de secuencias: resumen secuencias o episodios frecuentes en los datos (*frequent patterns*). Ejemplos: GSP y SPADE (*Sequential Pattern Discovery using Equivalent Class*)

Existe mucha bibliografía sobre el tema, para más detalles se pudiera consultar [30] y [31].

Los pasos principales para el procesamiento de los datos, utilizados en el modelo de referencia propuesto se describen a continuación:

Pre-procesamiento o limpieza de los datos: consiste en preparar la información que se va a analizar para que sirva de entrada a los algoritmos de minería de datos o texto. En esta etapa, por ejemplo, se eliminan caracteres especiales, espacios en blanco, etc.

Selección de la técnica de minería de datos y su aplicación: una vez que se tienen los datos organizados, hay que definir los algoritmos que se van a utilizar en su tratamiento. Pueden usarse varias técnicas para generar distintos modelos, aunque generalmente cada técnica obliga a un pre-procesado diferente de los datos.

Evaluación: los resultados obtenidos deben de ser sometidos a comprobación, es decir verificar que están libres de errores, ratificar que son útiles para los objetivos perseguidos, etc.

2. MODELO DE REFERENCIA

Con el objetivo de manejar las funciones de almacenamiento, procesamiento y visualización/publicación de los datos se diseña el modelo de referencia que se describe a continuación.

Las principales características del modelo propuesto son: a) Considera no sólo la heterogeneidad de los dispositivos y medios para la recolección de datos, sino la disparidad de los datos recolectados (diferentes formatos, unidades de medida, resoluciones, etc.) b) Especifica la forma de manejar eficientemente grandes volúmenes de datos. c) Considera la inclusión de diferentes tipos de técnicas de minería de datos o de texto, que permitan obtener información útil para la toma de decisiones. d) Provee los medios para entregar resultados en-línea (a tiempo real o de forma periódica) y a través de archivos históricos. f) Ofrece un API para que la información pueda ser accedida desde diferentes medios (aplicaciones, redes sociales, etc.) g) Provee la información de forma gráfica y a través de los reportes solicitados por los entes gubernamentales colombianos. h) Por último, en lo posible, los componentes del modelo deberían instanciarse con herramientas de código abierto o de elaboración propia ya que puede ser

necesario introducir modificaciones para el ensamblaje de los componentes.

Las capas del modelo de referencia propuesto se presentan en la Figura 2 y se describen a continuación:

La Capa de Captura de Datos tiene dos funciones: recolectar y almacenar de forma eficiente la información de contaminación ambiental que proviene de los diferentes tipos de sensores. **La Capa de Procesamiento** realiza tareas de preparación y clasificación de la información usando técnicas de minería de datos y/o texto. **La Capa de Visualización/ Publicación** tiene como función el control del acceso a la información procesada, así como su publicación por los diferentes medios ya mencionados.

FIGURA 2: Capas del Modelo de Referencia



A continuación se presentan y describen en forma detallada los componentes que integran cada una de estas capas, tal como se observan en la Figura 3. No sólo se describirán las funciones de los componentes sino también las posibles tecnologías que se pueden usar en cada caso.

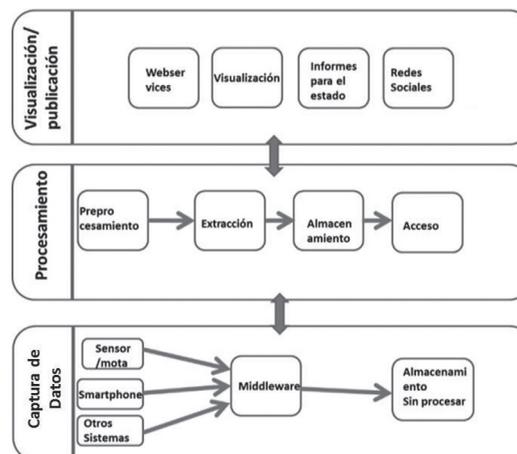
2.1 CAPA DE CAPTURA DE DATOS

En esta capa se encuentran todas las fuentes de entrada de la información (sensores y otros sistemas), así como los componentes para su tratamiento inicial y almacenamiento.

2.1.1 Sensores y otros sistemas

Este grupo de componentes abarca a las motas, a los sensores ubicados en los *Smartphone* y a otras plataformas de captura de indicadores de contaminación ambiental. Las motas tienen la capacidad de capturar la información del entorno geo-referenciada, procesarla y enviarla a un sistema central para su análisis. La información puede ser enviada usando protocolos de transmisión inalámbrica como Wi-Fi, o *Zigbee*. Los *Smartphones* pueden usar Wi-Fi, *Bluetooth* o redes de datos 2G, 3G o 4G. Otras plataformas de recolección de datos pueden usar una mezcla de estos tipos de redes o incluso conexiones más tradicionales como redes locales cableadas del tipo Ethernet o servicios de banda ancha sobre los cuales se puedan implementar esquemas similares a los servicios web (*web services*), como lo hace la plataforma *da-sense* [32].

FIGURA 3: Componentes de cada Capa del Modelo de Referencia



2.1.2 Middleware

Las principales funciones que debe cumplir el *middleware* del modelo de referencia planteado son: a) Esconder la heterogeneidad, soportando la captura de la información por medio de los diferentes protocolos de comunicación utilizados por los sensores, tales como Wi-Fi, IPv6, *Zigbee*; el *middleware* debe estar en la capacidad de procesar los distintos protocolos y enviar la información al siguiente componente de la capa. b) Proveer mecanismos que garanticen los parámetros de calidad de servicio, especialmente la eficiencia y el consumo de energía. c) Estar en capacidad de utilizar servicios de otras plataformas que también recolectan datos de contaminación ambiental.

Una de las formas de manejar la heterogeneidad es a través del uso de máquinas virtuales. Los *middlewares* Mate [6] y Magnet [7] emplean este enfoque. Para el mismo problema, otros sistemas utilizan una base de datos virtual relacional donde los sensores son las tablas y la consulta de información sobre los sensores se hace con instrucciones SQL como ocurre con los sistemas Cougar [8], SINA [9], DsWare [10].

Los problemas de la eficiencia y bajo consumo de energía de los dispositivos pueden abordarse desde el modelo de programación que ofrece el *middleware*. Por ejemplo, Impala [20] hace uso de agentes móviles y comunicación asincrónica. También ofrece mecanismos eficientes para la actualización de las redes, soporta aplicaciones dinámicas, es tolerante a fallas y su *overhead* es muy bajo. No obstante, no soporta sensores heterogéneos.

Como se puede apreciar, hoy día no hay muchas herramientas que resuelvan todos los requerimientos planteados para el *middleware* del modelo de referencia.

Las implementaciones del modelo tendrán que dar prioridad a unos requerimientos sobre otros, en función del problema de contaminación a resolver.

Otra opción es modificar los *middlewares* existentes usando *wrappers* o adaptadores para generar una interfaz común para el intercambio de información entre los diferentes componentes del sistema. En este sentido, uno de los protocolos de comunicación que está tomando mucha fuerza en dispositivos de baja capacidad de procesamiento y autonomía es MQTT (*Message Queue Telemetry Transport*) [33], un protocolo *machine-to-machine* (M2M), soportado en muchos tipos de sensores. Se trata de un protocolo de transporte muy liviano del tipo publicador/suscriptor donde la comunicación puede ser de uno a uno, o uno a muchos. Su modelo de suscripción está basado en tópicos a los que se registran sus suscriptores para recibir información proveniente de los nodos sensores. Otro protocolo específico para nodos con capacidades restringidas es el CoAP [34][35]. Se trata de un protocolo a nivel de aplicación pensado para ser usado entre dispositivos electrónicos simples, como sensores de baja potencia. Con este protocolo estos dispositivos pueden comunicarse de forma interactiva a través de internet; su modelo de interacción cliente/servidor es similar al de HTTP con la diferencia de que CoAP realiza los intercambios de mensajes de forma asíncrona por medio del protocolo de transporte UDP

2.1.3 Almacenamiento sin procesar

El modelo plantea realizar el almacenamiento de los datos crudos, provenientes de los sensores, en una base de datos no relacional como *MongoDB*, *HBase*, *Hypertable*, *Cassandra* o *RIAK*. Con este enfoque se logra un mejor rendimiento ya que este tipo de bases de datos se diseñó para manejar grandes volúmenes de información a una gran velocidad. Otras características importantes son su capacidad de crecer de forma horizontal y el manejo de muchas conexiones simultáneas.

El flujo de los datos en esta capa se realiza de la siguiente forma: el proceso se inicia en la recolección de los datos desde cualquiera de las tres fuentes de datos descritas anteriormente; luego, los datos son recibidos por el *middleware* que debe ser capaz de entender los diferentes protocolos y de dirigir los datos y metadatos a las bases de datos correspondientes.

2.2 CAPA DE PROCESAMIENTO

En esta capa se encuentran todas las funciones inherentes al almacenamiento de datos y las técnicas para producir información útil para los usuarios. A continuación se describen las funciones de los diferentes componentes:

2.2.1 Preprocesamiento

Los datos que provienen de los sensores pueden llegar con algunos problemas, como por ejemplo caracteres extraños, información incompleta o exceso de espacios en blanco. Es por esta razón que el modelo debe contemplar la inclusión de un componente que se encargue de preparar la información antes de ser procesada por las técnicas de minería de datos.

Entre las funciones de este componente están el realizar la limpieza de los datos y llevar la información a un formato único. Por limpieza se entiende eliminar caracteres extraños y espacios adicionales. También, deberá uniformar las escalas y realizar ajustes en el formato de algunos tipos de datos, como por ejemplo las fechas, que suelen venir en diferentes formatos. Finalmente, debe convertir mediciones a rangos de datos; por ejemplo, si se está midiendo ruido se pueden definir escalas de decibeles para poderlas segmentar por grupos (ejemplo: bajo, medio y alto) para su posterior análisis.

Después de pasar por este módulo, la información quedará normalizada para ser visualizada desde la capa superior. Estará disponible para la consulta directa de valores como el lugar de la medición, las fechas, los datos clasificados por grupos, etc. y/o puede pasar al Componente de Extracción donde se aplicarán técnicas que permitan descubrir patrones en los datos.

2.2.2 Extracción

Dependiendo de las necesidades particulares del problema que se está resolviendo, en este componente se pueden implementar técnicas o utilizar sistemas que apliquen técnicas de minería de datos para extraer conocimiento útil, como por ejemplo: *Weka*, *RapidMiner*, *Orange*, y *Knime*. Todas estas herramientas son de código libre y abierto.

Tres ejemplos de técnicas de minería de datos que pudieran ser de utilidad son: a) algoritmos de segmentación (*clustering*) para obtener patrones del comportamiento del agente contaminante, por ejemplo patrones del comportamiento del ruido según el tipo de actividad industrial; b) algoritmos de análisis de secuencias o episodios frecuentes (*frequent patterns*), para descubrir hechos comunes dentro de un determinado conjunto de datos correspondientes a un periodo determinado de tiempo y, c) técnicas de clasificación para clasificar el ruido o la temperatura en distintos niveles.

El uso de estas técnicas permitirá visualizar focos de contaminación cuando se combinan determinados tipos

de factores, para hacer luego predicciones sobre los comportamientos detectados. Las técnicas soportadas se deberían poder aplicar de forma individual o combinada, según el escenario donde se vayan a usar.

2.2.3 Almacenamiento

Este componente es el encargado de almacenar tanto los datos después de ser pre-procesados, como la información que resulta después de la aplicación de las diferentes técnicas de minería de datos. Una vez que la información se encuentra en este repositorio queda disponible para ser consultada por los módulos de la siguiente capa.

2.2.4 Acceso

Este componente gestiona la autenticación y la seguridad de la información, así como todos los datos relacionados con los usuarios, claves, perfiles y roles dentro del sistema.

2.3 CAPA DE VISUALIZACIÓN/PUBLICACIÓN

En esta capa se encuentran todas las funciones inherentes a la presentación de los datos, sus componentes realizan las siguientes funciones:

2.3.1 Visualización

Este componente se encarga de facilitar la **presentación** de la información generada por el sistema, aprovechado el hecho de que la información puede estar geo-referenciada, ya que al usarse teléfonos inteligentes, que cuentan con sensores GPS, la información puede asociarse al lugar donde ocurre la detección, lo que facilita la ubicación de eventos y elementos en mapas y diagramas.

Mediante este componente el usuario puede interactuar con una interfaz web donde podrá solicitar la información que requiera de forma gráfica, por ejemplo a través de mapas de calor y diagramas polares (*polar plots*). Para apoyar estos procesos se pudieran usar herramientas como *Google Maps* y *OpenStreetView*. La información que manejarán los usuarios a través de las interfaces de Visualización podrá ser en tiempo real, o por rangos de tiempo, según sean sus necesidades.

2.3.2 Web Services

El modelo de referencia debe permitir consultas de usuarios o de otros servicios y aplicaciones a través de *web services*. De esta forma un sistema que implemente el modelo de referencia puede ser empleado como un

componente o bloque de construcción que forme parte de sistemas más complejos y elaborados.

2.3.3 Informes para el Estado

Los requerimientos indican que la información sobre impactos ambientales debe presentarse a las empresas según los formatos que exigen los entes gubernamentales.

2.3.4 Redes Sociales

Este componente permitirá conectar al sistema que implemente el modelo de referencia a redes sociales, como Twitter y Facebook, para hacer notificaciones y seguimiento de determinados indicadores en zonas específicas. Así, la información podrá llegar a los usuarios de manera más rápida y eficiente, y éstos podrán consultarla donde quiera que se encuentren.

3. INSTANCIACIÓN DEL MODELO DE REFERENCIA

Para realizar la prueba de concepto se realizó la implementación de un prototipo y se desarrolló un conjunto de pruebas de campo en una empresa perteneciente al sector automotriz. Durante estas pruebas se realizaron mediciones de la contaminación auditiva en las instalaciones de los talleres de la empresa donde se reparan vehículos automotores

Debido a su naturaleza, la reparación de un vehículo es una actividad que puede generar altos niveles de ruido. Recientemente, la empresa ha mostrado interés porque sus procesos industriales tengan un mínimo impacto en el ambiente y ha hecho inversiones importantes en esa línea. También ha manifestado querer mejorar los entornos de trabajo de sus empleados, por lo que desea saber en qué momentos y en qué sedes se están presentando niveles altos de ruido que puedan afectar el bienestar de las personas. Al tener esta información se podrán tomar las medidas correctivas que se requieren.

La empresa manifestó los siguientes requerimientos:

- El sistema debería presentar la información de forma geo-referenciada y usar mapas de sus diferentes sedes para ubicar con mayor facilidad los eventos que se detecten.
- Los niveles de ruido se deberían poder clasificar en una escala con niveles: alto, medio y bajo.

El prototipo se desarrolló utilizando los siguientes componentes:

- Para el cliente móvil se empleó *NoiseTube* [36]. El código fuente de *NoiseTube* está disponible bajo licencia GNU LGPL v2.1
- Para el *middleware* se usó *middlewaresn* que soporta el protocolo MQTT. La especificación del protocolo MQTT se ha publicado abiertamente con una licencia libre de regalías durante muchos años. *Middlewaresn* es un producto de un Trabajo de Maestría de la Pontificia Universidad Javeriana (PUJ) [37].
- Para la comunicación del cliente móvil con Android se usó *mqtt-client-0.4.0* [38].
- Para procesar los datos se usó el algoritmo SMO (*sequential minimal optimization*) de *Weka* [39] (herramienta de código libre y abierto).
- Para los mapas el API de Google Maps [40].

A continuación se describe tanto el proceso de recolección de datos, como la descripción detallada de los componentes del modelo que fueron instanciados.

3.1 PROCESO DE RECOLECCIÓN DE DATOS

En algunos *Smartphones* que son propiedad de la empresa, se instaló un software para la medición de los decibeles de ruidos. El software fue desarrollado por la empresa *NoiseTube* y tiene versiones tanto para Android, como para iOS. El programa usa el micrófono del teléfono para capturar sonidos y ruidos. Para el prototipo se utilizaron dos *Smartphone* -un modelo Galaxy S2 y otro Galaxy S3- y un iPhone 4S. Estos dispositivos eran portados por dos personas que se encontraban en áreas seleccionadas. Se realizaron mediciones por dos semanas. Los datos fueron tomados por lapsos de tiempo de 5 minutos, cada 15 minutos por término de tres horas en cada sede. Los datos recolectados se enviaban al *middleware* por la red 3G o Wi-Fi.

3.2 COMPONENTES INSTANCIADOS DEL MODELO

3.2.1 Capa de Captura de Datos

En la Figura 4 se resaltan los componentes instanciados en la capa de captura (en gris), los cuales se explican posteriormente.

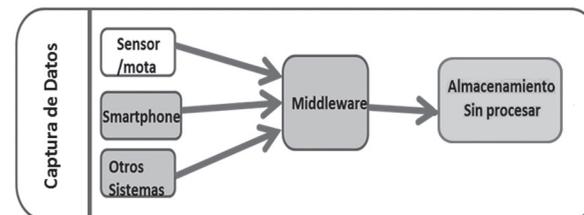
3.2.1.1 Smartphones y otros sistemas

En el prototipo, la captura de la información se puede realizar por dos canales diferentes: por medio de *Smartphones* y desde una plataforma externa (Otros Sistemas)

Para los *Smartphones* se usó un cliente de una plataforma para medir la contaminación auditiva alrededor de mundo (*NoiseTube*). El cliente de esta plataforma para Android, fue modificado para que enviara la información

recolectada al *middleware middlewaresn* mediante MQTT.

FIGURA 4: Capa de Captura de Datos con componentes instanciados.



El otro punto de entrada al sistema lo conforma la información proveniente de plataformas que almacenan información sobre contaminación. La plataforma *NoiseTube*, ofrece un *Web Service* que permite, mediante una url extraer información de contaminación auditiva en un rango de tiempo y una ubicación determinada. Una aplicación en cualquier computador o dispositivo puede extraer estos datos de la plataforma y enviarlos al *middlewaresn*. Para poder enviar la información recolectada por los teléfonos inteligentes a la plataforma *NoiseTube* se instaló la aplicación cliente en un iPhone 4S. Este software, toma las medidas y luego las envía a la plataforma *NoiseTube* mediante http. A través de un *Web Service* se descarga la información de la plataforma.

3.2.1.2 Middleware

Para almacenar o clasificar la información que viene de los dispositivos se usó el protocolo MQTT. El *middleware* utilizado fue *middlewaresn* que soporta los protocolos MQTT, UDP, SIP, HTTP. El *middleware* se modificó para lograr la captura de los datos y que éstos se almacenaran en una base de datos no relacional (*MongoDB*). A la estructura original del *middlewaresn* se le agregaron tres módulos, los cuales se pueden apreciar al estar encerrados en cuadros como se observa en la Figura 5. Los nuevos módulos tienen las siguientes funciones:

Cliente Móvil: Permite capturar información desde dispositivos *Smartphones* con Android usando el protocolo MQTT.

Plataformas de Mediciones de Ruido: Permite la entrada de datos de contaminación desde otras plataformas y sensores que se dedican a esta tarea.

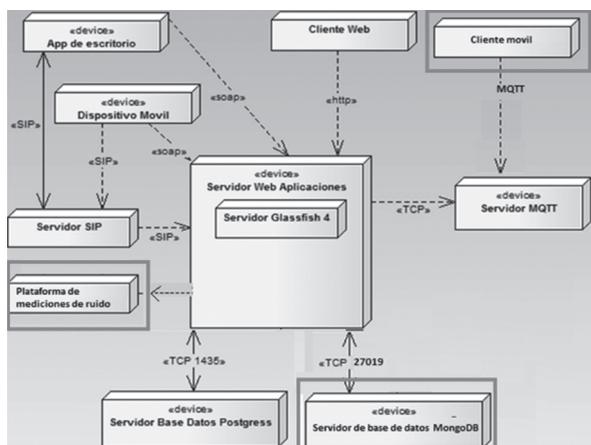
Servidor de Bases de Datos no estructurados: Permite el almacenamiento y la persistencia de la información capturada de una forma más ágil.

3.2.1.3 Almacenamiento sin procesar

El almacenamiento se realiza en la base de datos no relacional, de código abierto, *MongoDB*. Esta base de

datos fue seleccionada porque posee dos características muy importantes para el caso de estudio: primero, su gran velocidad de lectura y escritura de datos; en segundo lugar, su capacidad de hacer un manejo nativo de índices geo-referenciados. En la mayoría de los contextos donde se instancie el modelo de referencia será necesario contar con estas características debido a que la información que se captura son flujos continuos de mediciones, desde diferentes dispositivos al mismo tiempo, lo cual requiere de una importante velocidad de absorción de datos por parte del manejador de base de datos.

FIGURA 5: Arquitectura modificada del middleware *middlewaresn*.



Para almacenar los datos en *MongoDB* se crearon dos Tablas o *Collections*: a) Una tabla llamada *dependencias* que almacena las coordenadas de la dependencia de donde se capturan los datos. A esta colección se le creó un índice geo-referenciado con el campo ubicación, así cuando la información es almacenada se hace una búsqueda de las coordenadas y se extrae la ubicación a la que pertenece; este dato se agrega a la información enviada por el Smartphone. En la segunda tabla llamada *lecturaRuido* se almacenan las mediciones realizadas por los diferentes dispositivos. La información que se recolecta contiene la fecha de la medición, la medición de ruido, la latitud y longitud desde donde se hizo la medición y la ubicación, que es el nombre del lugar desde donde se tomó el registro de medidas.

3.2.2 Capa de Procesamiento

En esta capa se instanciaron 3 de los 4 módulos: pre-procesamiento, extracción y almacenamiento, tal como se puede apreciar en la Figura 6.

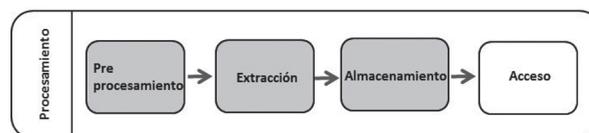
3.2.2.1 Preprocesamiento

Este componente realiza la conversión de las coordenadas (latitud y longitud) en la ubicación correspondiente a las dependencias de la empresa.

3.2.2.2 Extracción

Para atender a los requerimientos planteados por la empresa, es decir clasificar los niveles de ruido en alto, medio y bajo, se escogió el algoritmo SMO (*Sequential Minimal Optimization*), que es un algoritmo de aprendizaje supervisado para máquinas de soporte de vectores. Se estableció una clasificación de ruido con un nivel bajo con valores que van desde 1 a 55 decibeles, un nivel medio o normal con valores que van desde los 56 a los 80 decibeles y un nivel alto con valores comprendidos desde los 81 decibeles en adelante.

FIGURA 6: Capa de Procesamiento



Se utilizó la implementación del algoritmo en *Weka*. En el prototipo se integró una clase en *Java* con las librerías de *Weka* para configurar este clasificador. Utilizando el algoritmo se pudieron clasificar las medidas de decibeles del ruido en tres niveles, a los cuales se les asignó una paleta de colores. Esta asignación de colores se hizo con el objetivo visualizar la información a través de un mapa en la capa siguiente. Para el nivel bajo se asignó el color verde; para el nivel medio o normal, se asignó el color amarillo y para el nivel alto se asignó el color rojo. Dentro de cada color se manejan tonalidades de claro a oscuro a medida que el valor de los decibeles aumenta.

Al clasificador de ruido se le envía el valor de los decibeles de cada emisión y éste le retorna el color que corresponde a ese valor, el cual es almacenado posteriormente en la base de datos.

La relación entre los decibeles, los colores y la intensidad se puede observar en la Tabla 1.

3.2.2.3 Almacenamiento

El módulo de Almacenamiento es el encargado de la persistencia de los resultados del algoritmo (los

diferentes colores) en la base de datos de *MongoDB* para su posterior uso en la capa 3.

3.2.3 Capa de Visualización y Publicación

Para la instanciación del modelo, se decidió que en esta capa únicamente se implementaría el módulo de Visualización (Ver Figura 7), dado que con este módulo ya se podían atender adecuadamente los requerimientos de la empresa, y con el fin de poder contar rápidamente con el prototipo para poder cumplir con la fase de pruebas y evaluación.

La visualización de los niveles de ruido en las distintas sedes de la empresa se realizó con ayuda de mapas geográficos, utilizando el API de Google. Se generarán puntos de colores, de acuerdo a la información presente en la Tabla 1, desde donde se toman las mediciones. La información visualizada en el mapa se hace de acuerdo a los filtros de fecha inicial y final y al rango de horas en las que desea revisar las mediciones (esto hace parte de los parámetros del sistema). La información es tomada de la base de datos de tabla *lecturaRuido* donde se encuentran los códigos de colores que se generaron en la capa 2. Ejemplos de los resultados se pueden observar en las Figuras 8 y 9.

4. RESULTADOS

En el proceso de construcción de una instancia del modelo de referencia para la contaminación auditiva, se logró la rápida integración de componentes ya existentes y de código abierto como el cliente de recolección de ruido desarrollado por *NoiseTube*, el *middlewaresn*, el componente de *mqtt* (agregado al cliente móvil), el código de clasificación de *Weka* y el uso de librerías *Google Maps* para la visualización de la información en mapas según la ubicación de cada sede de la empresa. La definición previa del modelo de referencia permitió reducir el tiempo y la complejidad del ensamblaje de todos los componentes, lo cual puede ser un factor que incentive a las empresas al desarrollo y uso de este tipo de sistemas para mejorar las condiciones del medio ambiente y la salud en ambientes laborales.

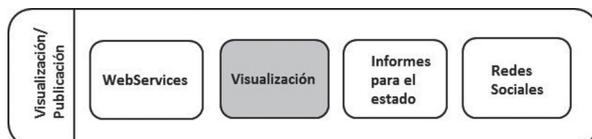
Es importante que las herramientas sean de código abierto o de elaboración propia ya que puede ser necesario introducir modificaciones para el ensamblaje de los componentes. Para construir el prototipo para la contaminación por ruido fue necesario introducir algunas modificaciones: por ejemplo, el cliente de *NoiseTube* que funciona en los Smartphones fue modificado para que ya no enviara los datos al servidor *noisetube.net* sino que usara el protocolo de comunicaciones MQTT, para enviar los datos al servidor *middlewaresn*. Este

middleware también se modificó para lograr que el almacenamiento de los datos se realizara en una base de datos *MongoDB*. Las Bases de Datos *NoSql* como *MongoDB* son adecuadas para el manejo eficiente de este tipo de datos recolectados con sensores por su gran velocidad de lectura y escritura, así como por el manejo nativo de índices georreferenciados.

TABLA 1. Relación entre decibles y colores

Intensidad	Color	Código de Color	Decib.
Baja	Verde claro	90FE00	0-42
Baja	Verde normal	89FD00	43-48
Baja	Verde encendido	C4FE04	49-55
Media	Amarillo claro	Fbfd01	56-62
Media	Amarillo normal	FDF900	63-67
Media	Amarillo encendido	FFE300	68-80
Alta	Rojo claro	FD7203	81-95
Alta	Rojo normal	FD2EE00	96-119
Alta	Rojo encendido	FD0100	120-N

FIGURA 7: Capa de Visualización / Publicación



De los resultados obtenidos a partir del prototipo implementado, podemos concluir que: con la

información recolectada, procesada y visualizada en los mapas sobre los niveles de ruido de las cuatro sedes de la empresa, monitoreadas por dos semanas con un total de 243.000 muestras, se encontró que una de las sedes de la empresa estaba presentando niveles de ruido altos con respecto a las otras sedes. Al realizar una revisión en el sitio se observó que una de las causas era la falta de mantenimiento en unas máquinas del taller de pintura y el funcionamiento no adecuado de los gatos hidráulicos. También se pudo establecer que los tapa oídos usados en las tareas de latonería no tenían el suficiente aislamiento para el ruido generado en esta actividad y se procedió a reemplazarlos por unos más robustos en todas sus sedes.

FIGURA 8: Medición de ruido sede calle 80 de la Empresa.

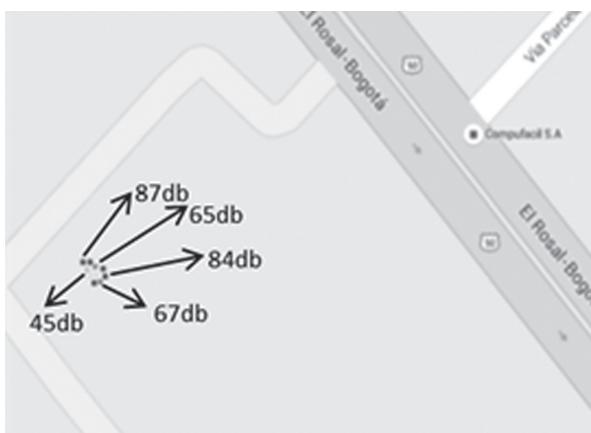


FIGURA 9: Medición de ruido en la sede de Tenjo de la Empresa, en la que se resaltan las mediciones en forma de mapas de calor



5. TRABAJOS RELACIONADOS

En el artículo se ha propuesto un modelo de referencia que servirá de guía para la implementación de sistemas para la recolección y almacenamiento de datos de contaminación industrial provenientes de diferentes

tipos de sensores. El modelo sugiere diversas formas de procesar la información para que, tanto usuarios, como aplicaciones, puedan extraer conocimiento útil y producir alertas que permitan a las empresas tomar medidas tanto preventivas como correctivas.

Se desarrolló una instancia del modelo para la medición de la contaminación auditiva en una empresa del sector automotriz.

En esta sección se mencionan un conjunto de trabajos relacionados. Los diversos trabajos tienen en común las características presentadas en la primera columna de la Tabla 2. Para hacer una descripción de estas propuestas se comenzará con la descripción de aquellos artículos que tienen menos semejanzas, hasta llegar a aquellos que presentan mayor relación con el trabajo aquí propuesto.

Hasenfratz y otros [43] presentan un sistema de detección móvil de bajo consumo y bajo costo para el monitoreo participativo de la calidad del aire. Utilizan hardware "of-the-shelf" para monitorizar la concentración de ozono. Su sistema, *GasMobile*, proporciona una alta precisión de datos aprovechando lecturas de sensores que se encuentran cerca de estaciones de medición estáticas. Este trabajo aunque no propone un modelo de referencia, se selecciona porque el sistema recolecta datos de calidad del aire usando sensores en teléfonos inteligentes.

En el trabajo de Godoy y otros [41] se recogen datos ambientales, tales como temperatura ambiental, presión relativa y humedad atmosférica, pero desde sensores tipo motas. El objetivo de este artículo es la construcción de un *middleware* que facilite la recolección, persistencia y publicación de datos en la Internet. El *middleware* se propone como una solución a la heterogeneidad existente en sensores y plataformas de software. En este sentido pudiera ser un componente del modelo de referencia propuesto en este artículo, en su capa de Captura de Datos. Los aspectos comunes se pueden observar en la tabla: medición de variables ambientales desde motas y consideración de la heterogeneidad.

Otro trabajo que también presenta estas tres características en común de recolectar datos ambientales, usar motas como nodos sensores y presentar soluciones a la heterogeneidad en diversas plataformas tecnológicas es el descrito por Navarro y otros [42]. En él los autores presentan el sistema INDAMS para la integración y gestión de datos recolectados por redes de sensores heterogéneas, e implementan un prototipo para el monitoreo de variables ambientales.

En el trabajo de Lane y otros autores [23] también se recolectan datos de contaminación auditiva desde

teléfonos inteligentes. Aunque no usan un modelo de referencia, proponen una arquitectura de tres capas similar a la propuesta en este artículo. Los niveles definidos son: la capa de sensores, la capa de aprendizaje (donde se aplican técnicas de minería de datos) y la capa de visualización (donde se presenta la información obtenida por medio de gráficas y tablas que pueden ser compartidas en redes sociales).

Los siguientes dos trabajos proponen también una arquitectura de tres niveles para la recolección, procesamiento y visualización de la información. En el trabajo de Chun-Hsiang Lee y otros [5], se plantea el problema de gestionar una gran cantidad de datos provenientes de sensores en dispositivos móviles de forma colaborativa. Aunque no es particular para datos de contaminación, si se abordan los problemas de la heterogeneidad y disparidad de la información recolectada, la existencia de propiedades de multi-resolución y multi-escala, diferentes niveles de incertidumbre y confiabilidad de la información, como por ejemplo, información generada por sensores defectuosos, mal calibrados o información manipulada maliciosamente. Otro aspecto común es que la arquitectura propuesta persigue obtener un valor adicional de los datos obtenidos usando técnicas de minería de datos o texto.

Schweizer y otros [44] presentan una iniciativa para la generación de mapas de calor que muestran la contaminación causada por el ruido, la humedad, el calor y el aire en la ciudad de Frankfurt. Esta iniciativa usa la aplicación *NoiseMap* en los teléfonos inteligentes para permite una mayor recolección de muestras de ruido. Luego, las muestras son transmitidas al servicio web *Da sense* [32]. Al igual que el modelo de referencia propuesto, la arquitectura de este sistema posee tres niveles: recolección a través de sensores, procesamiento de la información y visualización. En el artículo se presentan herramientas muy puntuales para el procesamiento y visualización de datos (ejemplo *clustering* y visualización a través de mapas de calor). El modelo de referencia propuesto presenta una mayor variedad de técnicas de minería de datos y otras posibilidades para la capa de presentación como la conexión a redes sociales y reportes para el estado.

6. CONCLUSIONES Y TRABAJOS FUTUROS

La contaminación industrial es un aspecto que afecta en forma importante la sostenibilidad del planeta. Para detectar la contaminación en forma oportuna y tomar las medidas correspondientes se necesitan datos. Hoy en día, se cuenta afortunadamente con la ayuda de las redes de sensores que permiten recolectar grandes cantidades de datos en el ambiente. No obstante, no es fácil procesar esta enorme cantidad de datos, más aún si se toma en cuenta la diversidad de tecnologías existentes.

Todo lo anterior motivó a diseñar el Modelo de Referencia presentado en este artículo. El modelo debe servir de guía para la construcción de sistemas que utilicen los datos recolectados por redes de sensores y provean información de utilidad para el diseño de estrategias de prevención de la contaminación ambiental. El modelo sirve de referencia para realizar el almacenamiento, procesamiento y visualización/publicación de la información de contaminación recolectada por los sensores utilizando tecnologías existentes.

De la investigación realizada se pueden derivar una serie de trabajos futuros entre los que se encuentran: a) El desarrollo de módulos que permitan la entrada de datos, no solo desde Smartphones, sino también desde motas o desde sensores más tradicionales. Para el uso de motas fijas en el área donde se desea medir el impacto del ruido, es necesario tener en cuenta la atenuación que sufre la onda de sonido, desde el lugar donde se genera el ruido hasta el sensor que lo registra, para poder aplicar el factor de corrección adecuado a la medición. b) Hacer la verificación del rendimiento y la disponibilidad del sistema cuando miles o millones de sensores estén enviando información a un servidor. c) A nivel de la capa 2 (Capa de Procesamiento) sería interesante probar otras técnicas de minería de datos. d) A nivel de capa 3 es importante instanciar el resto de módulos propuestos para la presentación de la información, como los *Web Services* y las redes sociales. e) Hacer prototipos del modelo para resolver otro tipo de problemas de contaminación (en el aire, agua etc.); esto permitirá evaluar nuevos desafíos que se pueden presentar en esos escenarios.

TABLA 2: Comparación de Trabajos Relacionados

Características/trabajos	Godoy y otros [41]	Navarro y otros [42]	Lane y otros [23]	Hasenfratz y otros [43]	Lee y otros [5]	Schweizer y otros [44]	Modelo de Referencia Propuesto en el artículo
Propone un modelo y/o arquitectura para la recolección, procesamiento y visualización y/o distribución de los datos a través de diferentes medios.			X		X	X	X
Recolecta Datos relacionados con variables del ambiente o contaminación ambiental	X	X	X	X		X	X
Usa motas como sensores	X	X				X	X
Usa sensores de teléfonos inteligentes			X	X	X	X	X
Aplica técnicas de minería de datos para obtener un valor adicional de los datos					X	X	X
Presenta soluciones a la heterogeneidad en las diversas tecnologías	X	X			X		X

7. REFERENCIAS BIBLIOGRÁFICAS

- [1] Lasso, J. C. (2014). La contaminación del aire, el mayor problema ambiental de salud pública en el mundo. Recuperado de: <http://www.radionacional.co/noticia/la-contaminacion-del-aire-el-mayor-problema-ambiental-de-salud-pblica-en-el-mundo>.
- [2] García, S. (2015). Contaminación Industrial. Recuperado de: <http://www.elfinanciero.com.mx/opinion/contaminacion-industrial.html>
- [3] Secretaría Distrital del Ambiente (s.f.). Red de Monitoreo de Calidad del Aire de Bogotá – RMCAB. Recuperado de: <http://ambientebogota.gov.co/red-de-calidad-del-aire>
- [4] Burke, J. A., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S., Srivastava, M. B. (2006). Participatory sensing. *World Sensor Web Workshop*.
- [5] Lee, C. H., Birch, D., Wu, C., Silva, D., Tsinalis, O., Li, Y., Yan, S., Ghanem, M., Guo, Y. (2013). Building a generic platform for big sensor data application, in Big Data. In *Proc. of the IEEE International Conference on Big Data*, 94–102.
- [6] Levis, P., Culler, D. (2002). Maté: A tiny virtual machine for sensor networks. *ACM Sigplan Notices*, 37(10), 85-95.
- [7] Barr, R., Bicket, J. C., Dantas, D. S., Du, B., Kim, T. W., Zhou, B., Sifer, E. G. (2002). On the need for system-level support for ad hoc and sensor networks. *ACM SIGOPS Operating Systems Review*, 36(2), 1-5.
- [8] Hall, S. P., Anderson, E. (2009). Operating systems for mobile computing. *Journal of Computing Sciences in Colleges*, 25(2), 64-71.
- [9] Srisathapornphat, C., Jaikaeo, C., Shen, CC. (2000). Sensor information networking architecture. *Proceedings of the International Workshops on Parallel Processing*.
- [10] Yu, X., Niyogi, K., Mehrotra, S., Venkatasubramanian, N. (2003). Adaptive middleware for distributed sensor environments. *IEEE distributed systems online* 5.
- [11] Eclipse Mosquitto (2014). An Open Source MQTT v3.1/v3.1.1 Broker [iot.eclipse.org](http://mosquitto.org). Recuperado de: <http://mosquitto.org/>

- [12] Saffirio, M. (2010). Modelos de Referencia. Recuperado de: <https://msaffirio.com/2010/11/01/modelos-de-referencia-reference-models/>
- [13] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey on sensor networks. *Communications magazine*, 40(8), 102-114.
- [14] Kahn, JM., Katz, RH., Piste, KSJ. (1999). Next century challenges: mobile networking for smart dust. *Proc. of the 5th annual ACM/IEEE international conference on Mobile computing and networking (Mobicom 99)*, Washington, USA, 271-278.
- [15] Culler, D., Estrin, D., Srivastava, M. (2004). Guest editors' introduction: Overview of sensor networks. *Computer*, (8), 41-49.
- [16] Lazarescu, M. T. (2013). Design of a WSN platform for long-term environmental monitoring for IoT applications. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 3(1), 45-54.
- [17] Ferdoush, S., Li, X. (2014). Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications. *Procedia Computer Science*, vol. 34, 103-110.
- [18] Arias, D. MR., Muñoz Rodríguez, F., Cely Callejas, J. D. (2012). "Design of a sensor network implemented in a monitored via web prototype." *Proc. of the IEEE Colombian Communications Conference (COLCOM 2012)*.
- [19] Pico, P. (2012). *Monitoreo del medio ambiente basado en sensores*.
- [20] Liu, T., Martonosi, M. (2003). Impala: A middleware system for managing autonomic, parallel sensor systems. *ACM SIGPLAN Notices*, 38(10), 107-118.
- [21] Soini, M. N., Van Greunen, J., Rabaey, J. M., Sydanheimo, L. T. (2007). Beyond sensor networks: Zuma middleware. *Proc. of the Wireless Communications and Networking Conference (WCNC 2007)*, 4318-4323.
- [22] Gerson, M. (Febrero 22, 2015). Smartphone Sales Grew Almost 10% Last Quarter. Benziga. Recuperado de: <http://www.benziga.com/analyst-ratings/analyst-color/16/02/6493276/smartphone-sales-grew-almost-10-last-quarter>
- [23] Lane, N. D., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., & Campbell, A. T. (2010). A survey of mobile phone sensing. *Communications Magazine*, 48(9), 140-150.
- [24] De Sanctis, M., Stallo, C., Parracino, S., Ruggieri, M., Prasa, R. (2012). Interoperability solutions between smartphones and wireless sensor networks. In : *Proc. in the IEEE first AESS European conference on satellite telecommunications (ESTEL 2012)*, 1-6.
- [25] Zhang, J., Chen, C., Ma, J., He, N., Ren, Y. (2011). USink: Smartphone-based mobile sink for wireless sensor networks. *Proc in Consumer Communications and Networking Conference (CCNC2011)*, 90-95.
- [26] Zöllner, S., Reinhardt, A., Guckes, H., Schuller, D., Steinmetz, R. (2011). On the integration of wireless sensor networks and smartphones in the logistics domain. *Proc. of the 10th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze*, 49-52.
- [27] González, R., Huerta, M. (2013). Análisis del uso de teléfonos inteligentes como nodos de una red inalámbrica de sensores. *Infociencia* 7, 35-41.
- [28] Products-Libellium (Octubre 1, 2014). Libellium. Recuperado de: <http://www.libellium.com/es/products/>.
- [29] Sensorcon (Octubre 1, 2014). Sensorcon Catalog Our Single Gas Meters and Environmental Sensors. Recuperado de: <http://sensorcon.com/products/>
- [30] Jain, N., Srivastava, V. (2013). Data Mining techniques: A survey paper. *IJRET: International Journal of Research in Engineering and Technology*, 2(11), 116-119.
- [31] Maimon, O., Rokach, L. (2005). *Data mining and knowledge discovery handbook 2*. Springer, New York.
- [32] Da Sense (Septiembre 22, 2016) www.da-sense.de. Recuperado de: <https://www.tk.informatik.tu-darmstadt.de/de/research/smart-urban-networks/da-sense/>
- [33] MQTT (diciembre 1, 2014) mqtt.org. Recuperado de: <http://mqtt.org/>
- [34] Shelby, Z., Hartke, K., Bormann, C. (Junio 2014) The constrained application protocol (CoAP) No. RFC 7252. Recuperado de: <https://www.rfc-editor.org/rfc/pdf/rfc7252.txt.pdf>

- [35] Bormann, C., Z., S. (Agosto 2016) Block-Wise Transfers in the Constrained Application Protocol (CoAP). RFC 7959, DOI 10.17487/RFC7959. Recuperado de: <http://www.rfc-editor.org/info/rfc7959>
- [36] Maisonneuve, N., Stevens, M., Niessen, M. E., Steels, L. (2009) NoiseTube: Measuring and mapping noise pollution with mobile phones. *Proc. of the Information Technologies in Environmental Engineering*, 215-228.
- [37] Ruíz García, E. E. (2012) *Integración de Redes Inalámbricas de Sensores IEEE 802.15.4 en Aplicaciones USN*. Memoria del Trabajo de Grado de Maestría en Computación y Sistemas, Pontificia Universidad Javeriana, Bogotá, Colombia.
- [38] MQTT (noviembre 15, 2014) mqtt-client : 0.4.0. Recuperado de: <https://www.versioneye.com/java/org.eclipse.paho:mqtt-client/0.4.0>
- [39] Balam, L. A. Clasificación de temperatura con weka en Java y NetBeans.
- [40] Google Circles. Google Maps. Recuperado de: <https://developers.google.com/maps/documentation/javascript/examples/circle-simple>
- [41] Godoy, D. A., Sosa, E. O., Neis, R., Díaz Redondo, R. (2014). Ambientes Inteligentes: middleware de soporte para la captura, almacenamiento y publicación de datos de una red de sensores inalámbricos. *Proc. del XX Congreso Argentino de Ciencias de la Computación*, Buenos Aires.
- [42] Navarro, M., Bhatnagar, D., Liang, Y. (2011) An integrated network and data management system for heterogeneous WSNs. *IEEE Eight International Conference on Mobile Ad-Hoc and Sensor Systems*, 819-824.
- [43] Hasenfratz, D., Saukh, O., Sturzenegger, S., Thiele, L. (2012) Participatory air pollution monitoring using smartphones. *Proc. of the 2nd International Workshop on Mobile Sensing*, Beijing, China.
- [44] Schweizer, I., Bärthel, R., Schulz, A., Probst, F., Mühläuser, M. (2011) NoiseMap-real-time participatory noise maps. *Proc. on the 2nd Int'l Workshop on Sensing Applications on Mobile Phones (PhoneSense'11)*.

