

DOI: 10.30972/eitt.704777

Fusión y análisis de datos para la toma de decisiones en un escenario vehicular complejo

MSc. Raimundo Vázquez ; Ing. Carlos Torres, MSc. Mariguetti(*)

Resumen

Se desarrolla un procedimiento que permite a una App predecir el estado del tránsito complejo articulado con una API de Google Maps y librerías de Máquina de Aprendizaje. Para tales fines se define una zona de tráfico vehicular donde se estiman los tiempos invertidos por un conductor al recorrer dichas calles. Cada uno de los ejemplos se asocia a un estado de tráfico en cuatro categorías: bajo, medio, moderado y alto. De esta manera se crea un conjunto de entrenamiento que es la base de información de un algoritmo de inteligencia artificial. Se elige una red neuronal del tipo *MultiplayerPerceptron* de una sola capa y 28 neuronas. El clasificador dio una *TP Rate* elevada y *FP Rate* muy bajo, indicando instancias de clasificación muy satisfactoria para las cuatro categorías. La Precisión del modelo muestra una buena sensibilidad y la Memoria una adecuada especificidad. El área ROC supera en todos los casos el valor 0.9, reflejando un test muy bueno. La matriz de confusión está formada por una sola diagonal acusando una buena sensibilidad y precisión.

Palabras clave: estado del tránsito, densidad de tráfico, estimación del tráfico.

(*) GUDA-Grupo Universitario de Automatización | Universidad Tecnológica Nacional - Facultad Regional Resistencia French 414. 3500 Resistencia-Chaco. República Argentina | Tel: +54 362 4432928. / Fax: +54 362 4432683/ e-mail: ray_vazquez_2005@hotmail.com

Introducción

En los últimos años, la industria automotriz está equipando vehículos con sistemas sofisticados y, a menudo, costosos en la asistencia a la conducción. Sin embargo, esta tecnología vehicular está más enfocada en facilitar la conducción y no en monitorear al conductor (Yi Zheng, 2016). Para tales fines, se da seguimiento a la forma de manejo del conductor dentro del automóvil con la intención de prevenir siniestros utilizando un modelo de trabajo como lo describe (Sendra, 2016). También se destaca el trabajo de (Stephan, 2012) donde muestra una arquitectura novedosa en la visión automotriz para organizar capas de abstracción en relación a los sensores, datos, niveles semánticos, razonamiento y asignación de recursos. A partir de estos valores, se analiza las señales y se alerta al conductor para evitar posibles siniestros.

El propósito de este trabajo es implementar un procedimiento de predicción en una zona de alta densidad de tráfico el estado de congestión vehicular utilizando tecnología móvil. También se brinda una alternativa en aplicaciones en el campo de la comunicación de vehículo a vehículo (V2V) (Hideyuki, 2005), Vehículo-a-infraestructura (V2I) (Sung-Yeop, 2010) y tecnologías vehículo-peatón (V2P) (Liu, 2015). Dichos sistemas fueron diseñados para transmitir información básica de seguridad entre los vehículos y facilitar las advertencias a los conductores acerca de accidentes inminentes. De esta manera se pretende contribuir en soluciones tecnológicas para abordar los problemas de seguridad vial. Usando este informe se podría facilitar la toma de datos relevantes para procesar y mejorar la toma de decisión del conductor.

Conceptos Fundamentales

Entorno de Desarrollo Integrado Android Studio

Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Está disponible para las plataformas Microsoft Windows, macOS y GNU/Linux. Dicha herramienta permite implementar aplicaciones o App en dispositivos móviles (celulares o Tablet). Se recurren a librerías de uso libres y código fuente abierta, que facilitan el manejo de: comunicación bluetooth, ubicación mediante tecnologías GPS, etc.

Analizador de Tráfico del Google Maps

Una de las prestaciones más interesantes de Google Maps es su función como navegador vehicular. El cálculo de rutas suele funcionar realmente bien, pero además hay un dato especialmente importante para los usuarios: el tiempo estimado de llegada a destino. Esos datos disponibles, en una determinada zona, se realizan tomando en cuenta elementos básicos como la velocidad máxima permitida y la velocidad recomendada. Además se intentan captar los datos históricos de las velocidades medias en esas carreteras y calles por la que el conductor pasa, e incluso los tiempos que han invertido otros usuarios en rutas previas. Otros datos que entran en juego son, por supuesto, los de tráfico, con la información en tiempo real que afecta especialmente al tiempo estimado de llegada. Este último factor es el que acaba teniendo más impacto sobre la estimación final. Actualmente existen programas informáticos encargados de la toma de datos representativos del tránsito en un escenario vehicular complejo. Dichos modelos están basados en sistemas de representación del conocimiento y en análisis de los datos (Vadim, 2013). El procedimiento para implementar un clasificador que permita estimar el estado de congestión vehicular en utilizando tecnología móvil se obtuvo de (Android, 2020).

Herramienta WEKA

Es una extensa colección de algoritmos de aprendizaje de máquinas o ML desarrollado por la universidad de Waikato, implementado en Java, ver figura 1.



Figura 1: Programa Weka.

Es útil para clasificar datos mediante los interfaces dentro de cualquier aplicación, por ejemplo celulares o *Tablet*. Además Weka contiene las herramientas necesarias para realizar transformaciones sobre los datos, tareas de predicción, regresión, clustering, asociación y visualización (Bostjam, 2020). Weka está programado en Java y es independiente de la arquitectura, ya que funciona en cualquier plataforma sobre la que haya una máquina virtual Java y también está disponible en *Android Studio*.

Hipótesis de Trabajo

Es posible clasificar el estado del tránsito vehicular complejo utilizando una herramienta ML, una API de Google Maps integrado a una aplicación. La fusión de datos se realiza mediante la herramienta denominada Weka. Esta hipótesis será comprobada experimentalmente mediante una App.



Figura 2: Tráfico vehicular en tiempo real utilizando la API de Google Maps

La App obtiene la información del tráfico de todas las calles encerradas en el rectángulo en una hora determinada del día. Es decir, se extraen conjuntos de datos donde se sabe previamente la congestión del tráfico. Dicha información se

clasifican en: baja, mediana, elevada y alta. La figura 3 muestra un mallado de la zona vehicular.

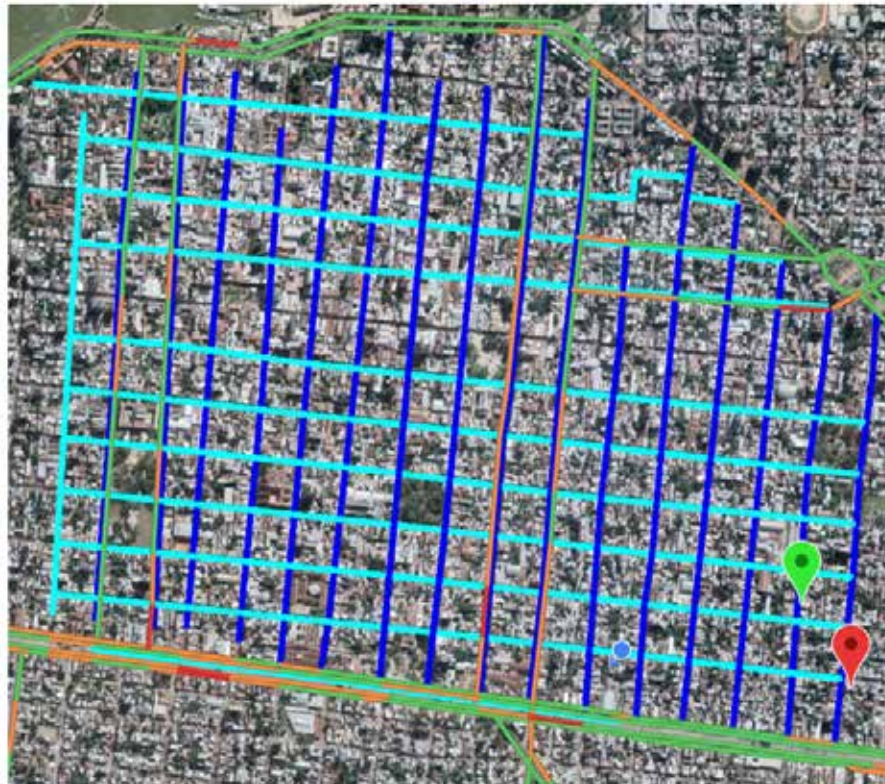


Figura 3: Mallado de la zona vehicular

Las primeras 11 calles se visualizan en forma horizontal (color celeste) y las 17 calles restantes están dispuestas en forma vertical (color azul). La figura 4 muestra el tiempo estimado en cada calle dentro de la zona vehicular en diferentes horas del día.

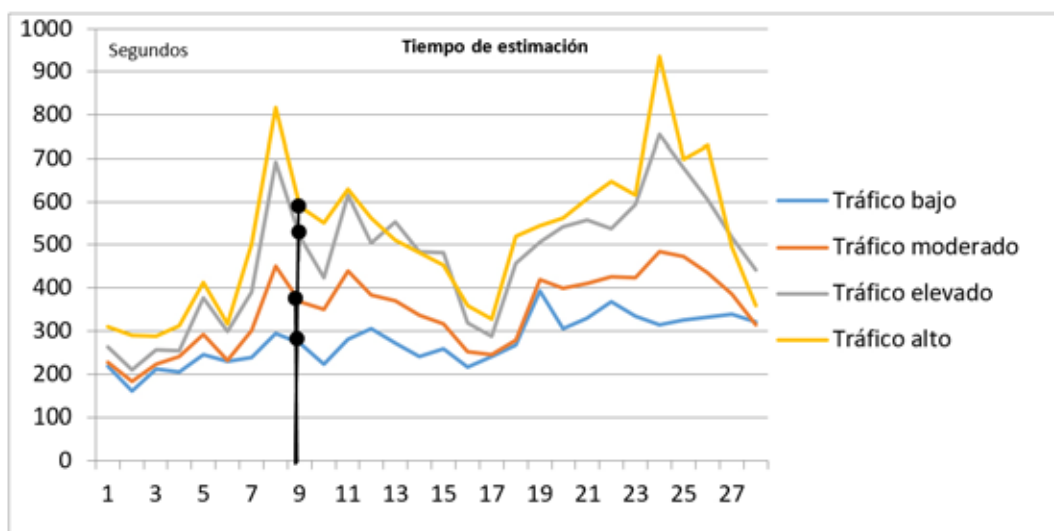


Figura 4: Representación de los tiempos de estimación en la zona vehicular

Se observa, por ejemplo, el tiempo necesario en recorrer la calle N°9 utilizando un automóvil durante cuatro momentos del día. La gráfica muestra una relación entre el tiempo de estimación y el grado de congestión.

Procedimiento

Se desarrollada una App que obtiene información de los tiempos estimados para recorrer cada calle. Se capturan los datos en situaciones de densidad de tráfico conocido. Se utiliza una red neuronal del tipo *MultiplayerPerceptron* como se visualiza en la figura 5. Se extraen los patrones de los ejemplos. Luego se clasifican los estados del tránsito en cuatro categorías: bajo, moderado, elevado y alto.

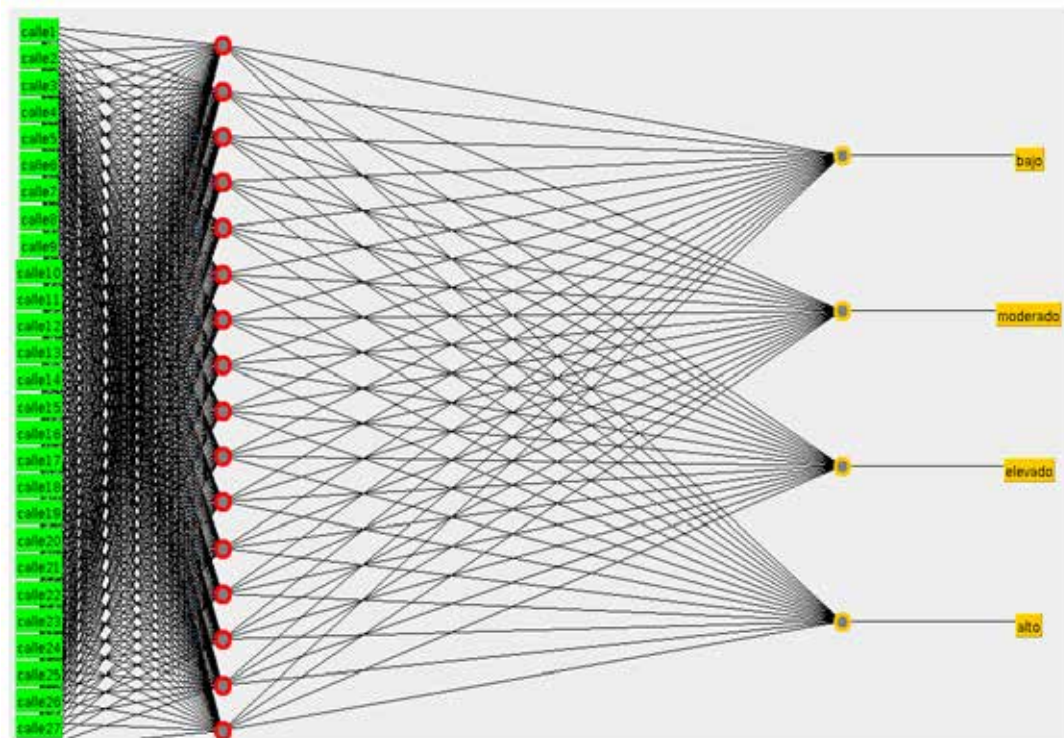


Figura 5. Red neuronal MultiplayerPerceptron.

Finalmente se obtiene un modelo predictivo de densidad del tráfico en un escenario vehicular. Los resultados de la evaluación se visualizan en la figura 6.

La columna TP Rate tiene una tasa elevada de verdaderos positivos, indicando instancias de clasificación muy satisfactoria para cada clase. La columna FP Rate indica falsos positivos iguales a 0. La Precisión muestra una buena sensibilidad y

```

=== Summary ===

Correctly Classified Instances      23          100   %
Incorrectly Classified Instances    0           0   %
Kappa statistic                    1
Mean absolute error                 0.0134
Root mean squared error             0.0165
Relative absolute error             3.5836 %
Root relative squared error         3.8043 %
Total Number of Instances          23

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
      1      0      1      1      1      1      bajo
      1      0      1      1      1      1      moderado
      1      0      1      1      1      1      elevado
      1      0      1      1      1      1      alto
Weighted Avg.  1      0      1      1      1      1

=== Confusion Matrix ===

 a b c d  <-- classified as
6 0 0 0 | a = bajo
0 6 0 0 | b = moderado
0 0 6 0 | c = elevado
0 0 0 5 | d = alto

```

Figura 6. Resumen del modelo predictivo.

la Memoria una adecuada especificidad. El área ROC supera en todos los casos el valor de 0.9 reflejando un test muy bueno. La matriz de confusión está formada por una sola diagonal acusando una buena sensibilidad y precisión.

Resultados

A la izquierda de la figura 7 muestra la imagen de la App encargada de la toma de datos de una zona vehicular. Posteriormente se genera una ventana emergente visualizada en la imagen de la derecha. El botón denominado Neurona genera un evento que habilita la función encargada de predecir el estado del tráfico según los datos obtenidos. En este caso el clasificador predice un tráfico moderado.

Consideraciones Finales

La App desarrollada se conecta a la red celular y no utiliza los protocolos de comunicación ni ancho de banda de los otros sistemas relacionados al tráfico vehicular tales como V2V, V2I, V2P y V2X. Existen sistemas de monitoreo de estados

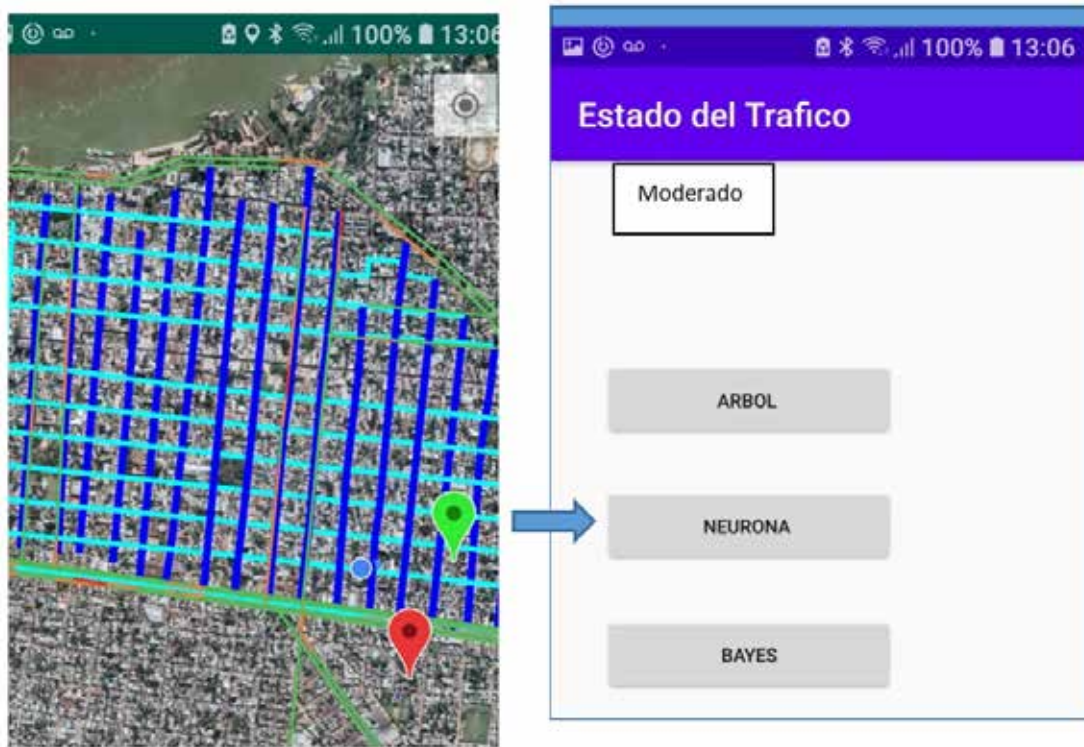


Figura 7. Clasificación del estado vehicular utilizando una App

del tráfico complejo utilizando procesamiento imagen articulado con una sofisticada conexión de sensores. En este sentido, la aplicación desarrollada, no necesita potencia de procesamiento de datos ni acceso a sensores del exterior y se pueden generalizar su uso en otros sectores de la ciudad.

Los datos enviados por el Google Maps pueden brindar además informaciones relacionadas a: peatones, ciclistas, tiempo de espera de autobuses etc.

El lenguaje utilizado para el manejo de las librerías ML y las API Google Maps fue Java, permitiendo generalizar su implementación en notebook o PC de escritorio.

Conclusiones

Se desarrolló un procedimiento que permite a una App estimar el estado del tránsito complejo articulado con una API de Google Maps y librerías de ML.

Se define una zona de tráfico vehicular donde se estiman los tiempos invertidos por un conductor al recorrer dichas calles. Cada uno de los ejemplos se asocia con un estado de tráfico conocido con la clase: bajo, medio, moderado y alto. Los ejemplos obtenidos son la base de información por el cual un algoritmo de inteligencia artificial logra obtener patrones del estado de tránsito complejo.

Se eligió un predictor denominado MultiplayerPerceptron de una sola capa y 28 neuronas. El resultado de la clasificación generó una TP Rate elevada y FP Rate muy bajo, indicando instancias clasificación muy satisfactoria para las cuatro clases.

La Precisión muestra una buena sensibilidad y la Memoria una adecuada especificidad.

El área ROC supera en todos los casos el de valor 0.9, reflejando un test muy bueno.

La matriz de confusión está formada por una sola diagonal acusando una buena sensibilidad y precisión.

REFERENCIAS

- Android*, <https://developers.google.com/android/guides/setup>
- Bestjam Kaluza. *Instal WEKA HOW-TO. Implement cutting-edge data minig aspects in Weka to your application. Amazon 2020. EEUU.*
- Hideyuki Shinonaga and Tomohiro Nishida. *Adaptive Media Switching for Future Vehicle-to-Vehicle Communication 2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications. Takayuki Warabibo, Kenji Saito, Keizo Sujiyama,*
- Liu Zhenyu, Pu Lin, Zhu Konglin, Zhang Lin. *Design and evaluation of V2X communication system for vehicle and pedestrian safety.. The Journal of China Universities of Posts and Telecommunications 2015.*
- Sendra Sandra, Jimenez Jose, Lloret Jaime. *Low-Cost Vehicle Driver Assistance*

System for Fatigue and Distraction Detection. Signal Theory, Telematics and Communications Department (TSTC), Universidad de Granada 2017.

Stephan Matzka, Member, IEEE, Andrew M. Wallace, and Yvan R. Petillot, Member, IEEE. *Efficient Resource Allocation for Attentive Automotive Vision Systems. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 13, NO. 2, JUNE 2012*

Sung-Yeop Pyun, Helena Widiart, Yong-Jin Kwon and Dong-Ho Cho. *TDMA-based Channel Access Scheme for V2I Communication System using Smart Antenna. 2010 IEEE Vehicular Networking Conference..*

Vadim Indelman, Andrew Melim, and Frank Dellaert. *Incremental Light Bundle Adjustment for Robotics Navigation. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) November 3-7, 2013. Tokyo, Japan.*

Yi Zheng, Lisheng Jin*, Linlin Gao, Keyong Li, Yan Wang and Faji Wang. *“Development of a Distributed Cooperative Vehicles Control Algorithm Based on V2V Communicatio” Procedia Engineering 137 2016.*

Abstract

A procedure is developed that allows an App to predict the state of complex traffic articulated with a Google Maps API and Learning Machine libraries. For such purposes, a vehicular traffic zone is defined where the times invested by a driver when traveling these streets are estimated. Each of the examples is associated with a traffic condition in four categories: low, medium, moderate and high. In this way, a training set is created that is the information base of an artificial intelligence algorithm. A neural network of the MultiplayerPerceptron type with a single layer and 28 neurons is chosen. The classifier gave a high TP Rate and a very low FP Rate, indicating very satisfactory classification instances for the four categories. The Precision of the model shows a good sensitivity and the Memory an adequate specificity. The ROC area exceeds 0.9 in all cases, reflecting a very good test. The confusion matrix is made up of a single diagonal, showing good sensitivity and precision.

Keywords: *traffic status, traffic density, traffic estimate.*