

¿Cuánto Falta? Modelos Predictivos del Tiempo de Llegada de Autobuses

Nicolas Miccio Palermo¹ (0009-0002-7416-5599 ) , Marcelo Armentano² (0000-0002-9172-9061 ) , and Antonela Tommasel² (0000-0001-6091-8305 )

¹ Facultad de Ciencias Exactas, UNICEN

nicolasmicciopalermo@gmail.com

² ISISTAN, CONICET-UNICEN

{marcelo.armentano, antonela.tommasel}@isistan.unicen.edu.ar

Resumen Predecir con precisión los tiempos de llegada de los autobuses es fundamental para mejorar la movilidad urbana y optimizar los servicios de transporte público. Las demoras y la incertidumbre en los horarios generan frustración en los pasajeros y dificultan la planificación eficiente de los viajes. En este contexto, contar con estimaciones confiables y en tiempo real sobre los tiempos de llegada puede ayudar a los usuarios a reducir la espera y tomar decisiones informadas. Este trabajo analiza distintos enfoques predictivos, incluyendo Regresión Lineal, ARIMA, redes LSTM (*Long Short-Term Memory*) y GRU (*Gated Recurrent Units*), para estimar los tiempos de llegada a partir de datos reales de GPS de autobuses en la ciudad de Tandil (Buenos Aires, Argentina). Los resultados experimentales mostraron que los modelos de aprendizaje profundo, en particular LSTM, superan significativamente a los enfoques tradicionales, lo que resalta su potencial para optimizar los sistemas de transporte público. Además del desarrollo y evaluación de los modelos predictivos, se diseñó e implementó una aplicación móvil que integra estas predicciones y ofrece a los usuarios información en tiempo real sobre los horarios estimados de llegada y posibles demoras.

Keywords: Transporte público, Redes neuronales, Movilidad urbana

How Much Longer? Estimating Bus Arrival Times with Predictive Models

Abstract. Predicting bus arrival times accurately is essential for improving urban mobility and enhancing public transportation services. Delays and uncertainty in bus schedules can lead to passenger frustration and inefficient travel planning. In this context, providing real-time, reliable arrival time estimates can help commuters reduce waiting times and make informed decisions. This work explores different predictive approaches, including Linear Regression, ARIMA, Long Short-Term Memory (LSTM), and gated recurrent units (GRU), to estimate bus arrival

times based on real-world bus GPS data from the city of Tandil (Buenos Aires, Argentina). Experimental results demonstrate that deep learning models, particularly LSTM, can significantly outperform traditional approaches, highlighting their potential to optimize public transportation systems. In addition to developing predictive models, we provide a mobile application that integrates the prediction models, offering users real-time information on bus arrival times.

Keywords: Public Transportation, Deep Learning, Urban Mobility

1 Introducción

El transporte constituye un componente esencial en la vida cotidiana de las personas y representa un eje central en la planificación y el desarrollo urbano, especialmente ante el crecimiento poblacional y el aumento sostenido del tránsito vehicular (Hassannayebi et al., 2023). En este contexto, el diseño de sistemas de transporte público eficientes resulta clave para garantizar una movilidad urbana equitativa y sostenible (M. Chen et al., 2004; Hassannayebi et al., 2023). Esta necesidad es particularmente evidente en ciudades intermedias como Tandil (Buenos Aires, Argentina), donde el transporte público constituye el principal medio de desplazamiento para una gran parte de la población, al tratarse de un servicio accesible, económico y de uso extendido.

No obstante, uno de los principales desafíos que enfrentan los usuarios de estos sistemas es la incertidumbre respecto a los tiempos de llegada de los autobuses. La falta de información confiable genera frustración, pérdida de tiempo y una percepción negativa del servicio (Chowdhury & Ceder, 2016; Hassannayebi et al., 2023; Singh & Kumar, 2022). Esta problemática se ve agravada por la alta variabilidad en los tiempos de viaje, atribuible a factores como la congestión vehicular, la distribución de pasajeros y las condiciones meteorológicas. Dichas variaciones no solo disminuyen la calidad del servicio ofrecido, sino que también desincentivan su uso (M. Chen et al., 2004).

En este escenario, contar con modelos capaces de predecir con precisión los tiempos de arribo de los autobuses a sus puntos de parada o estaciones se vuelve fundamental para mejorar tanto la experiencia de los usuarios como la eficiencia operativa del sistema. La provisión de información actualizada en tiempo real puede mitigar los efectos negativos de las irregularidades en las frecuencias, optimizar la planificación de recorridos y facilitar la recuperación ante interrupciones del servicio (M. Chen et al., 2004). Sin embargo, la efectividad de estos modelos depende en gran medida de la calidad y confiabilidad de los datos empleados. Aspectos como la cobertura del GPS, la frecuencia de actualización y la presencia de datos incompletos o erróneos pueden comprometer seriamente la precisión de las predicciones. Por ello, garantizar la fiabilidad de los datos constituye un requisito indispensable en el desarrollo de soluciones predictivas robustas.

En este trabajo se propone la evaluación de distintos enfoques de predicción aplicados a la estimación de los tiempos de llegada de autobuses en la ciudad de Tandil, utilizando datos reales recolectados mediante dispositivos GPS instalados en las unidades. El objetivo principal es el análisis comparativo de diversos

enfoques de modelado que permitan generar estimaciones confiables en tiempo real, contribuyendo a una mejor planificación de viajes, la reducción de los tiempos de espera y, en última instancia, a una mayor satisfacción de los usuarios del sistema. Como complemento, se desarrolló una aplicación móvil³ que permite visualizar las predicciones generadas, facilitando el acceso a esta información por parte de la ciudadanía.

2 Trabajos relacionados

La predicción de los tiempos de llegada en sistemas de transporte público ha sido ampliamente estudiada debido a su relevancia para mejorar la eficiencia, la confiabilidad y la experiencia del usuario (Ge et al., 2024; Zhao et al., 2018). Los enfoques abarcan desde modelos estadísticos tradicionales hasta arquitecturas avanzadas de redes neuronales, dependiendo su elección de factores como la calidad de los datos, el contexto del sistema y los recursos disponibles (Ma et al., 2015). A continuación, se presentan trabajos relevantes que emplean las técnicas más comunes para estimar tiempos de traslado y llegada.

Uno de los enfoques más tradicionales de predicción es la regresión lineal, que modela la relación entre variables independientes y una variable dependiente. Cardona et al., 2015, por ejemplo, usaron este modelo para estimar los tiempos de llegada en Pereira (Colombia), basándose en distancia, velocidad y condiciones simuladas de tráfico. Si bien obtuvo buenos resultados en tramos simples (por ejemplo, sectores del recorrido sin paradas ni intersecciones semaforizadas), el modelo no pudo generalizar a contextos más dinámicos, debido al incumplimiento de supuestos claves del modelo lineal como normalidad, homocedasticidad e independencia de errores.

Otro enfoque clásico es el modelo ARIMA (modelos autorregresivos integrados de media móvil), apropiado para series temporales univariadas (aquellas que representan la serie de tiempo mediante una única variable) estacionarias. Suwardo et al., 2010 utilizaron este modelo para predecir los tiempos de viaje entre dos ciudades de Malasia utilizando información sobre estaciones/paradas, frecuencia del servicio y velocidades de los autobuses. Aunque los resultados fueron prometedores, la falta de comparación con otros modelos limita la evaluación de su efectividad. Por su parte, Rashidi y Ranjitkar, 2015 destacaron su precisión y simplicidad frente a modelos alternativos. Sin embargo, estos modelos usualmente suponen que los factores externos se mantienen constantes o se integran directamente en el modelo (Kumar et al., 2025), lo que puede limitar su aplicabilidad cuando las condiciones del tráfico varían a lo largo del día.

Además de los enfoques estadísticos y secuenciales, diversos modelos de clasificación han sido utilizados. Por ejemplo, Li, 2017 utilizó *Random Forest* para predecir los tiempos de llegada en Beijing, utilizando variables como número de intersecciones, distancia, tipo de día, precipitación y visibilidad. Los resultados superaron a modelos clásicos y de redes neuronales, aunque con riesgo de filtración por la partición de datos. De forma similar, Liu et al., 2021 utilizaron

³ Actualmente, la ciudad de Tandil no dispone de una aplicación oficial que brinde estimaciones en tiempo real sobre el transporte público.

K-NN para predecir el flujo de tráfico urbano en función de la latitud, longitud y flujo en un momento dado, reportando mejores resultados que *Random Forest* y ARIMA, aunque sin detallar volumen ni división de datos. Finalmente, Vangeneugden et al., 2024 utilizaron redes bayesianas para predecir retrasos en trenes en Bélgica, en función de los tiempos de llegada y partida a las estaciones, junto también con los retrasos de otros trenes en la misma estación. Si bien los autores evaluaron distintas variantes del modelo incluyendo diversas fuentes de información, no realizaron una comparación con otros enfoques, lo que dificulta evaluar la capacidad de generalización de los resultados.

El Modelo Oculto de Markov (HMM) permiten inferir variables no observables (por ejemplo, condiciones de tráfico o patrones de comportamiento vehicular) a partir de variables como ubicación o tiempos de llegada previos. Z. Chen et al., 2016 aplicaron un HMM para predecir tres condiciones de tráfico urbano (fluido, moderado, congestionado) utilizando datos históricos de flujo vehicular y velocidad en intersecciones. Comparado con métodos tradicionales como promedios históricos o modelos autorregresivos, el HMM mostró mejores resultados, especialmente durante horarios pico y días laborales. Sin embargo, su rendimiento disminuyó ante eventos imprevistos como accidentes, lo que sugiere que una integración con modelos híbridos o fuentes de datos en tiempo real podría mejorar su efectividad en entornos altamente dinámicos.

Ante las limitaciones de los modelos estadísticos tradicionales y los enfoques probabilísticos, las redes neuronales recurrentes, como LSTM (*Long Short-Term Memory*) y GRU (*Gated Recurrent Unit*), se han destacado por capturar patrones complejos en datos secuenciales de mediano a largo plazo. Ge et al., 2024 exploraron variantes del modelo LSTM para la predicción del tiempo de llegada de autobuses considerando la cantidad de recorridos completos realizados por el autobús hasta el momento, el tiempo del último recorrido, el tiempo de descanso entre recorridos, el tiempo entre distintos autobuses realizando el mismo recorrido y el tiempo de inicio del recorrido. Si bien los resultados parecieron prometedores, dado que todas las comparaciones realizadas solo incluyeron variantes de LSTM, no es posible determinar la superioridad del enfoque o la generalización de sus observaciones. Por su parte, Zhao et al., 2018 propusieron un modelo GRU integrando datos de sensores (velocidad y volumen vehicular). Los resultados mostraron una mejora significativamente en la precisión en comparación con otros modelos de redes neuronales, tanto para situaciones de tráfico normal, como para situaciones anormales como ser lluvia o la presencia de accidentes, lo que refuerza su aplicabilidad en escenarios reales.

En resumen, los enfoques clásicos como la regresión lineal destacan por su relativo bajo costo computacional y fácil interpretación, pero se ven limitados en entornos variables y relaciones no lineales. ARIMA mejora la predicción temporal, aunque pierde efectividad ante dinámicas complejas o factores externos variables. Los modelos de clasificación y probabilísticos, como *Random Forest* o HMM, ofrecen mejoras frente a métodos tradicionales, pero su efectividad depende fuertemente de cómo se representan los datos y de la calidad de las observaciones disponibles. Finalmente, las redes neuronales recurrentes, ofrecen un

equilibrio entre precisión y adaptabilidad, siendo capaces de capturar patrones temporales complejos en entornos cambiantes, aunque requieren mayores recursos computacionales y volúmenes significativos de datos para su entrenamiento.

3 Predicción de tiempos de llegada

3.1 Datos Utilizados

La calidad y cantidad de datos son claves en sistemas de predicción, ya que inconsistencias, valores faltantes o registros anómalos pueden afectar la precisión de los modelos y, por ende, la experiencia del usuario. En este estudio se utilizaron cerca de cuatro millones de registros correspondientes a seis líneas de autobuses urbanos de Tandil (500 a 505). Cada línea cuenta con un promedio de 750 mil registros, los cuales incluyen identificador del vehículo, marca temporal, velocidad y coordenadas geográficas (latitud y longitud)⁴.

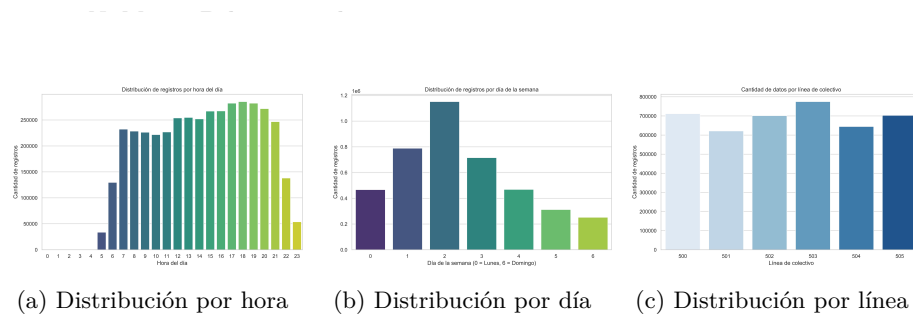
Cobertura temporal. La evaluación de la cobertura temporal de los datos es clave para asegurar que el modelo se adapte a distintos momentos del día y la semana. Se analizó la distribución horaria de los registros (Figura 1a), observándose mayor concentración durante las horas pico, en línea con la demanda del servicio. Sin embargo, se detectaron anomalías en los extremos horarios (alrededor de las 5:00 y 23:00 hs), posiblemente ligadas a variaciones operativas al inicio o fin de los recorridos. La distribución semanal (Figura 1b) resulta en general equilibrada, salvo por un leve aumento de registros los miércoles, posiblemente por una mayor cantidad de unidades en circulación. Durante los fines de semana, la caída en los registros refleja la reducción habitual en las frecuencias.

Distribución por línea de autobús y unidad. La cantidad de registros por línea (Figura 1c) es relativamente uniforme, destacándose la de recorrido más extenso y mayor demanda por su mayor volumen de datos. A nivel de unidad, se observaron mayores variaciones, atribuibles a diferencias en la frecuencia de circulación. Dado que el análisis se realiza de forma agregada por línea, estas disparidades no afectan significativamente la representatividad de los datos.

Registros nulos y duplicados. No se detectaron valores nulos, pero sí registros duplicados, definidos como aquellos con igual línea, unidad, marca temporal y coordenadas. En promedio cerca de 40 mil por línea (alrededor del 5 %) eran duplicados. Estos podrían deberse a errores de transmisión o capturas repetidas en intervalos breves. Al no aportar información adicional fueron eliminados.

Filtrado por recorrido. En el análisis de las velocidades se identificaron valores atípicos superiores a 120 km/h, incompatibles con el comportamiento de un autobús urbano. Al visualizar las ubicaciones asociadas, muchos de estos registros aparecían fuera del área de cobertura, en zonas rurales o autovías. Esto evidenció la necesidad de un filtrado geográfico para eliminar puntos fuera de

⁴ Los datos fueron recolectados por terceros en octubre y noviembre de 2017 desde una aplicación de transporte pública y oficial discontinuada en 2022. Los datos utilizados, el código de aplicación y el código para reproducir la evaluación se encuentran disponibles públicamente en el repositorio: <https://github.com/NicolasMP51/BusNow>.



(a) Distribución por hora (b) Distribución por día (c) Distribución por línea
 Figura 1: Gráficos de la cantidad de registros para cada hora del día, día de la semana y línea de autobús respectivamente.

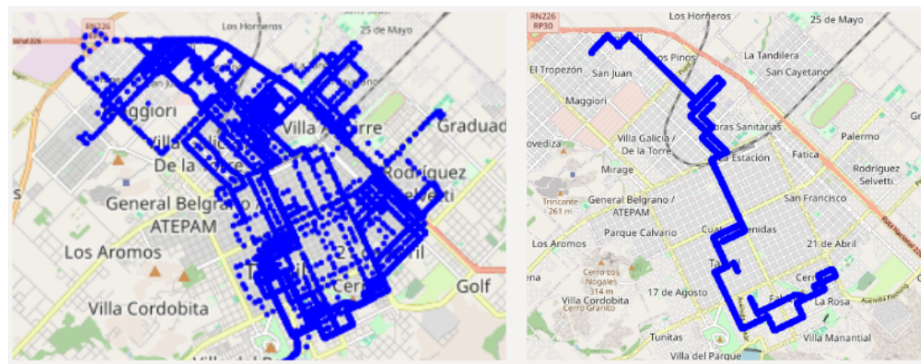


Figura 2: Ubicación de los registros de la línea 500 antes y después del filtrado

los recorridos oficiales. Esta etapa fue clave para asegurar que el conjunto de datos refleje adecuadamente las condiciones reales del sistema de transporte. La Figura 2 muestra el antes y después de este proceso para una de las líneas.

3.2 Modelos utilizados

A partir del análisis de los trabajos relacionados se seleccionaron cuatro modelos representativos: Regresión Lineal, ARIMA, LSTM y GRU. Esta elección responde tanto a la adecuación de estos enfoques a las características del conjunto de datos como a su potencial para generar estimaciones precisas en un entorno urbano real. La diversidad metodológica de los modelos seleccionados permite comparar el desempeño de distintas aproximaciones bajo condiciones operativas del sistema de transporte, atendiendo tanto a la complejidad temporal de los datos como a los requerimientos prácticos de implementación.

Modelo de Regresión Lineal. Se exploraron dos enfoques basados en regresión lineal múltiple. El primero consistió en estimar el tiempo de viaje entre pares de puntos geográficos, considerando como variables independientes la distancia, la velocidad, la hora del día y el día de la semana. Para ello, se agruparon trayectos por línea y unidad dentro de un mismo día y en intervalos horarios específicos. Si bien este enfoque resultó conceptualmente claro, generó un crecimiento significativo en el tamaño de los datos, lo que limitó su aplicabilidad en escenarios de entrenamiento y prueba realistas. El segundo enfoque propuso la

predicción de la velocidad de los autobuses como variable intermedia, modelando la línea y la unidad como variables explicativas. Esta alternativa buscó reducir la dimensionalidad y capitalizar la hipótesis de que la velocidad se encuentra más asociada al entorno geográfico que al recorrido específico. Este modelo ofreció una representación más eficiente y flexible de los datos, facilitando su integración con trayectos de distinta duración y resolución temporal.

Modelo ARIMA. Para aplicar ARIMA fue necesario transformar los datos de entrada a series temporales con intervalos regulares, en este caso, de un minuto. A partir de esta representación, se generaron modelos individuales por línea de autobús utilizando las posiciones geográficas registradas para estimar ubicaciones futuras en una ventana de n minutos. Dado que ARIMA permite estimar solo una variable a la vez, se ajustaron dos modelos paralelos por serie: uno para latitud y otro para longitud. También se exploró una variante que predecía la distancia recorrida entre puntos sucesivos; sin embargo, esta aproximación introdujo errores significativos al asumir trayectorias rectas, cuando en la práctica los recorridos incluyen curvas y desvíos.

Modelo LSTM. En la implementación inicial, se emplearon capas LSTM para modelar las dependencias temporales en los datos de ubicación, utilizando secuencias históricas de posiciones geográficas y tiempos de paso por dichas posiciones. En una primera instancia, el modelo se alimentó exclusivamente con estas secuencias, sin incorporar variables contextuales, con el fin de evaluar su capacidad para inferir coordenadas futuras a partir de la dinámica del recorrido. Posteriormente, se amplió la entrada con variables adicionales como la hora del día, la velocidad, el tiempo entre registros, el día de la semana y marcadores de inicio o fin de recorrido. Un desafío particular surgió al generar predicciones encadenadas: algunas variables adicionales podían derivarse de manera directa (como el tiempo entre paradas), pero otras, como la velocidad, tenían un impacto acumulativo en la trayectoria estimada. Para abordar este problema, se reestructuró el modelo para que también predijera la velocidad, garantizando así una coherencia entre la dinámica del recorrido y las estimaciones espaciales en función de la zona, el día y el horario.

Modelo GRU. Se implementó como alternativa a LSTM, buscando una arquitectura más liviana con menor cantidad de parámetros. Se replicaron las configuraciones empleadas previamente: se entrenaron modelos con distintos intervalos de muestreo (uno, dos y cinco minutos) y se compararon dos variantes. La primera consideraba únicamente las coordenadas geográficas como entrada, mientras que la segunda incorporaba variables contextuales adicionales, como velocidad, hora del día, día de la semana y tiempos entre registros. Esta evaluación permitió analizar el impacto de la reducción de complejidad arquitectónica sobre el desempeño predictivo, en escenarios comparables a los del modelo LSTM.

3.3 Aplicación Móvil

Como complemento a los modelos de predicción, se desarrolló una aplicación móvil orientada a mejorar la experiencia de viaje en el transporte público de

Tandil. La aplicación cuenta con un mapa interactivo que permite visualizar en tiempo real las líneas de autobús, sus recorridos, estaciones/paradas y la ubicación actual de las unidades. A partir de estos datos, se estiman los tiempos de llegada a cada parada utilizando el modelo LSTM entrenado previamente. La aplicación incluye una funcionalidad de planificación de viajes que permite al usuario calcular rutas entre dos puntos de la ciudad. También ofrece la posibilidad de guardar lugares frecuentes y consultas recientes para facilitar el acceso a la información. La lógica de predicción y gestión de datos se realiza en un servidor central, lo que reduce la carga de procesamiento en los dispositivos móviles y garantiza la actualización constante de la información.

La aplicación fue desarrollada utilizando *Android Studio* y el lenguaje de programación *Kotlin*, mientras que el servidor se implementó en *Python*. Para el manejo de datos en tiempo real, se integró *Firebase Realtime Database*, lo que permite actualizar constantemente la ubicación de los autobuses. Además, se incorporó la API de *Google Maps* para la visualización e interacción con el mapa de manera intuitiva y eficiente.

4 Evaluación experimental

Para la implementación de los distintos modelos de predicción se emplearon bibliotecas tales como *TensorFlow*, *Keras*, *NumPy* y herramientas de *scikit-learn* para las etapas de preprocesamiento y evaluación. En el caso del modelo de regresión lineal, se utilizó la clase *Sequential* de *Keras*, construyendo una arquitectura compuesta por múltiples capas densas con funciones de activación ReLU, regularización L2 y capas de Dropout con el objetivo de mitigar el sobreajuste. El modelo fue entrenado utilizando el optimizador Adam, con la función de pérdida MAE y la métrica adicional MSE para el monitoreo del rendimiento. Para los modelos ARIMA, se utilizó la biblioteca *pmdarima*, específicamente la función *auto_arima*, la cual permite realizar una búsqueda automática de los hiperparámetros del modelo, orden (p,d,q), para cada serie temporal (latitud y longitud), sin incorporar estacionalidad. En el caso de las arquitecturas basadas en redes neuronales recurrentes, tanto los modelos LSTM como GRU fueron implementados mediante la clase *Sequential*, combinando una o más capas LSTM o GRU (según correspondiera), seguidas por capas densas. Ambos modelos fueron entrenados utilizando el optimizador Adam y la función de pérdida MSE. En lo que respecta a hiper-parámetros, se evaluaron diferentes configuraciones estructurales, como la cantidad de unidades por capa y la disposición de las capas, a fin de analizar su impacto en el desempeño, seleccionando la combinación con mejor rendimiento. Finalmente, para la evaluación del desempeño de los modelos se emplearon métricas comúnmente utilizadas en tareas de regresión: MAE, MSE, RMSE y R² (esta última únicamente aplicada en los modelos de regresión lineal y ARIMA).

Modelo regresión lineal. A partir de los datos previamente procesados, analizados y filtrados, se calculó el tiempo de viaje entre pares de puntos geográficos para cada combinación de línea y unidad de autobús dentro de un mismo día y en franjas horarias específicas. Esta metodología generó más de 8.000 pares por

Tabla 1: Resultados del Modelo de Regresión Lineal
(a) Predicción posición (b) Predicción velocidad

Line	MAE	MSE	RMSE	R ²
500	8.827	125.417	11.199	0.101
501	8.857	128.901	11.353	0.216
502	10.311	169.371	13.014	0.097
503	9.917	156.084	12.493	0.194
504	9.336	141.159	11.881	0.084
505	8.963	135.187	11.627	0.187

Line	MAE	MSE	RMSE	R ²
500	8.827	125.417	11.199	0.101
501	8.857	128.901	11.353	0.216
502	10.311	169.371	13.014	0.097
503	9.917	156.084	12.493	0.194
504	9.336	141.159	11.881	0.084
505	8.963	135.187	11.627	0.187

cuarteto (línea, unidad, día, horario), lo que extrapolado a todas las combinaciones posibles durante el período de observación resultó en millones de registros, dificultando tanto su procesamiento como su evaluación en un entorno realista. En este contexto, se optó por trabajar con un subconjunto de aproximadamente 10.000 registros, correspondientes a una línea y unidad específicas, en el rango horario de 11:00 a 13:00 del día 15/10/2017. Esta fecha fue seleccionada por ser el primer día con registros disponibles y por coincidir con un domingo, lo que presumiblemente reduce la variabilidad causada por el tránsito y la demanda del servicio, ofreciendo mayor regularidad en los recorridos. Los resultados obtenidos (Tabla 1a) mostraron un desempeño variable del modelo según la línea de autobús analizada. Si bien algunas líneas presentaron valores razonables de R^2 , los MAE y RMSE continúan siendo elevados en términos prácticos.

Para el segundo enfoque se consideró la estimación de velocidades por horario y posición. Con dicha información, los tiempos de viaje podrían calcularse posteriormente utilizando la distancia recorrida en cada segmento y la velocidad estimada para ese tramo. Para este modelo, se incluyó la línea y la unidad como variables explicativas, bajo la hipótesis de que la velocidad se encuentra más relacionada con la zona geográfica que del recorrido específico. Para su entrenamiento se utilizaron, aproximadamente 2,2 millones de registros, reservando cerca de un millón para evaluación, aplicando una división aleatoria 70/30. Sin embargo, los resultados obtenidos con esta segunda estrategia (Tabla 1b) fueron incluso inferiores a los del modelo original de predicción de tiempos de viaje. Estos resultados evidencian su falta de capacidad para capturar patrones no lineales y variaciones temporales frecuentes, propias de la tarea.

Modelo ARIMA. Dado que este modelo está diseñado para estimar una única variable a lo largo del tiempo, fue necesario definir dos modelos, uno para latitud y otro para longitud. Sin embargo, tras evaluar distintos modelos para cada línea de autobús, se observó que las predicciones de los siguientes 10 pasos/minutos resultaban altamente imprecisas, ubicando incluso los puntos predichos fuera del recorrido real de la línea (ejemplificado en la Figura 3 para la línea 505). Este comportamiento no fue exclusivo de un momento específico, sino que se repitió sistemáticamente para cualquier combinación de día/hora evaluada.

Asimismo, se observó que cuando las variaciones entre coordenadas consecutivas eran pequeñas, el modelo tenía dificultades para identificar patrones significativos. ARIMA es más efectivo cuando existe una tendencia clara en los datos, y si las diferencias espaciales entre puntos no se ajustan bien a un modelo basa-

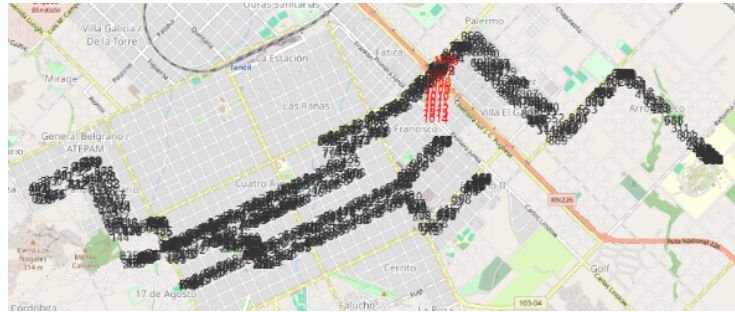


Figura 3: Predicciones del Modelo ARIMA para posiciones (línea 505)

do en diferencias, es probable que amplifique pequeñas fluctuaciones en lugar de capturar una estructura relevante. Por otra parte, ARIMA es un modelo lineal, lo que limita su capacidad para representar trayectorias no lineales. En recorridos con curvas, giros o trayectorias zigzagueantes, las coordenadas espaciales no siguen un patrón lineal predecible, y el modelo intenta ajustarse incorrectamente a una estructura inexistente. Otro inconveniente relevante fue el modelado separado de la latitud y la longitud, lo que ignora su relación geográfica intrínseca. Como resultado, puede ocurrir que una coordenada predicha sea razonable, pero su combinación con la otra no represente una ubicación coherente.

Frente a estos problemas, se exploraron alternativas sin lograr mejoras. Una de las variantes consistió en predecir la distancia entre puntos consecutivos en lugar de las coordenadas geográficas. Sin embargo, este enfoque asumía trayectorias rectilíneas entre puntos, cuando en realidad los recorridos podían incluir curvas o desvíos que afectaban considerablemente la precisión. Se observó que, si bien los primeros pasos de la predicción presentaban errores moderados, estos se acumulaban rápidamente con el tiempo, volviendo el modelo ineficaz para estimaciones a largo plazo. La Tabla 2 ilustra las predicciones de distancia recorrida frente al valor real, así como los errores acumulados después de 20 minutos, para la línea 505. Aunque los errores iniciales eran relativamente bajos, luego de 20 minutos la desviación superaba el kilómetro en todos los modelos evaluados (y en algunos casos los 2 kilómetros), indicando una pérdida sustancial de precisión.

En conclusión, aunque ARIMA puede ofrecer buenos resultados en condiciones ideales con datos lineales y bien estructurados, las particularidades de los datos espaciales utilizados en este trabajo no se ajustaban a sus supuestos matemáticos. Dado que el modelo mostró limitaciones incluso en contextos reducidos y controlados, resultaba improbable que pudiera escalar efectivamente a conjuntos de mayor tamaño o complejidad.

Modelos LSTM y GRU. Los modelos LSTM y GRU fueron entrenados y evaluados utilizando tres conjuntos de datos diferenciados por el intervalo temporal entre mediciones: 1, 2 y 5 minutos. Esta segmentación permitió analizar cómo influye la frecuencia de muestreo en el rendimiento predictivo y el costo computacional. Para cada conjunto, se implementaron dos variantes: una *símplice*, que solo incluye las coordenadas (latitud y longitud), y otra *compleja*, que

Tabla 2: Predicciones del Modelo ARIMA para distancias (en metros).

Minuto	Predicción	Valor Real	Error	Acumulado	Minuto	Predicción	Valor Real	Error	Acumulado
1	239.983	427.471	-187.488	187.488	11	262.594	385.123	-122.529	1149.229
2	234.134	0.800	233.334	-46.646	12	263.801	266.459	-2.658	1151.887
3	245.659	694.863	-449.205	402.559	13	264.856	312.007	-47.152	1199.039
4	247.966	349.194	-101.228	503.788	14	265.766	463.111	-197.345	1396.383
5	251.373	337.537	-86.164	589.952	15	266.566	0.800	265.766	1129.817
6	253.076	187.831	65.246	524.706	16	267.267	1162.909	-895.642	2025.459
7	255.883	275.581	-19.779	544.485	17	267.889	156.270	111.619	1913.849
8	257.899	351.932	-94.033	648.518	18	268.416	601.697	-333.280	2247.129
9	259.732	411.947	-152.215	790.732	19	268.885	386.873	-117.989	2365.113
10	261.219	497.186	-235.967	1026.699	20	269.294	200.846	68.449	2296.664

Tabla 3: Comparación de Tiempo de Entrenamiento para los Distintos Modelos

Línea	Modelo	Tiempo (seg)	Línea	Modelo	Tiempo (seg)
500	GRU simple	531.992	503	GRU simple	509.841
	LSTM simple	505.397		LSTM simple	427.147
	GRU complejo	862.302		GRU complejo	843.859
	LSTM complejo	784.059		LSTM complejo	858.832
501	GRU simple	414.565	504	GRU simple	420.465
	LSTM simple	384.451		LSTM simple	399.969
	GRU complejo	782.329		GRU complejo	762.256
	LSTM complejo	782.348		LSTM complejo	709.761
502	GRU simple	476.864	505	GRU simple	456.136
	LSTM simple	453.014		LSTM simple	401.285
	GRU complejo	900.428		GRU complejo	815.527
	LSTM complejo	772.071		LSTM complejo	765.212

incorpora variables adicionales como velocidad, hora del día y día de la semana, y además predice la velocidad como salida secundaria.

Los resultados obtenidos⁵ muestran que ambos modelos, especialmente LSTM, son capaces de capturar patrones temporales complejos cuando los datos tienen una alta resolución temporal. Los mejores desempeños se obtuvieron con intervalos de 1 minuto, donde los errores de predicción fueron significativamente menores. Sin embargo, este beneficio presenta como desventaja un mayor tiempo de entrenamiento, producto del mayor volumen de datos. A medida que se incrementó el intervalo (2 y 5 minutos), la precisión de los modelos disminuyó de manera considerable, lo cual se atribuye a la pérdida de información sobre variaciones temporales sutiles entre puntos consecutivos. GRU mostró un comportamiento muy similar al de LSTM respecto a la sensibilidad a la frecuencia de muestreo. En intervalos de 1 minuto, los errores fueron comparables a los de LSTM, mientras que con intervalos más amplios el rendimiento decayó. Esto refuerza la importancia de contar con datos de alta frecuencia para lograr predicciones precisas en trayectorias de vehículos.

La Tabla 3 muestra los tiempos de entrenamiento para cada modelo y línea de autobús. En general, los modelos *complejos* requirieron más tiempo de entrenamiento que los modelos *simples*, debido a un mayor número de parámetros y mayor complejidad del problema. Contrario a lo que se esperaría por la arquitectura más liviana de los GRU, los modelos LSTM resultaron, en promedio,

⁵ Las tablas correspondientes a los intervalos 2 y 5 pueden encontrarse en el repositorio.

Tabla 4: Comparación de métricas para los distintos modelos

Línea	Modelo	MAE	MSE	Línea	Modelo	MAE	MSE
500	GRU simple	9.150e-4	2.565e-6	503	GRU simple	1.174e-3	5.640e-6
	LSTM simple	8.370e-4	2.374e-6		LSTM simple	1.099e-3	5.519e-6
	GRU complejo	8.700e-4	2.440e-6		GRU complejo	1.135e-3	5.599e-6
	LSTM complejo	8.390e-4	2.350e-6		LSTM complejo	1.149e-3	5.622e-6
501	GRU simple	9.810e-4	3.105e-6	504	GRU simple	9.370e-4	3.054e-6
	LSTM simple	9.570e-4	3.057e-6		LSTM simple	9.630e-4	3.093e-6
	GRU complejo	9.640e-4	3.091e-6		GRU complejo	8.970e-4	3.016e-6
	LSTM complejo	9.350e-4	3.076e-6		LSTM complejo	9.150e-4	2.905e-6
502	GRU simple	8.770e-4	2.086e-6	505	GRU simple	3.137e-3	2.907e-5
	LSTM simple	9.000e-4	2.089e-6		LSTM simple	3.119e-3	2.817e-5
	GRU complejo	8.920e-4	2.078e-6		GRU complejo	3.151e-3	2.913e-5
	LSTM complejo	8.660e-4	1.983e-6		LSTM complejo	3.079e-3	2.772e-5

ligeramente más rápidos que los GRU equivalentes. Esto sugiere que, más allá del número de parámetros, otros factores como el tamaño de las capas, el optimizador utilizado o la naturaleza de los datos pueden influir significativamente en el entrenamiento.

En lo que respecta a la predicción de un solo paso (1 minuto, Tabla 4), en general, tanto los modelos LSTM como los GRU mostraron un desempeño comparable en la mayoría de las líneas, aunque los *LSTM complejos* obtuvieron los mejores resultados. Su mayor capacidad para capturar dependencias a largo plazo podría explicar esta ventaja. Para las líneas 500, 501, 502 y 504, las diferencias entre modelos fueron mínimas, indicando que cualquiera de las variantes podría ser utilizada con buenos resultados. En cambio, en la línea 503 se registraron errores más altos en todos los modelos, con el *LSTM simple* mostrando el mejor desempeño relativo. La línea 505 presentó los mayores errores en general, siendo también en este caso el *LSTM complejo* el modelo con menor MSE.

Para evaluar la viabilidad práctica de estos modelos, se realizaron 20 predicciones consecutivas por línea (Tabla 5). En lo que respecta al tiempo de inferencia, se observaron diferencias mínimas entre los modelos, rondando aproximadamente un segundo por secuencia. El modelo *LSTM complejo* destacó por su baja media y mediana de error en varias líneas, y por presentar desviaciones estándar menores, lo que refleja mayor estabilidad en las predicciones. Por ejemplo, en la Línea 500 alcanzó una media de 345.77 (± 448.92) metros. En contraposición, los modelos simples (tanto LSTM como GRU) presentaron errores mayores y más dispersos; en la Línea 502, por ejemplo, el *GRU simple* mostró una media de 498.13 (± 593.42). El *GRU complejo*, si bien no fue consistentemente el mejor, tuvo un rendimiento competitivo en algunas líneas. En la Línea 501, por ejemplo, superó al *LSTM complejo* en consistencia, con una desviación estándar más baja, aunque con una media levemente superior.

En conclusión, tanto LSTM como GRU funcionaron adecuadamente en contextos de alta resolución temporal, pero el modelo *LSTM complejo* se posicionó como la opción más adecuada para predicciones encadenadas, al lograr un equilibrio superior entre precisión, estabilidad y eficiencia.

Tabla 5: MAE para predicciones encadenadas

Línea	Modelo	Media (s Desv. Est.)	Mediana	Línea	Modelo	Media (s Desv. Est.)	Mediana
500	GRU simple	625.58 (s 687.81)	381.28	503	GRU simple	392.03 (s 451.56)	261.15
	LSTM simple	694.31 (s 838.91)	407.90		LSTM simple	479.26 (s 525.22)	305.09
	GRU complejo	406.63 (s 326.77)	315.78		GRU complejo	572.00 (s 544.15)	406.04
	LSTM complejo	675.27 (s 689.73)	448.24		LSTM complejo	364.95 (s 443.61)	233.40
501	GRU simple	498.13 (s 593.42)	295.53	504	GRU simple	1367.26 (s 1220.77)	960.18
	LSTM simple	385.53 (s 451.00)	244.22		LSTM simple	1269.43 (s 1213.76)	846.62
	GRU complejo	365.51 (s 416.01)	235.34		GRU complejo	1265.83 (s 1189.85)	867.97
	LSTM complejo	395.56 (s 405.79)	270.82		LSTM complejo	1268.29 (s 1213.93)	854.31
502	GRU simple	888.63 (s 986.16)	595.59	505	GRU simple	394.85 (s 481.99)	252.95
	LSTM simple	669.15 (s 945.34)	382.69		LSTM simple	663.18 (s 820.04)	371.01
	GRU complejo	597.12 (s 845.23)	350.10		GRU complejo	438.58 (s 525.82)	278.29
	LSTM complejo	570.94 (s 816.12)	335.97		LSTM complejo	345.77 (s 448.92)	222.74

5 Conclusiones

Este trabajo abordó la estimación de los tiempos de llegada de autobuses en la ciudad de Tandil (Buenos Aires, Argentina) con el objetivo de mejorar la experiencia de los usuarios del transporte público. Se exploraron diversos enfoques metodológicos para la predicción de tiempos de llegada, comparando técnicas clásicas y modelos basados en redes neuronales. Para ello, se implementaron y evaluaron modelos predictivos como regresión lineal, ARIMA, LSTM y GRU, permitiendo identificar sus fortalezas y limitaciones en el contexto local. Además, se desarrolló una aplicación móvil que visualiza en tiempo real tanto la ubicación de los autobuses como las predicciones generadas, potencialmente facilitando el acceso a esta información por parte de la ciudadanía.

El trabajo, no obstante, presenta ciertas limitaciones. La disponibilidad y calidad de los datos afectaron la precisión de los modelos, particularmente en situaciones atípicas como desvíos de los recorridos predefinidos o eventos no recurrentes, los cuales requirieron un proceso exhaustivo de limpieza y preprocesamiento. Por otro lado, si bien se consideraron modelos avanzados, su implementación práctica puede estar condicionada por los recursos computacionales disponibles. También quedó fuera del alcance del trabajo una validación de la aplicación móvil complementaria y de la calidad percibida de los modelos de predicción con usuarios o actores institucionales.

Más allá de su dimensión técnica, el trabajo tiene un potencial impacto social significativo. Un sistema de transporte más predecible y eficiente podría contribuir a mejorar la calidad de vida de los habitantes, reducir los tiempos de espera y aumentar la confianza en el uso del transporte público. Para los actores institucionales, el trabajo tiene el potencial de colaborar en el diseño y la optimización de la gestión del transporte, contribuyendo a decisiones sobre la cantidad de unidades en circulación en momentos específicos, la definición de recorridos o incluso la ubicación de paradas. A largo plazo, estas mejoras podrían promover una movilidad urbana más sostenible e inclusiva, especialmente en los sectores de la población más dependientes del transporte público.

En el futuro, el sistema podría mejorar con la incorporación de nuevas fuentes de información y el ajuste en tiempo real de los modelos a condiciones cambiantes, como congestiones, accidentes o el estado meteorológico. Además, el desarrollo de herramientas de monitoreo y visualización para los operadores contribuiría

a consolidar una plataforma integral de apoyo a la gestión del transporte público en ciudades intermedias como Tandil.

Referencias

- Cardona, O. A. S., Zea, J. D. H., Santa Chávez, J. J., & Escobar, J. W. (2015). Modelos de regresión lineal para estimación de tiempos de viaje en sistemas de transporte masivo. *Cienc. Ing. Neogranad.*, 25(1), 77-89.
- Chen, M., Liu, X., Xia, J., & Chien, S. I. (2004). A dynamic bus-arrival time prediction model based on APC data. *Computer-Aided Civil and Infrastructure Engineering*, 19(5), 364-376.
- Chen, Z., Wen, J., & Geng, Y. (2016). Predicting future traffic using hidden markov models. *2016 IEEE 24th international conference on network protocols (ICNP)*, 1-6.
- Chowdhury, S., & Ceder, A. A. (2016). Users willingness to ride an integrated public-transport service: A literature review. *Transp. Pol.*, 48, 183-195.
- Ge, Z., Yang, L., Li, J., Chen, Y., & Xu, Y. (2024). Bus Schedule Time Prediction Based on LSTM-SVR Model. *Mathematics*, 12(22), 3589.
- Hassannayebi, E., Farjad, A., Azadnia, A., Javidi, M., & Chunduri, R. (2023). A data analytics framework for reliable bus arrival time prediction using artificial neural networks. *Int. Jour. of Data Science & Analytics*, 1-20.
- Kumar, B. A., Singh, R., Shaji, H. E., & Vanajakshi, L. (2025). Bus Arrival Time Prediction: A Comprehensive Review. *IEEE Trans. Intell. Transp.*
- Li, J. (2017). Bus arrival time prediction based on random forest. *2017 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology (FMSMT 2017)*, 867-872.
- Liu, Y., Yu, H., & Fang, H. (2021). Application of KNN prediction model in urban traffic flow prediction. *2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT)*, 389-392.
- Ma, X., Tao, Z., Wang, Y., Yu, H., & Wang, Y. (2015). Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *ransp. Res. Part C Emerg.*, 54, 187-197.
- Rashidi, S., & Ranjitkar, P. (2015). Estimation of bus dwell time using univariate time series models. *Journal of advanced transportation*, 49(1), 139-152.
- Singh, N., & Kumar, K. (2022). A review of bus arrival time prediction using artificial intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12(4), e1457.
- Suwardo, W., Napiah, M., & Kamaruddin, I. (2010). ARIMA models for bus travel time prediction. *J. Inst. Eng. Malaysia*, 71(2), 49-58.
- Vangeneugden, M., Luong, N. Q., & Mercelis, S. (2024). Towards Efficient Rail Transportation: Bayesian Network Modeling for Predicting Passenger Train Delays Using Secondary Train Information. *2024 IEEE Conference on Artificial Intelligence (CAI)*, 1425-1431.
- Zhao, J., Gao, Y., Qu, Y., Yin, H., Liu, Y., & Sun, H. (2018). Travel time prediction: Based on gated recurrent unit method and data fusion. *IEEE Access*, 6, 70463-70472.