UNIVERSIDAD
DE GRANADA

Escuela Técnica Superior de Ingenierías Informática y
Telecomunicación

Máster Universitario en Ciencia de Datos e Ingeniería de Computadores

TRABAJO DE FIN DE MÁSTER

# Estudio sobre la Efectividad del Positional Encoding en Transformers para Series Temporales y Diseño de Mecanismos Adaptados

Presentado por:
Cristhian Moya Mota

Curso académico 2024-2025

# Estudio sobre la Efectividad del Positional Encoding en Transformers para Series Temporales y Diseño de Mecanismos Adaptados

Cristhian Moya Mota

Cristhian Moya Mota *Estudio sobre la Efectividad del Positional Encoding en Transformers para Series Temporales y Diseño de Mecanismos Adaptados*.

Trabajo de fin de Grado. Curso académico 2024-2025.


**Responsable de tutorización**

Julian Luengo Martín
*DECSAI*

Diego Jesús García Gil
*LSI*

Máster Universitario en Ciencia de Datos e Ingeniería de Computadores

Escuela Técnica Superior de Ingenierías Informática y Telecomunicación

Universidad de Granada

Declaración de originalidad

D./Dña. Cristhian Moya Mota

Declaro explícitamente que el trabajo presentado como Trabajo de Fin de Grado (TFG), correspondiente al curso académico 2024-2025, es original, entendido esto en el sentido de que no he utilizado para la elaboración del trabajo fuentes sin citarlas debidamente.

En Granada a 29 de julio de 2025

Fdo: Cristhian Moya Mota

*Dedicatoria (opcional)*

# Agradecimientos

Agradecimientos (opcional, ver archivo `preliminares/agradecimiento.tex`).

# Summary

An english summary of the project (around 800 and 1500 words are recommended).
File: `preliminares/summary.tex`

# Índice general

# 1. Introducción

En este primer punto, describiremos de forma breve la motivación a realizar este TFM. Analizaremos la importancia de poseer herramientas capaces de realizar una predicción a largo plazo en series temporales (en inglés, *Long-term Time Series Forecasting*, LSTF), pero, sobre todo, centrándonos en un aspecto clave: la codificación posicional (*Positional Encoding*, en adelante, PE), común a prácticamente todas las alternativas actuales del estado del arte.

Procederemos a justificar su importancia, así como a formular los objetivos que se cubren con la realización de este trabajo.

## 1.1. Motivación

En la actualidad, los datos son uno de los bienes más preciados. Las telecomunicaciones nos han permitido alcanzar un volumen inimaginable de información digital, la cual no somos prácticamente capaces de procesar, y extraer conocimiento útil se convierte en una tarea complicada. Su gran variedad y modalidad hace necesario disponer de modelos multimodales cada vez más complejos para procesarlos, como los modelos fundacionales [1], para tratar así de disponer de una herramienta cercana a ser capaz de procesar todo tipo de información.

Hemos podido apreciar grandes avances en el procesamiento de texto, con los grandes modelos de lenguaje como GPT, el cual se ha convertido en una herramienta que usamos habitualmente para resolver nuestras dudas. Ahora, incluso permite generar imágenes, video y audio, y pensar profundamente las respuestas.

Sin embargo, estos no son los únicos tipos de datos que podemos emplear para aprender. Los datos basados en flujos, y las series temporales, son un recurso clave que podemos emplear para resolver multitud de problemáticas. Podemos predecir a largo plazo, clasificar fenómenos, o bien, incluso detectar anomalías. En este trabajo, nos centraremos sobre todo en la primera tarea y las dificultades que existen en este ámbito.

### 1.1.1. Importancia de las Series Temporales

Las series temporales son un tipo de datos caracterizados por una secuencia organizada en el tiempo. Generalmente, consisten en una o varias variables observadas a intervalos regulares, con el objetivo de registrar la evolución de un fenómeno. Suelen recogerse de forma periódica mediante sensores o procesos automatizados (por ejemplo, la temperatura de un motor medida por un termostato electrónico), aunque también pueden originarse de forma irregular, como en el caso de registros manuales. Es importante que, cuando se trata de múltiples variables, estas se muestreen con la misma frecuencia para facilitar su análisis y evitar la necesidad de técnicas de imputación, que podrían introducir distorsiones en los resultados.

El objetivo: utilizar estos datos para aprender qué ocurrirá en instantes futuros, realizando predicciones en un horizonte concreto sobre el que es desconocido su comportamiento. Sin embargo, es importante destacar que no todos los fenómenos pueden ser predichos aunque dispongamos de un conjunto de datos suficiente y adecuado. Es esencial que, para que la técnica sea efectiva, estemos ante una tarea de forecasting, es decir, de predicción de elementos futuros, pero pudiéndonos apoyar en información pasada para comprender el fenómeno que estamos estudiando. Por ejemplo, tratar de predecir la temperatura que hará en las próximas 2 horas, en intervalos de 15 minutos,es una tarea viable: disponemos de gran cantidad de recursos históricos, como la tendencia de las horas anteriores, y los valores históricos de días previos, o incluso, el comportamiento en años anteriores, que pueden ser útiles.

Pero también podríamos enfrentarnos a escenarios más complejos, como el que gestiona diariamente Red Eléctrica para estimar el consumo de electricidad y responder con los tipos de energía más adecuados, evitando tanto el exceso como el déficit en la red. Para ello, se apoyan no solo en datos históricos de consumo, sino también en la información en tiempo real proveniente de sensores y mediciones realizadas en distintas estaciones eléctricas, lo que permite anticipar fluctuaciones y minimizar problemas de sincronización. A esto se suman otros factores externos que pueden influir, como las condiciones meteorológicas o incidencias en redes interconectadas de países vecinos. Tras la experiencia vivida en abril de 2025, queda en evidencia la importancia de esta tarea, ya que en caso de error, las consecuencias pueden ser muy graves: personas atrapadas en ascensores, hospitales en emergencia, problemas de tráfico, interrupción de la cadena de frío en los alimentos... Las pérdidas económicas se estimaron entre 1.600 y 2.500 millones de euros.

Por tanto, disponer de buenas técnicas de estudio para series temporales es una herramienta clave en multitud de aplicaciones. Manualmente, podemos realizar aproximaciones posibles para casos sencillos, donde apreciamos un comportamiento repetitivo en el tiempo de la serie (estacionalidad) o un comportamiento estrictamente creciente o decreciente linealmente (tendencia). Sin embargo, cuando el problema se vuelve más complejo o involucra múltiples variables, recurrir a matemáticas simples o técnicas intuitivas ya no es suficiente. En estos casos, es necesario contar con herramientas más sofisticadas que permitan modelar el comportamiento de la serie de forma aproximada, pero lo más fiel posible a la realidad.

Es aquí donde entran en juego métodos ampliamente utilizados, como la descomposición de la serie en componentes fundamentales: tendencia, estacionalidad y ruido. Una de las técnicas estadísticas más reconocidas en este ámbito es ARIMA (AutoRegressive Integrated Moving Average) [2]. Se basa en utilizar una ventana deslizable sobre la que ir calculando una media móvil, es decir, un intervalo de amplitud $q$ en torno a un valor central, el cual se va moviendo de izquierda a derecha para evaluar con todas las instancias de la serie, y en una componente autoregresiva, que trata de incorporar información de instantes anteriores al que se está evaluando (hasta p valores). También se integra la serie tantas veces como sea necesario para suavizar su comportamiento. De este modo, es posible generar predicciones razonables, siempre y cuando el problema cumpla con ciertas condiciones que abordaremos más adelante.

No obstante, incluso técnicas como ARIMA presentan limitaciones importantes, especialmente cuando se enfrentan a horizontes de predicción amplios. A medida que intentamos anticipar valores más alejados en el tiempo, el modelo se ve obligado a basarse cada vez más

en sus propias predicciones previas en lugar de los datos observados, lo que provoca una acumulación progresiva de errores.

Esto plantea la necesidad de modelos más robustos, capaces de capturar dependencias de largo alcance y manejar múltiples variables de forma conjunta, tratando de minimizar el error evitando su acumulación. Podemos recurrir, como alternativa, a una de las herramientas ya empleadas en los grandes modelos de lenguaje que mencionamos anteriormente: los Transformers [11]. Pero su uso no es directo; debemos de adaptar su funcionamiento a las series temporales, ya que estos modelos, originalmente diseñados para tareas de lenguaje natural, no conservan de manera adecuada la información temporal del orden de los datos.

### 1.1.2. La dificultad de predicción a largo plazo: información posicional

Para poder adaptar modelos basados en Transformers para lenguaje natural a series temporales, es imprescindible introducir mecanismos de codificación posicional (positional encoding) que permitan al modelo interpretar correctamente la secuencia en el tiempo [7]. Sin este componente, los Transformers serían incapaces de distinguir el orden de la entrada, lo cual es esencial para predecir correctamente la evolución de un fenómeno.

Encontrar una codificación eficiente, sencilla y coherente con la estructura de los datos se encuentra en continua búsqueda en el estado del arte, pero la mayoría de ellas presentan los siguientes inconvenientes:

- **Falta de captura de la estructura**. En muchas ocasiones, se opta por utilizar codificaciones sencillas basadas en senos y cosenos, de la misma forma que se hace en procesamiento de lenguaje, pero de esta forma, estamos perdiendo información de patrones estacionales e información local de interés. Esto ocurre, por ejemplo, en modelos como Informer [12], uno de los primeros en adaptar los Transformers a series temporales.
- **Dificultad para adaptarse a diferentes escalas temporales y falta de semántica**. Propuestas simples, como la ya mencionada codificación sinusoidal, pueden ser ineficientes cuando se busca trabajar con distintas granularidades temporales. Al tratarse de una codificación fija, sus valores no se ajustan al cambiar la frecuencia de muestreo (por ejemplo, de segundos a minutos) , ya que mantienen una amplitud y periodicidad predefinidas. Por otro lado, aproximaciones más complejas, como las basadas en convoluciones, intentan capturar patrones locales dentro de la secuencia, pero tienden a centrarse en un corto plazo, lo que dificulta la detección de relaciones de largo alcance o multiescala. Además, este tipo de codificación no incorpora explícitamente la semántica temporal (como la hora del día o el día de la semana), lo que limita su interpretabilidad y generalización en tareas donde esa información es relevante.
- **Complejidad**. Para paliar las dificultades de los métodos más simples, se han evaluado alternativas más complejas, que permiten aumentar el rendimiento: es el caso de Autoformer, que incorpora información autoregresiva al modelo, tomando así cierta similaridad con la componente AR de ARIMA; o bien, se proponen encodings basados en transformadas de Fourier. Pero estas aumentan la necesidad de cómputo y el tiempo necesario para su entrenamiento, lo cual hace surgir otra vertiente alternativa, basada en modelos más sencillos, como Reformer [9], que trata de convertir la eficiencia cuadrática de los Transformers, a lineal. O Informer, que persigue lograr eficiencia

*n log n* mediante su ProbSparse Attention [12]. Pero, a veces puede conseguirse el efecto contrario: empeorar en exceso los resultados, a consta de un menor tiempo de entrenamiento e inferencia. Por ejemplo, en Informer, al muestrearse en atención solo ciertos vectores de entrada, perdemos información local que puede ser clave.

La misión de este trabajo se centra en tratar de encontrar una alternativa que sea capaz de mantenerse cercana a la semántica del problema, permitiendo adaptabilidad en cuanto a parámetros, y tratando de alcanzar un equilibrio entre eficiencia y rendimiento.

## 1.2. Justificación

Tras estudiar el problema, surge la necesidad de crear una alternativa adecuada a los modelos de codificación existentes, incorporando dentro de ella la semántica del propio problema, e información local que permita una mayor adaptabilidad de los modelos creados, reduciendo las tasas de error. Aunque este objetivo ya es tratado por diversos modelos y líneas de investigación del estado del arte, en este trabajo buscamos replantear el positional encoding para que sea adaptable y capaz de incorporar información global y local, dando mayor importancia a una u otra en función de la entrada dada.

Todo esto se plantea aprovechando el conocimiento adquirido a partir de técnicas más clásicas, como ARIMA, e incorporando información estadística que resulte comprensible para prácticamente cualquier persona que utilice el modelo. De este modo, aunque se recurra posteriormente a modelos profundos como los Transformers, la codificación posicional introducida al inicio en los datos puede ser ajustable y fácilmente interpretable en función del contexto del problema.

Hasta ahora, gran parte de los esfuerzos se han centrado en adaptar soluciones desarrolladas originalmente para otros dominios, principalmente el procesamiento de lenguaje natural, al contexto de las series temporales. Sin embargo, estas adaptaciones no siempre se alinean completamente con las particularidades de los datos temporales, lo que puede derivar en pérdidas de información relevantes o en un rendimiento subóptimo. Esta desconexión pone de relieve la necesidad de desarrollar codificaciones específicamente diseñadas para capturar la estructura temporal propia de este tipo de datos. En este documento, abordaremos esta problemática desde una perspectiva crítica, comparando distintos métodos existentes y proponiendo nuevas soluciones, algunas de ellas híbridas, que buscan mejorar la representación temporal en modelos basados en Transformers.

## 1.3. Objetivos

Teniendo en cuenta los conceptos descritos anteriormente, y analizando el estado del arte en la actualidad, vemos justificada la necesidad de encontrar nuevas codificaciones posicionales con el objetivo de conseguir:

1. Realizar una revisión exhaustiva del estado del arte sobre positional encoding y su aplicación en modelos Transformer para series temporales, estudiando el funcionamiento de las diferentes técnicas y extrayendo el conocimiento útil para crear nuevas alternativas.

2. Analizar las propuestas originales de positional encoding adaptadas a series temporales, realizando un análisis crítico y explorando las diferentes vías alternativas.
3. Evaluar sistemáticamente la efectividad de los positional encoding estándar en tareas de forecasting sobre benchmarks de series temporales, haciendo uso de varios conjuntos de datos de diferente procedencia y estructura.
4. Establecer criterios claros para el diseño de nuevas codificaciones que permitan encontrar nuevas formas de incorporar información útil, pero además sirvan como guía para investigaciones futuras, identificando buenas prácticas, limitaciones comunes y condiciones bajo las cuales cada enfoque resulta más adecuado.
5. Comparar el impacto de las distintas codificaciones posicionales no solo en la precisión de las predicciones, sino también en aspectos como la capacidad de generalización y su eficiencia computacional.

En este documento, se recoge el estudio del estado del arte actual, realizando un estudio crítico y comparativo de las diferentes propuestas, haciendo también hincapié en el propio funcionamiento de los Transformer y como afecta a la codificación y procesado de la información. Se buscará, como ya se ha mencionado, crear un resumen con diferentes técnicas, nuevas y otras conocidas, que por separado o en su conjunto, de forma híbrida, propongan soluciones adecuadas a diferentes conjuntos de datos provenientes del mundo real.

Adicionalmente, se proporciona un repositorio en GitHub[1] que contiene todo el código desarrollado, así como las referencias a los conjuntos de datos utilizados. Este recurso se mantiene accesible con el objetivo de facilitar su análisis, reutilización y posible mejora en trabajos futuros.

[1] https://github.com/hexecoded/TFM

# 2. Tendencias y Estado del arte

Antes de adentrarnos de lleno en el análisis del problema y la realización de nuevas metodologías, es necesario estudiar y comprender el estado actual de esta temática en el estado del arte. Debido a que no se ha visto tan explotada como otros ámbitos de la IA, como es el caso de los modelos convolucionales o el aprendizaje automático supervisado tabular, podemos encontrar continuos cambios y nuevas vertientes que pueden inspirarnos a la hora de abordar el problema.

Hasta hace una década, buena parte de las técnicas diseñadas especialmente para series temporales y predicción a largo plazo, se basan principalmente en la descomposición las mismas en subcomponentes mucho más sencillas de procesar, las cuales pueden seguir un enfoque similar a divide y vencerás: descomponer la red en diferentes elementos, y mediante un mecanismo de agregación (aditivo, multiplicativo, o estadístico), recomponer la solución a la tarea. Sin embargo, se trata de una estrategia últimamente menos utilizada, en decadencia a favor de nuevas técnicas

Actualmente, una de las principales tendencias consiste en adaptar modelos originalmente desarrollados para otras modalidades. Un ejemplo de ello es el uso de convoluciones, ampliamente utilizadas en visión por computador, con el objetivo de capturar patrones locales en las secuencias y reducir la dimensionalidad del modelo. Previamente, también se han explorado arquitecturas recurrentes, como las Recurrent Neural Networks (RNN) y sus variantes más avanzadas, como las Long Short-Term Memory (LSTM) [6], aunque estas presentaban ciertas limitaciones en cuanto a capacidad de paralelización y en la modelización de relaciones temporales de mayor alcance, ya que para lograrlo aumentaba en exceso el tamaño de la red para incorporar más conexiones.

Más recientemente, ha ganado protagonismo la adaptación de mecanismos provenientes del procesamiento de lenguaje natural, en particular los Transformers, debido a su capacidad para modelar dependencias a largo plazo de forma eficiente. Además, comparten una motivación estructural con las series temporales: la necesidad de preservar una secuencia ordenada y coherente en la entrada, lo que permite aprovechar su arquitectura basada en atención para tareas de predicción.

A continuación, nos adentraremos de lleno en dichas propuestas, con el fin de aclarar, en primer, los conceptos clave acerca de las series temporales, y además, entender las problemáticas que surgen durante su procesado.

## 2.1. Procesamiento básico de Series Temporales

En la introducción, se han presentado brevemente las características fundamentales de las series temporales: su estructura temporal, caracterizada por muestreo, la importancia del orden de medición, y la necesidad de mantener las relaciones temporales para lograr aprender

de forma efectiva.

Pero, en esta sección, profundizaremos en los aspectos clave del preprocesamiento necesario para su análisis de manera más formal, ya que a diferencia de otros tipos de datos, las series temporales requieren una atención especial a la dimensión temporal, y cualquier alteración en su estructura puede afectar directamente la capacidad del modelo para aprender los patrones subyacentes en ella. Por ello, es fundamental centrarnos en cuestiones como la frecuencia de muestreo, la consistencia temporal, el tratamiento de valores faltantes, y la normalización de las variables.

### 2.1.1. Modelos basados en descomposición

Las series temporales pueden exhibir una gran cantidad de patrones y comportamientos, los cuales son interesantes de estudiar y visualizar con claridad para estudiar que posible enfoque seguir a la hora de resolver el problema. Diferenciando cada una de las partes, podemos tratar de resolver cada una de ellas por separado, y posteriormente, construir un modelo agregado capaz de solucionar nuestro problema.

Frecuentemente, podemos encontrar 3 comportamientos, los cuales son fácilmente identificables incluso gráficamente, que nos permiten obtener información bastante valiosa acerca del problema que estamos estudiando, y nos facilitan la decisión de escoger la técnica adecuada:

- **Tendencia**. Es el movimiento de los valores de serie a largo plazo, es decir, el comportamiento mostrado por los datos a la hora de estudiar su progresión desde el inicio de la serie hasta su final. El objetivo es encontrar si esta mantiene una dirección, de manera general, en su periodo de muestreo, buscando si sigue un comportamiento creciente, decreciente o constante, al igual que en el caso de estudio de monotonía clásico en funciones. Normalmente, se representan mediante comportamientos sencillos, y no se ven afectadas por grandes variaciones o el impacto de otras componentes de la serie, como variables aleatorias u otros factores externos. Normalmente, podemos modelarla haciendo uso de funciones elementales, como ecuaciones lineales o funciones exponenciales, logarítmicas o polinomiales.

  A veces, se requieren procesos de suavizado para así poder eliminar el efecto de otras componentes ruidosas que rodean la tendencia real de la serie. Esto se puede conseguir, de manera similar a la convolución, desplazando una ventana a lo largo de la serie, de manera que los valores dentro de ella sean suavizado. La forma más habitual de lograrlo es con una media móvil, la cual, cuanto mayor amplitud tenga, mayor reducción de ruido conseguiremos.

  En la figura 2.1, podemos ver un ejemplo de su cálculo representado gráficamente sobre el dataset Air Passengers [4].

- **Estacionalidad**. Son fluctuaciones periódicas y predecibles dentro de la serie temporal, las cuales siguen una determinada frecuencia y que pueden ser fácilmente modelables una vez se observa al menos un periodo. Normalmente, coinciden con unidades de medida de calendario, pudiendo encontrar así frecuecias semanales, mensuales, trimestrales, anuales... etc. Normalmente, son un factor visualmente sencillo de identi-

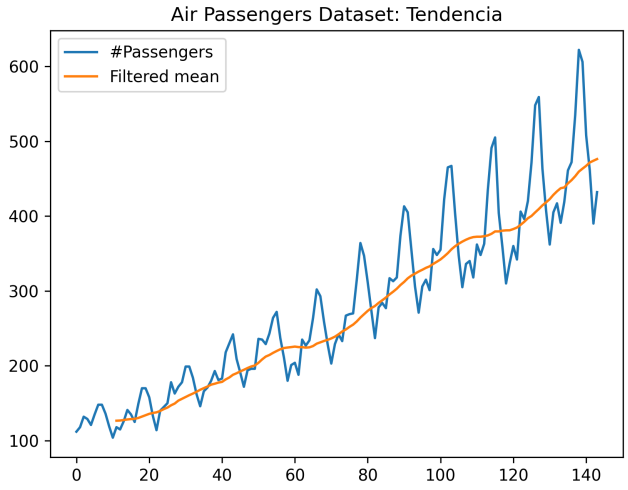


Figura 2.1.: Estudio de la tendencia en el dataset Air Passengers

ficar, el cual apenas varía ente períodos, y que nos permite obtener información muy valiosa para el modelado.

Podemos encontrar este comportamiento, por ejemplo, en los desplazamientos realizados durante las épocas vacionales: podremos observar un patrón de crecimiento en estas en Navidad y las vacaciones de verano, principalmente julio y agosto; y en el resto de meses el comportamiento será a la baja. Y eso ocurrirá con un período anual, ya que dichas fechas se ubican siempre en el mismo lugar. Diferente caso sería con Semana Santa, ya que su fecha no es coincidente todos los años y no cumple al 100 % de estacionalidad como las otras dos, ya que aunque es acotable en calendario, no es exactamente coincidente anualmente.

En el caso del dataset anterior, se puede observar esta estacioalidad claramente (figura 2.2)

- **Ciclo**. Son también fluctuaciones de la serie temporal, pero, a diferencia de ser regulares, siguen un período irregular de logitud fija. Normalmente, está vinculado a eventos que no están especialmente vinculados a fenómenos de calendario. El ejemplo más representativo podría ser la tendencia de crecimiento económico de un país, o el comportamiento de las acciones en bolsa de una sociedad anónima.

  En la figura 2.3 podemos un ejemplo de este comportamiento con los permisos concedidos para la construcción de viviendas en EEUU [8], donde no podemos encontrar patrones claros como en Air Passengers.

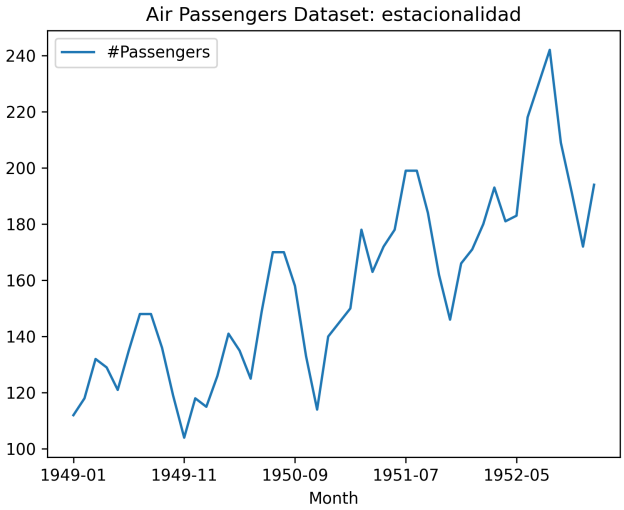


Figura 2.2.: Estudio de la estacionalidad en el dataset Air Passengers

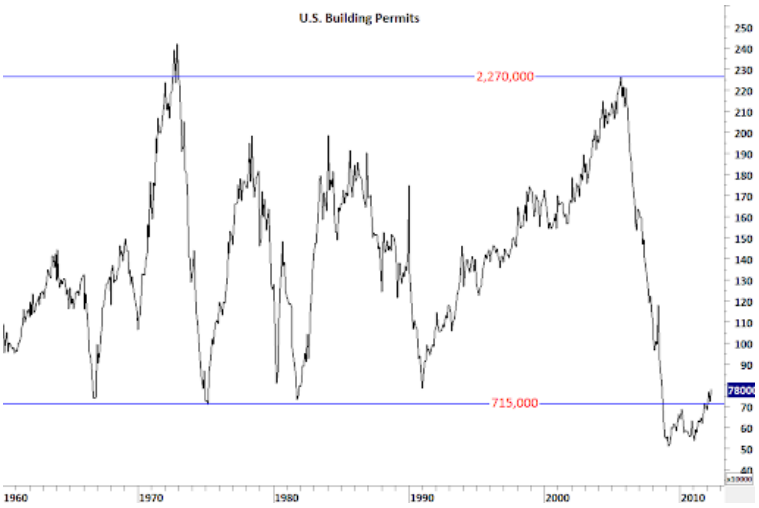


Figura 2.3.: Estudio de ciclos en USA building permits

Una vez comprendidos estos conceptos, podremos utilizar su información para decidir qué modelo se adapta mejor a nuestro problema y además tratar de modelar la tendencia y la estacionalidad. Anteriormente, mencionamos la gran utilidad de modelos como ARIMA, que funciona bajo este concepto. Pero no se trata del único modelo existente bajo este paradigma, sino uno de los más utilizados. En función del mecanismo de agregación de la solución, podemos clasificar las técnicas en dos grupos:

- **Descomposición aditiva**. Realiza una separación de la serie temporal varias componentes, las cuales son modeladas por separado y sumadas entre si para dar lugar a

la función predicha final (ecuación 2.1.1). Es la más sencilla de usar en casos simples. Es indicada cuando las fluctuaciones estacionales por las variaciones entorno a la tendencia no varían con el valor de la serie temporal, es decir, pueden prácticamente mantenerse entre a lo largo de toda la serie sin apenas cambios. En general, se usa cuando los elementos no depende los unos de otros.

$$y(t) = T(t) + S(t) + C(t) + E(t)$$

- **Descomposición multiplicativa**. Se utiliza esta alternativa cuando las diferentes componentes dependen en nivel general de la serie (ecuación 2.1.1). Es el caso de las series en las cuales los aumentos en la tendencia provocan aumentos también los picos de las tendencias estacionales o los ciclos. Por ejemplo, en el dataset de Air Passengers, el comportamiento que se da es de este tipo.

$$y(t) = T(t) \times S(t) \times C(t) \times E(t)$$

Donde, en ambos casos:

- *T(t)*: Componente de la tendencia
- *S(t)*: Componente estacional
- *C(t)*: Componente cíclica
- *E(t)*: Componente aleatoria y error

La elección de cada método depende, por tanto, de los datos, y nos beneficiaremos, como ya hemos visto a lo largo de la definición, de una adecuada visualización de los mismos cuando sea posible.

A continuación, definiremos en mayor detalle 3 de los algoritmos basados en descomposición más usados: STL [5], ARIMA [3] y Prophet [10].

#### 2.1.1.1. STL

Esta técnica, llamada *Seasonal Time-Series Decomposition*, nos permite descomponer la serie en partes más sencillas de modelar, separando la serie en tres componentes básicas: tendencia, estacionalidad y resto, siguiendo así el esquema visto anteriormente. De esta forma, podemos comprender en mayor detalle cómo evolucionan los valores a lo largo del tiempo, y comprender mejor cómo modelar su comportamiento. Para obtener un modelo para cada una de ellas, normalmente se sigue un proceso de 3 pasos que nos permite aislar la información.

1. **Extracción de la tendencia**. Se comienza extrayendo la tendencia de la serie para así obtener el comportamiento subyacente de la misma, y saber si es creciente, decreciente o se mantiene constante. En este algoritmo, se consigue mediante un proceso de suavizado, llamado *Locally Estimated Scatterplot Smoothing* (Loess) de manera iterativa. Para ello, primero se comienza aplicando Loess, de manera que se suavizan los valores realizando una regresión no parámetrica, donde se observan los valores cercanos a cada timestamp de manera que se realiza un promedio que reduce las diferencias en el entorno. Si la serie es muy compleja o ruidosa, es posible que sea necesario repetir

este proceso varias veces, por lo que se convierte en un proceso iterativo, utilizando tamaños mayores de ventana entorno a cada valor.

Dicha ventana se estima a través de pesos, los cuales se ven reducidos cuanto más alejados al valor están. Estos se rigen por el valor de amplitud $h$, que recogen las ecuaciones clásicas de la regresión de Loess. Dado un conjunto de datos $(t_i, y_i)$, el valor suavizado en $t_0$ se calcula como:

$$\hat{y}(t_0) = \mathbf{x}_0^\top \hat{\boldsymbol{\beta}}(t_0)$$

donde $\hat{\boldsymbol{\beta}}(t_0)$ se obtiene minimizando la suma ponderada de errores:

$$\hat{\boldsymbol{\beta}}(t_0) = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} w_i(t_0) \left(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}\right)^2$$

Los pesos $w_i(t_0)$ dependen de la distancia temporal y se definen, habitualmente, con el kernel tricúbico:

$$w_i(t_0) = \left(1 - \left|\frac{t_i - t_0}{h}\right|^3\right)^3, \quad \text{para} \quad |t_i - t_0| < h$$

El valor arrojado será 0 fuera del intervalo, y un valor entre 0 y 1 dentro de él, seleccionando así la relevancia de cada punto respecto a sus valores del entorno.

2. **Extracción de la estacionalidad**. Para extraer la estacionalidad, debemos aislarla de la tedencia que modelamos en el paso anterior. Para ello, basta con restar dicha componente a la serie original, y así disponer ahora únicamente del los patrones estacionales y el ruido.
$$y'(t) = y(t) - trend(t)$$

   A partir de aquí, debemos dividir la serie sin tendencia en subseries estacionales, agrupando los datos según la posición que ocupan dentro del ciclo; por ejemplo, si identificamos patrones con frecuencia mensual, se agruparían todos los valores correspondientes al mismo mes en años distintos, y sobre cada una de estas subseries se aplica un suavizado Loess de forma individual. Así, podremos capturar con precisión el patrón que se repite en cada estación, de manera similar a un promedio. Gracias a este proceso, podremos reconstruir una componente estacional coherente con las variaciones periódicas observadas en nuestro conjunto de datos, al que denominaremos *s(t)*.

3. **Obtención del resto**. Una vez extraídas tanto la tendencia como la estacionalidad, la componente residual se obtiene simplemente como la diferencia entre la serie original y la suma de las dos componentes anteriores:

$$r(t) = y(t) - trend(t) - s(t)$$

   Esta parte representa la variabilidad no explicada por los patrones sistemáticos de largo plazo ni por los ciclos periódicos, e incluye tanto el ruido aleatorio como cualquier información no capturada por el modelo. Si bien no es modelable, es muy útil para

identificar si la serie es correctamente modelable por este método, ya que si observamos demasiada información en esta componente, nos está indicando que una parte importante de la información no está siendo recogida. Lo deseable es la que la varianza descrita sea la mínima posible. Sin embargo, hay algunos casos en los que encontrar puntos extremos es de utilidad, y se trata de la búsqueda de valores anómalos. Si la serie restante permanece estable y con poca varianza, a excepción de ciertos puntos, puede ser de interés estudiar qué ocurrió en dichos instantes, ya que su comportamiento se sale del comportamiento habitual modelado.

En su publicación original, podemos encontrar un ejemplo de ejecución sobre el conjunto de datos *Daily Carbon Dioxide Data* (ver figura 2.4), que recoge información sobre las emisiones de $CO_2$ desde el 17 de abril de 1974 hasta el 31 de diciembre de 1986, y permite observar la información recogida por cada una de las componentes.

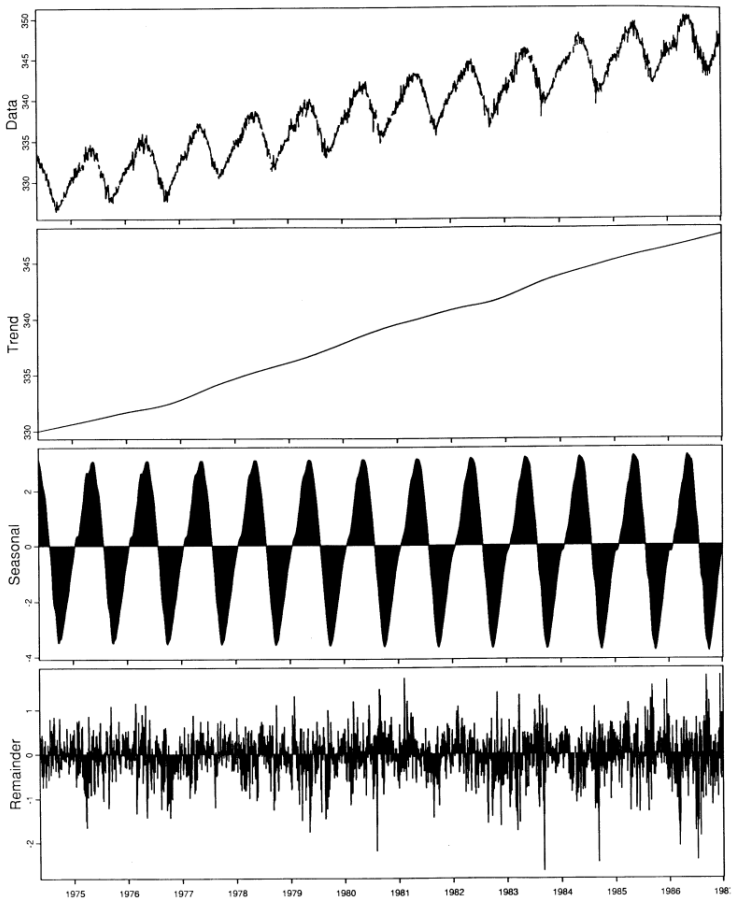


Figura 2.4.: Ejemplo de caso de estudio empleando STL: Daily Carbon Dioxide Data [5]

En este caso, podemos apreciar visualmente una serie fácilmente modelable por este enfoque, ya que la curva describe una tendencia creciente bastante clara, y los periodos estacionales están bastante bien delimitados y no se ven influenciados por el crecimiento recogido en la tendencia. Es un ejemplo perfecto para el uso de los modelos basados en descomposi-

ción aditiva.

Sin embargo, esta forma de modelado presenta varias limitaciones, que acotan su funcionamiento a problemas simples:

- Sólo se limita a descomponer series de naturaleza aditiva, por lo que aquellas en las que las diferentes componentes se vean influenciados entre sí requerirán técnicas propias de modelos multiplicativos.
- Pueden perderse gran cantidad de detalles explicados por los datos, debido a la configuración del parámetro de suavizado. Si alisamos de más la serie, podemos estar perdiendo información clave acerca de nuestro problema, dejando completamente de lado ciertos comportamientos sutiles, lo cual aumenta la tasa de error. Debemos escoger adecuadamente los valores al aplicar Loess.
- Es costoso computacionalmente en caso de series de larga duración, ya que el cálculo de aplicar el kernel se hace para cada punto en cada iteración sobre el conjunto de datos, tanto para encontrar la tendencia como para luego generalizar la estacionalidad.

Pero, sin lugar a dudas, la principal desventaja de este método es que no consigue por sí solo lo que deseamos resolver en este trabajo: predecir, sobre todo, a largo plazo. STL es un procedimiento muy útil que nos permite entender el comportamiento de los datos, y descomponerlo para comprender el funcionamiento de cada componente, pero no es posible por sí solo realizar predicciones. Dependemos del apoyo de otros procedimientos que nos permitan estimar , basándonos en este historial, la progresión de las componentes extraídas como la tendencia (modelable por regresión), y la estacional, la cual podría ser evaluada de manera simplificada por su valor medio histórico por intervalo.

Esto lo hace en un método interesante desde el punto de vista de análisis, y que permite dar lugar a modelos que sí son capaces de estimar predicciones futuras, pero no es una herramienta por sí sola que nos permita resolver la tarea que nos concierne en este trabajo.

#### 2.1.1.2. ARIMA

Basándonos de nuevo en el método de descomposición de series, podemos encontrar el modelo **ARIMA** (*AutoRegressive Integrated Moving Average*) una técnica estadística ampliamente utilizada para el modelado de series temporales univariantes. Como ya adelantamos en la introducción del proyecto, su estructura se basa en la combinación de tres componentes principales: una parte autoregresiva (AR), una parte integrada (I) y una parte de media móvil (MA). Cada una de estas partes permite capturar diferentes aspectos del comportamiento temporal de los datos.

##### Componente de Integración

Por integración, entendemos el número de veces que es necesario integrar la serie temporal original para que esta se vuelva estacionaria. De manera resumida, una serie estacionaria es aquella cuyas propiedades estadísticas, como la media, la varianza y la autocorrelación entre valores, se mantienen constantes a lo largo del tiempo, como ya vimos en la definición de los modelos aditivos.

La diferenciación nos permite eliminar tendencias o patrones sistemáticos de crecimiento o decrecimiento, facilitando el modelado de las componentes autoregresivas (AR) y de media móvil (MA). A pesar de que su nombre nos pueda insinuar un proceso complejo de integración analítica, al realizarse sobre valores numéricos, no es más que una diferencia de valores:

$$y'_t = y_t - y_{t-1}$$

Este proceso puede aplicarse tantas veces como sea necesario para lograr estacionariedad. Dicho número de repeticiones viene controlado por el valor de orden $d$, que indica el número de veces que se ha diferenciado la serie. En general, el modelo ARIMA supone que tras aplicar $d$ diferencias, la serie resultante $y^{(d)}$ puede ser modelada por un proceso estacionario ARMA$(p, q)$. Por tanto, el componente $I$ de ARIMA es un procesamiento indispensable de la serie cuando no se cumpla el requisito de estacionariedad, y permitirá calcular sobre la serie resultante las componentes AR y MA con mayor facilidad.

Para saber cuando deja de ser nececesario aplicar el proceso iterativo de integración, podemos hacer uso de dos posibles mecanismos: uno gráfico, basado en el gráfico de autocorrelación de valores, y otro estadístico, basado en el test de Dickey-Fuller aumentado (ADF).

- El análisis gráfico consiste en observar el comportamiento de la función de autocorrelación (ACF) tras aplicar una o varias diferenciaciones. Si la serie es no estacionaria, las valores siguientes de autocorrelación decaen lentamente, indicando que queda información relevante en dichas componentes. Por el contrario, si la serie es estacionaria, la ACF suele caer rápidamente a cero, estableciendo que no es necesario continuar diferenciando. La autocorrelación en un retardo $k$ equivale a:

  $$\rho_k = \frac{\text{Cov}(y_t, y_{t-k})}{\sigma^2}$$

  donde $\sigma^2$ es la varianza de la serie y $\text{Cov}(y_t, y_{t-k})$ es la covarianza entre los valores actuales y los desplazados $k$ periodos atrás.

- El test ADF, por su parte, nos proporciona la misma conclusión de manera numérica, para lograr un mayor rigor estadístico en la solución. Su funcionamiento radica en la búsqueda de una raíz unitaria en la serie, en cuyo caso significa que la serie presenta comportamiento no estacionario.

  De forma general, el test se basa en ajustar una regresión del tipo:

  $$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta y_{t-i} + \epsilon_t$$

  donde $\Delta y_t = y_t - y_{t-1}$ representa la primera diferencia de la serie, $\gamma$ es el parámetro clave para detectar la raíz unitaria, y $\epsilon_t$ es el error.

  Sin embargo, lo más relevante para su uso es comprender cómo funciona el test de hipótesis. Como hipótesis nula ($H_0$), define que la serie no es estacionaria, es decir, que presenta una raíz unitaria, y como ($H_1$) podremos aceptar que la serie sea estacionaria. Para ello, basta con observar el valor de p:

- Si el **p-valor** del test es significativo se puede rechazar la hipótesis nula. En ese caso, la serie puede considerarse estacionaria, y podemos parar el proceso iterativo de diferenciación.
- Si el **p-valor** es mayor al valor de significación, no podemos rechazar la hipótesis nula, lo que indica que la serie sigue siendo no estacionaria, y debemos continuar el proceso de integración.

#### Componente autoregresiva (AR)

Esta parte del modelo representa la relación entre el valor actual de la serie y sus valores pasados. Supone que el valor presente puede explicarse como una combinación lineal de observaciones anteriores, los cuales se denominan retardos o lags. Normalmente, el número de lags a emplear es un hiperparámetro clave del modelo, el cual es controlado por $p$.

Matemáticamente, podemos expresarlo como:

$$z_t = w_1 y_{t-1} + w_2 y_{t-2} + \cdots + wi_p y_{t-p} + \varepsilon_t$$

donde:

- $w_1, w_2, \ldots, w_p$ son los coeficientes autoregresivos que indican el peso o importancia de cada valor pasado. Cuando usamos el valor p, establecemos el número de valores con peso 1 que tomará nuestro modelo, mientras que el resto se mantienen a 0. Pero, podríamos tener variantes en las que dichos pesos fueran valores flotantes entre 0 y 1 que permitan una ponderación más refinada de los valores.
- $\varepsilon_t$ es el término asociado al error en el instante $t$.

Gracias a la componente AR, podemos capturar relaciones de corto plazo y patrones de dependencia temporal, siempre que la serie sea estacionaria. Es habitual que muchas series temporales muestren este tipo de estructura, donde los valores pasados son predictivos del comportamiento futuro inmediato, como ya vimos con el caso de las temperaturas y el consumo eléctrico. Por tanto, la elección del parámetro $p$ es clave: un valor pequeño implica que sólo los valores más recientes tengan influencia, mientras que un valor mayor permite capturar relaciones más amplias en el tiempo, pero puede aumentar el riesgo de sobreajuste y se pierda localidad. En definitiva, se trata de un elemento esencial del rendimiento de ARIMA, por lo que es clave elegir adecuadamente su valor.

#### Componente de media móvil (MA)

Por último, nos queda describir el funcionamiento de la media móvil. Esta tiene como objetivo modelar la parte estocástica de la serie temporal, es decir, los errores de predicción cometidos en instantes anteriores. Anteriormente, vimos que la parte autoregresiva se basaba en realizar la media sobre sus mismos valores, mientras que aquí tratamos de extraer información de los residuos.

Formalmente, un modelo MA de orden $q$ se define como:

$$y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

donde:

- $y_t$ es el valor observado de la serie en el instante $t$,

- $\mu$ es la media de la serie (si está centrada, puede ser cero),
- $\varepsilon_t$ es el término de error en el instante $t$,
- $\theta_1, \ldots, \theta_q$ son los coeficientes del modelo MA.

Esto nos permitirá capturar patrones aleatorios que no pueden explicarse mediante una simple tendencia o correlación temporal, y por tanto, mejorando los resultados y el rendimiento del modelo, al suavizarse los errores pasados para acercanos más a la serie real. Por tanto, el número de errores a considerar $q$ debe escoger de manera adecuada. Para facilitar esta decisión, podemos basarnos de nuevo en dos enfoques: en emplear la información del gráfico ACF, o bien, de manera experimental:

- El análisis gráfico mediante ACF consiste en observar el gráfico de la función de autocorrelación (ACF) de la serie diferenciada, y ver cómo se comportan los lags. En un modelo MA puro, los primeros $q$ retardos (*lags*) suelen mostrar autocorrelación significativa, pero luego suelen decrecer de manera repentina a cero. Una buena estimación puede ser tomar como valor de $q$ el último lag con autocorrelación significativa antes del corte. Es decir, si a partir del punto q+1 se produce un descenso abrupto, tomamos $q$.
- Si la gráfica no nos proporciona una respuesta clara, y la serie es modelable en un tiempo razonable, se puede realizar una búsqueda hiperparámetros, probando diferentes valores de $q$, y entrenando un modelo ARIMA para cada combinación posible (junto a valores de $p$ y $d$), y evaluar su rendimiento utilizando métricas como el *Mean Squared Error* (MSE). Para facilitar el proceso, podemos usar técnicas clásicas como grid search, que no es más que realizar todas las combinaciones posibles de parámetros p,q y d. Pero, de manera más avanzada, existen adaptaciones de ARIMA en frameworks como auto_arima, que usan una estimación automática de los hiperparámetros.

#### 2.1.1.3. Prophet

### 2.1.2. RNN y LTSM

### 2.1.3. Transformers

## 2.2. Positional Encoding en Transformers

## 2.3. Conjuntos de datos disponibles

# 3. Selección y preprocesado de los conjuntos de datos

# 4. Modelos de encoding posicional y entorno de trabajo

# 5. Análisis comparativo de positional encoding sobre las bases de datos

# 6. Conclusiones y trabajos futuros

# A. Ejemplo de apéndice

Los apéndices son opcionales.

Este fichero `apendice-ejemplo.tex` es una plantilla para añadir apéndices al TFG. Para ello, es necesario:

- Crear una copia de este fichero `apendice-ejemplo.tex` en la carpeta `apendices` con un nombre apropiado (p.e. `apendice01.tex`).
- Añadir el comando `\input{apendices/apendice01}` en el fichero principal `tfm.tex` donde queremos que aparezca dicho apéndice (debe de ser después del comando `\appendix`).

# Glosario

La inclusión de un glosario es opcional.
Archivo: `glosario.tex`

$\mathbb{R}$ Conjunto de números reales.

$\mathbb{C}$ Conjunto de números complejos.

$\mathbb{Z}$ Conjunto de números enteros.

# Bibliografía


[1] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. Ré, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang. On the opportunities and risks of foundation models, 2022.

[2] G. E. P. Box and G. Jenkins. *Time Series Analysis, Forecasting and Control*. Holden-Day, Inc., USA, 1990.

[3] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, 1970.

[4] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976. AirPassengers dataset.

[5] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6(1):3–73, 1990.

[6] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[7] H. Irani and V. Metsis. Positional encoding in transformer-based time series models: A survey, 2025.

[8] A. Kibar. U.s. housing starts. u.s. building permits. https://blog.techcharts.net/index.php/2012/06/19/u-s-housing-starts-u-s-building-permits/, 2012. Consultado el 25 de julio de 2025.

[9] N. Kitaev, Łukasz Kaiser, and A. Levskaya. Reformer: The efficient transformer, 2020.

[10] S. J. Taylor and B. Letham. Forecasting at scale. *The American Statistician*, 72(1):37–45, 2018.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[12] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting, 2021.