

# **Predizione e Correzione Temporale/spaziale del Traffico Urbano da Dati Reali**

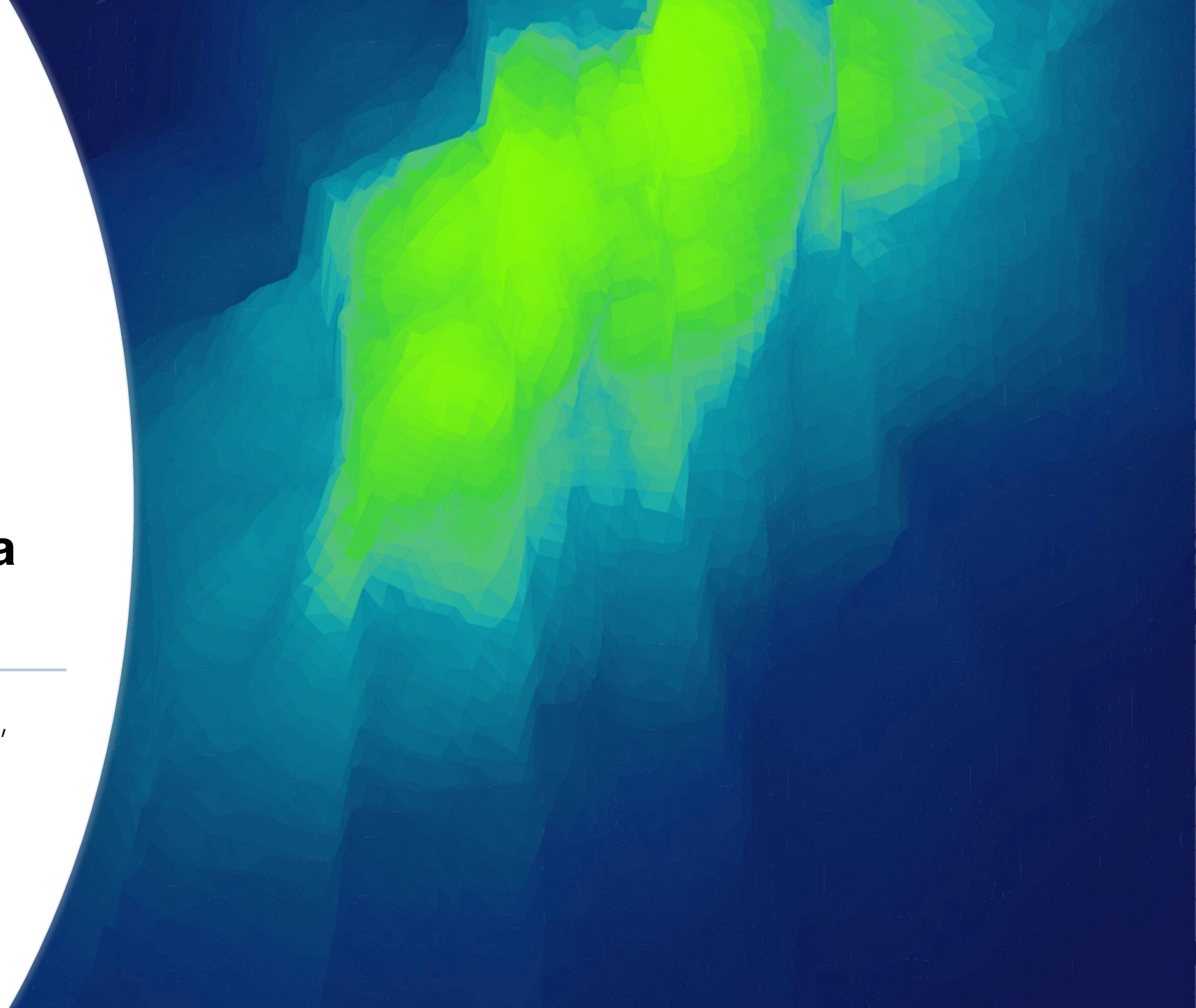
---

Autori: Agostino Vallefuooco,  
Giuseppe Di Lorenzo

Relatore: Francesco A. N. Palmieri

Correlatore: Giovanni Di Gennaro

Correlatore: Amedeo Buonanno



# Introduzione



## Contesto

- Aumento demografico / Traffico dati
- Errori dei sensori

## Obiettivi

- Predizione accurata del traffico stradale
- Accurato riempimento di mancanze nel dataset

# Formalizzazione

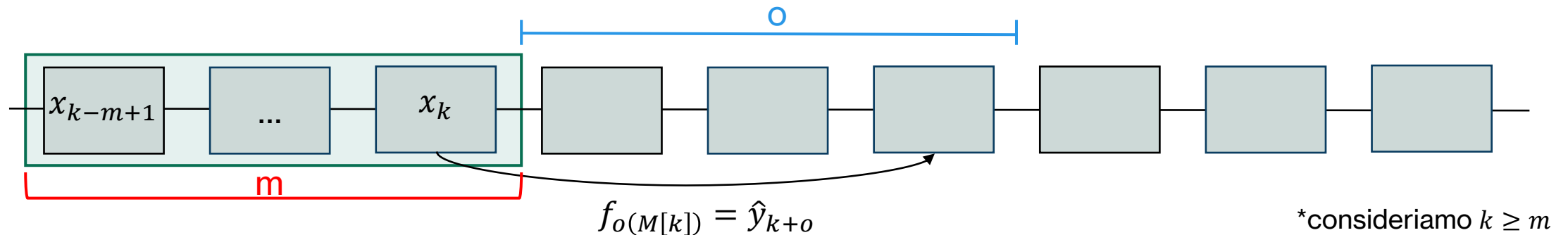
La previsione dei flussi di traffico è il processo di previsione delle condizioni di traffico per un periodo futuro sulla base delle informazioni storiche sul traffico.

Considerando i dati a disposizione ordinati come una tabella:

$$Q = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1f} \\ q_{21} & q_{22} & \cdots & q_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ q_{r1} & q_{r2} & \cdots & q_{rf} \end{bmatrix} = \begin{bmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_r^\top \end{bmatrix}$$

- Finestra di Look-Back:  $M[k] = [x_{k-m+1}, x_{k-m+2}, \dots, x_k]^\top$

- L'obiettivo è:  $\hat{y}_{k+o} = f_o(M[k])$

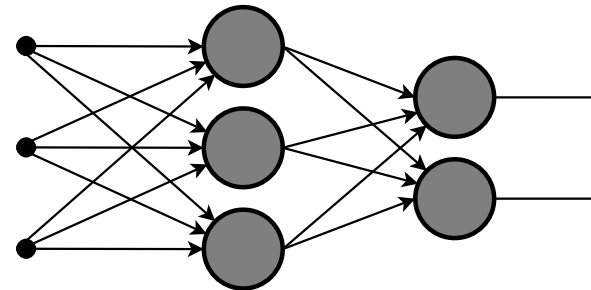


# Deep Learning

DL insieme di tecniche inerenti alle ANN organizzate in diversi strati, dove l'informazione viene elaborata in maniera sempre più completa

Reti utilizzate:

- Fully connected



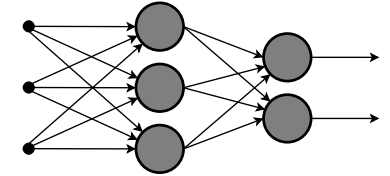
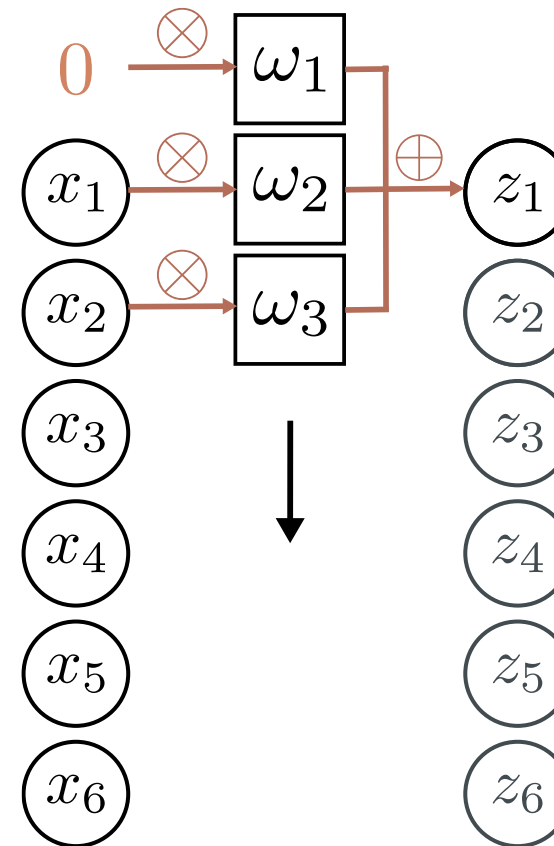
$$\varphi_L(W_L \varphi_{L-1}(\cdots \varphi_1(W_1 x + b_1) \cdots) + b_L)$$

# Deep Learning

DL insieme di tecniche inerenti alle ANN organizzate in diversi strati, dove l'informazione viene elaborata in maniera sempre più completa

Reti utilizzate:

- Fully connected
- Convolutional neural network

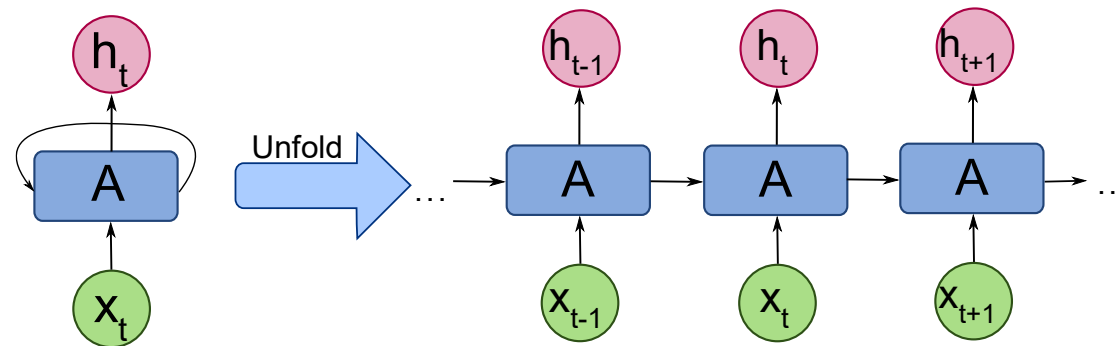
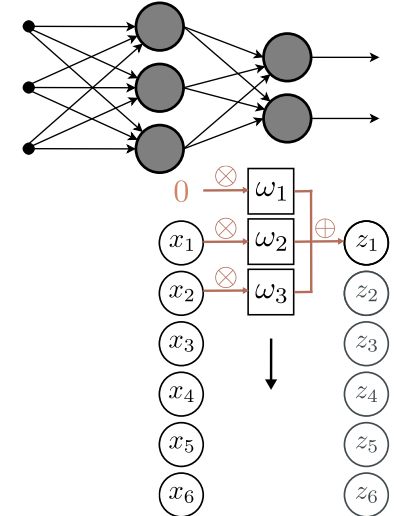


# Deep Learning

DL insieme di tecniche inerenti alle ANN organizzate in diversi strati, dove l'informazione viene elaborata in maniera sempre più completa

Reti utilizzate:

- Fully connected
- Convolutional neural network
- Recurrent Neural Network:



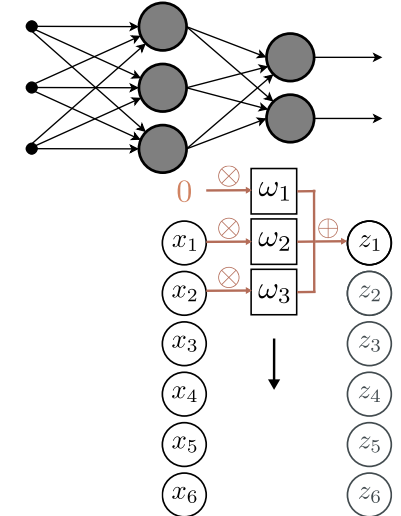
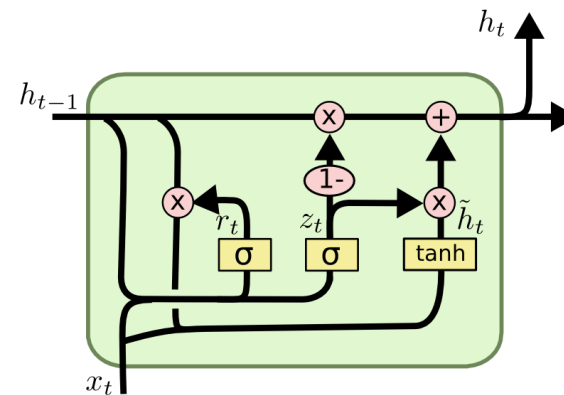


# Deep Learning

DL insieme di tecniche inerenti alle ANN organizzate in diversi strati, dove l'informazione viene elaborata in maniera sempre più completa

Reti utilizzate:

- Fully connected
- Convolutional neural network
- Recurrent Neural Network:
  - Gated Recurrent unit



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

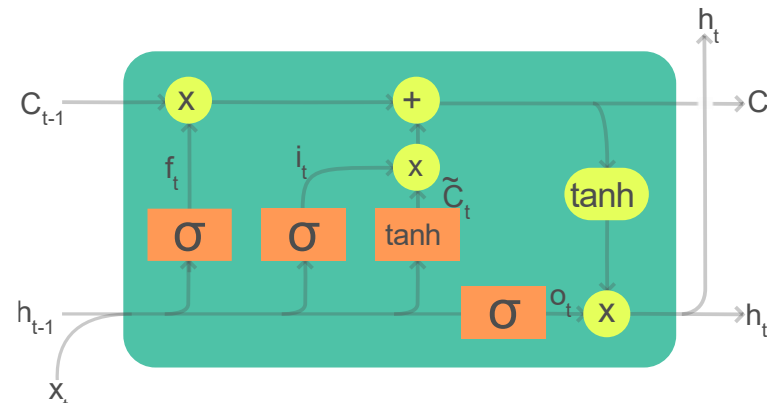
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

# Deep Learning

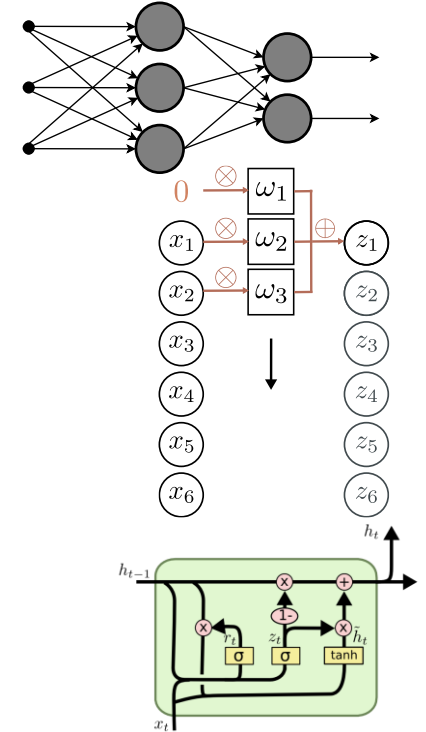
DL insieme di tecniche inerenti alle ANN organizzate in diversi strati, dove l'informazione viene elaborata in maniera sempre più completa

Reti utilizzate:

- Fully connected
- Convolutional neural network
- Recurrent Neural Network:
  - Gated Recurrent unit
  - Long Short-Term Memory



$$\begin{aligned}f_t &= \sigma(W_f[h_{t-1}, x_t] + b_f) \\i_t &= \sigma(W_i[h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_c[h_{t-1}, x_t] + b_c) \\C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\o_t &= \sigma(W_o[h_{t-1}, x_t] + b_o) \\h_t &= o_t \odot \tanh(C_t)\end{aligned}$$





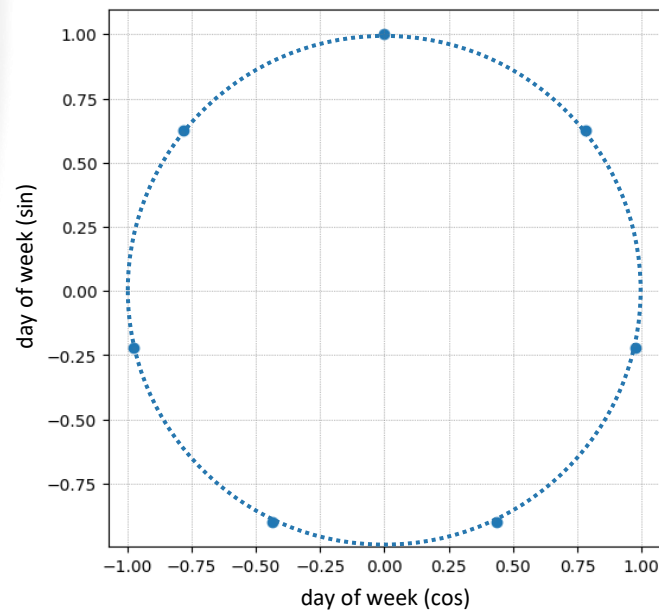
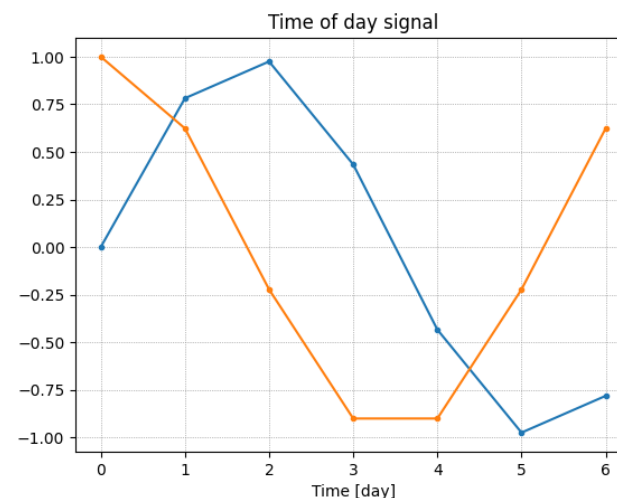
**V** ● Università  
● degli Studi  
della Campania  
*Luigi Vanvitelli*



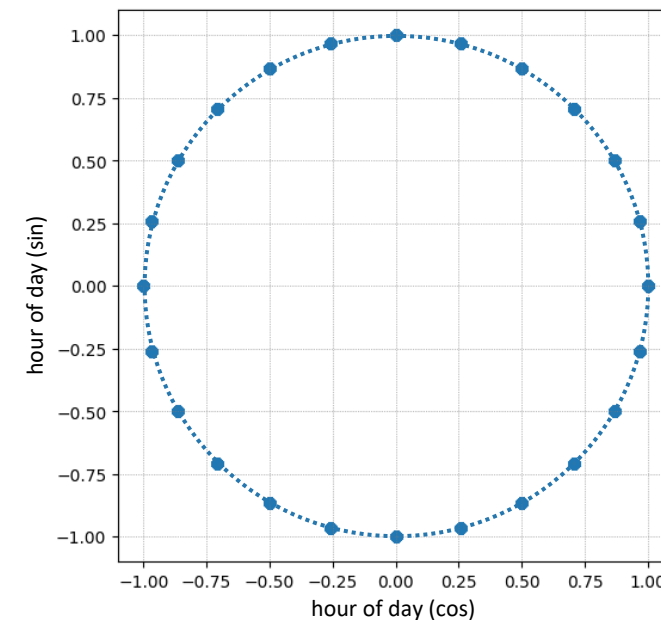
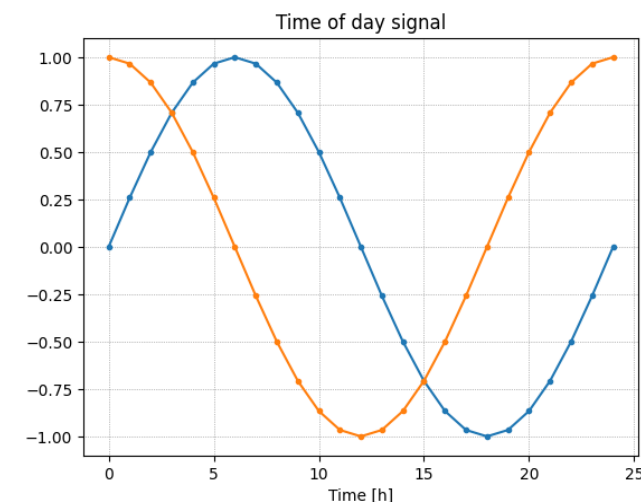
# Preparazione dei Dati

- Codifica ciclica
- Normalizzazione e divisione del dataset

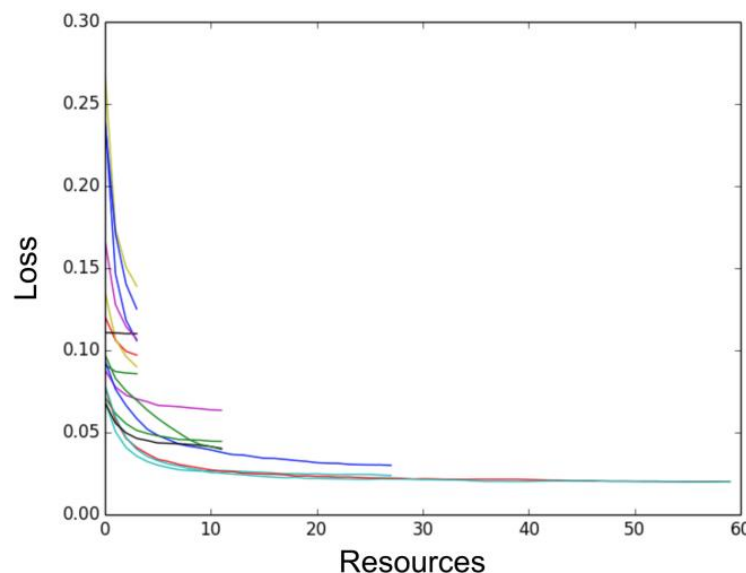
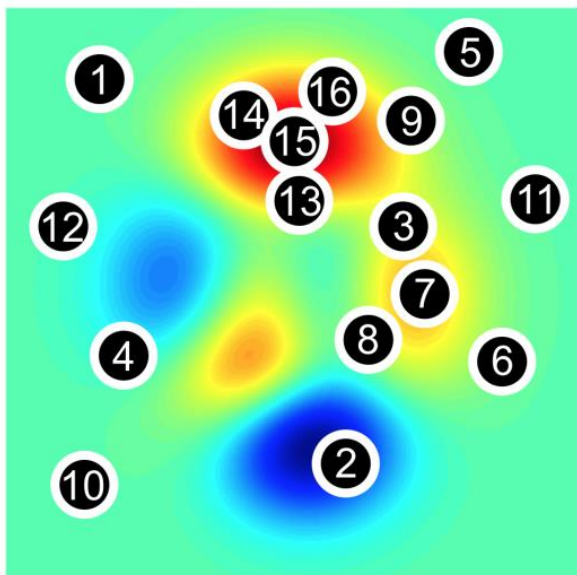
Giorno della settimana



Ora della giorno



# Algoritmo di tuning : Hyperband

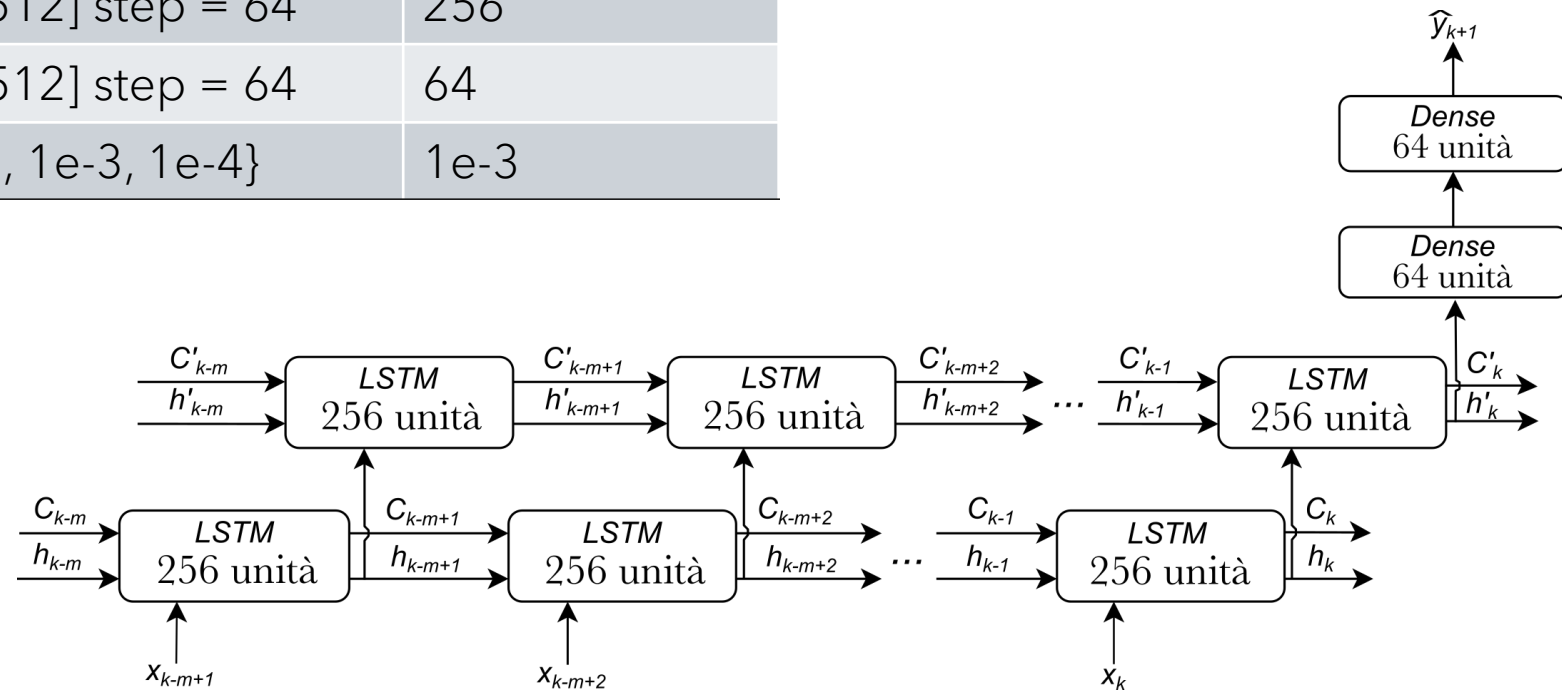


1. Definizione budget e spazio esplorazione
2. Generazione casuale delle configurazioni
3. Allocazione del budget ad ogni configurazione, proporzionale al numero di configurazioni rimanenti
4. Allenamento fino ad esaurimento del budget e valutazione
5. Scartate le configurazioni meno prestanti
6. Ripetere dei passi 3-5 fino ad avere una configurazione

# Hyperband: configurazioni e risultati

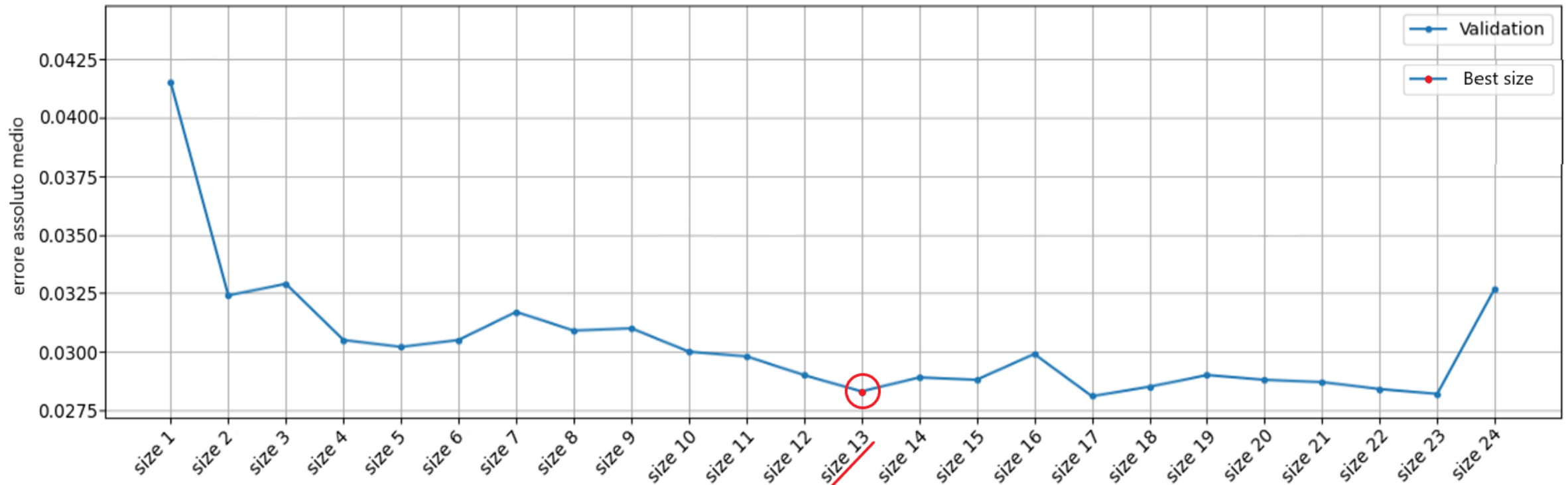
Hyperparameter	configuration	result
LSTM layer	{1, 2}	2
LSTM 1° layer unit	[64, 512] step = 64	256
LSTM2 2° layer unit	[64, 512] step = 64	256
Dense unit	[64, 512] step = 64	64
Learning rate	{1e-2, 1e-3, 1e-4}	1e-3

## Hyperparameter Tuning con Hyperband



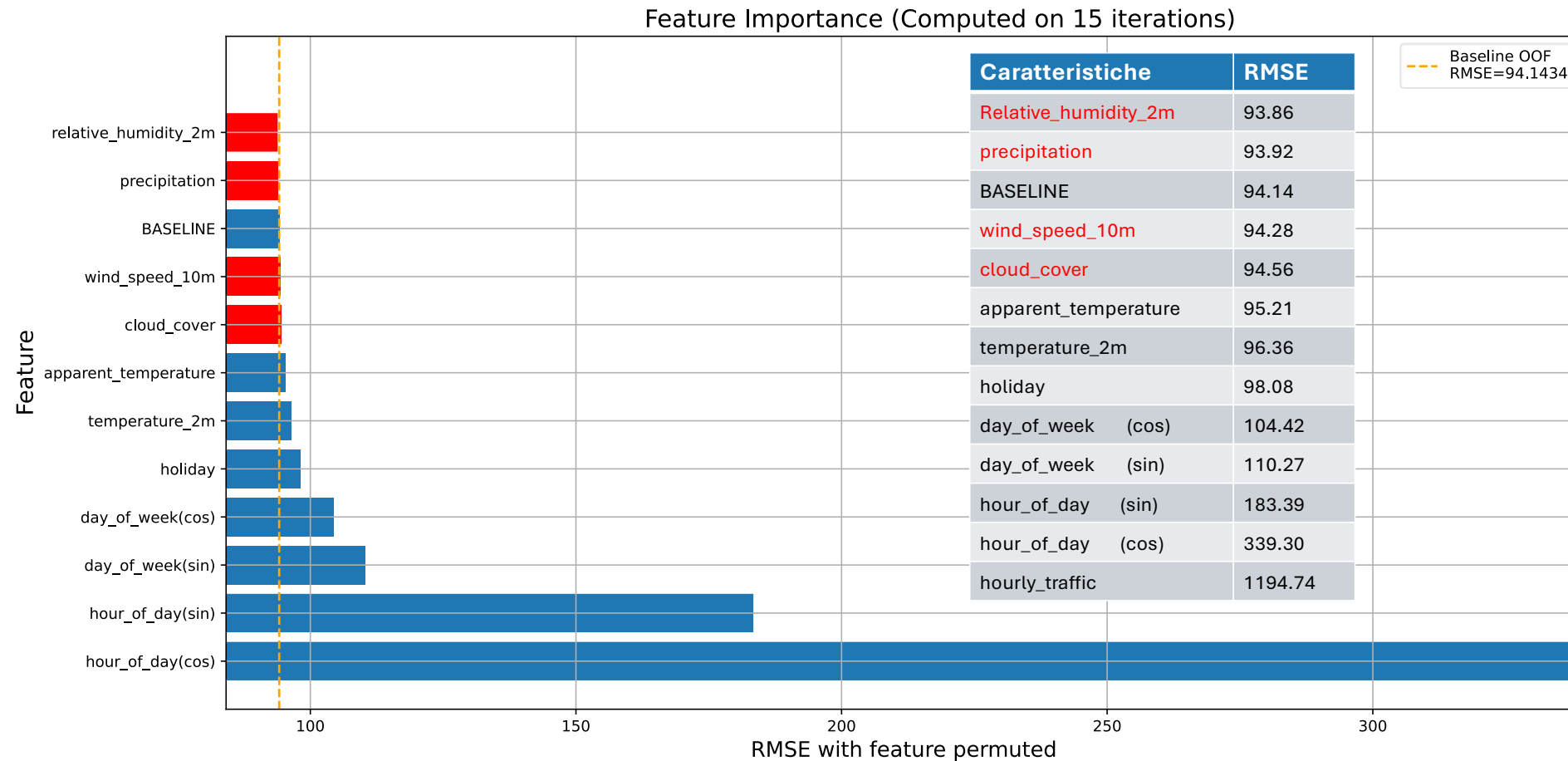
# Regolazione ingresso: dimensione finestra

- Modello addestrato su diverse lunghezze della finestra
- Per ognuna valutata il MAE calcolato su 5 iterazioni



# Regolazione ingresso: Permutation feature importance

- Metrica adottata RMSE
- Permutata ogni feature 10 volte
- Punteggio ottenuto come media di 15 iterazioni



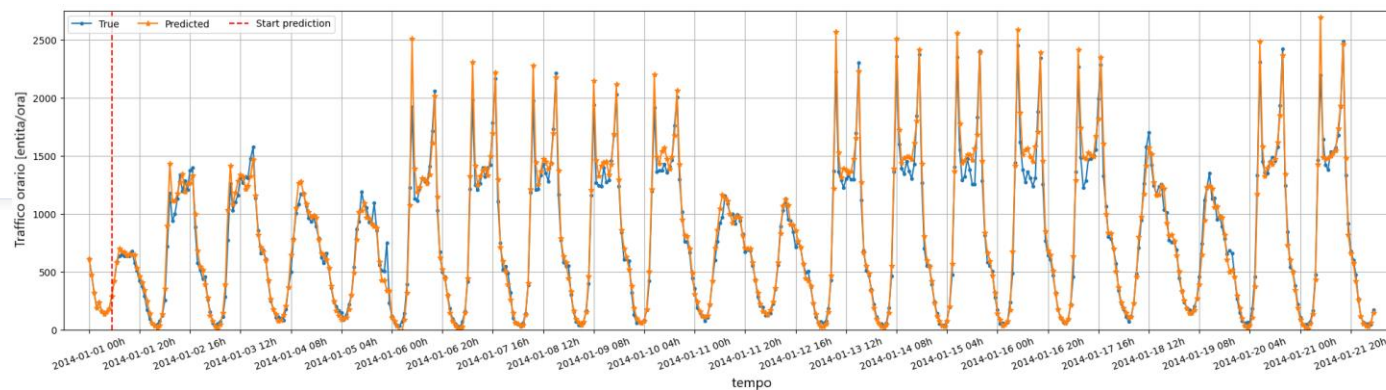
# Regolazione ingresso

modello pre regolazione ingresso  
Vs  
modello post regolazione ingresso

Modello pre regolazione ingresso:

Test MAE = 63.30

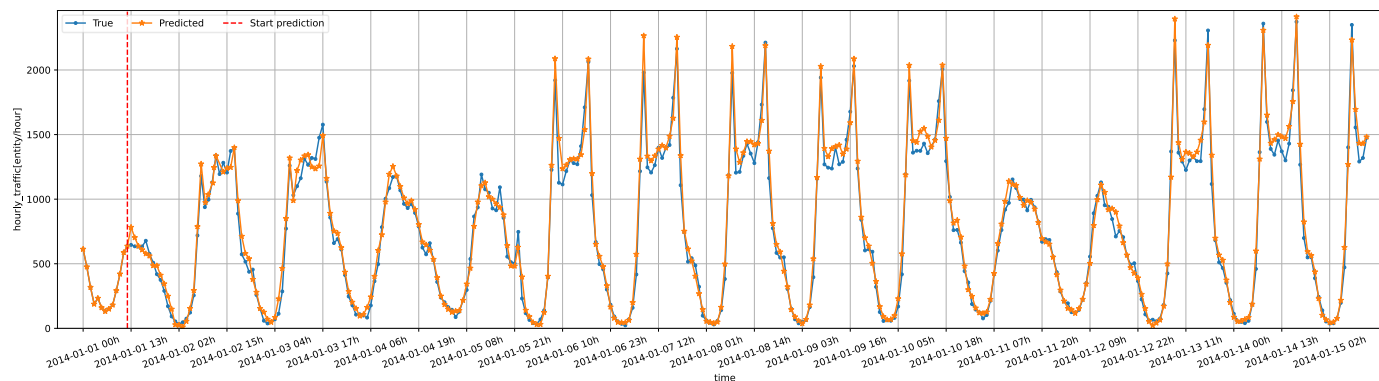
Test RMSE = 94.14



Modello post regolazione ingresso:

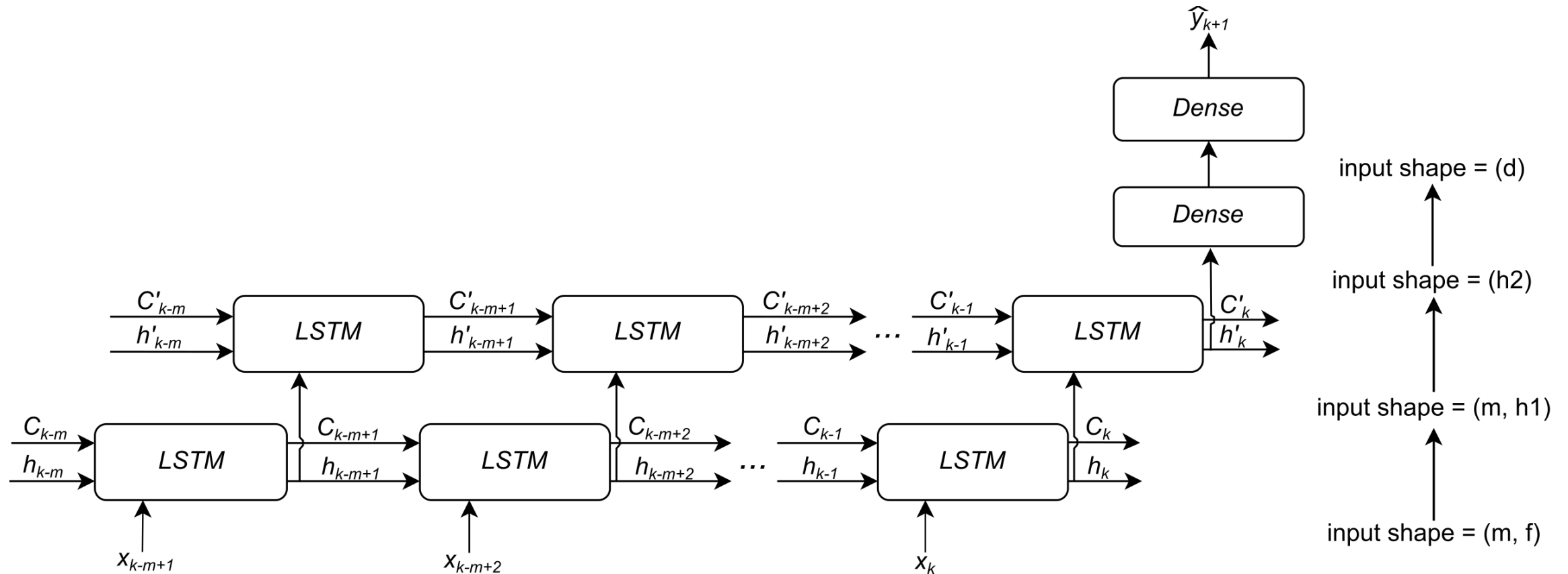
Test MAE = 60.09

Test RMSE = 90,12

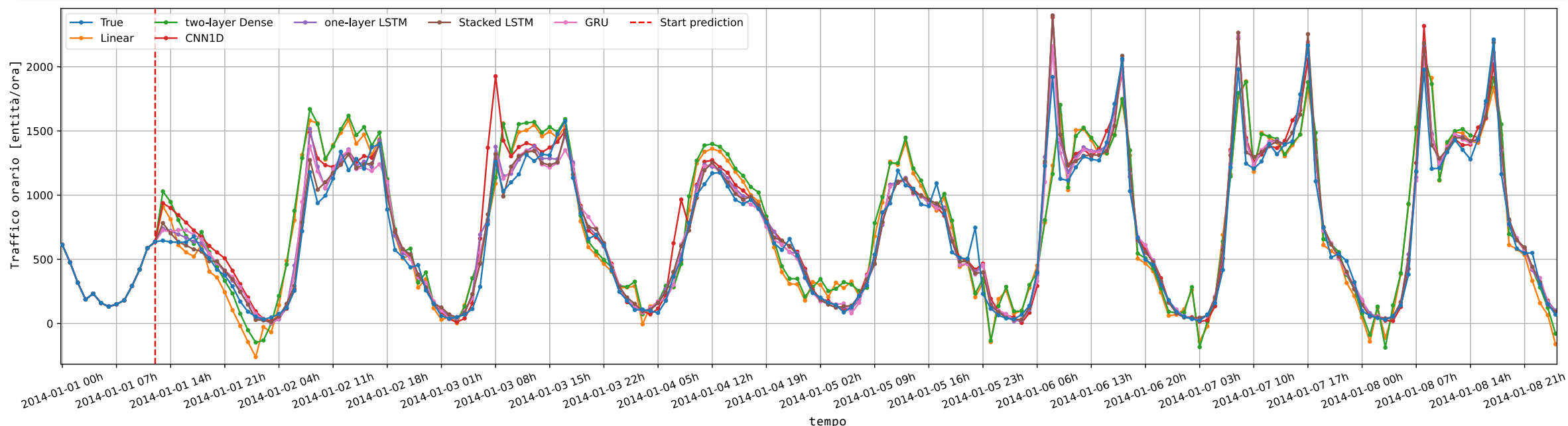




# Stacked LSTM



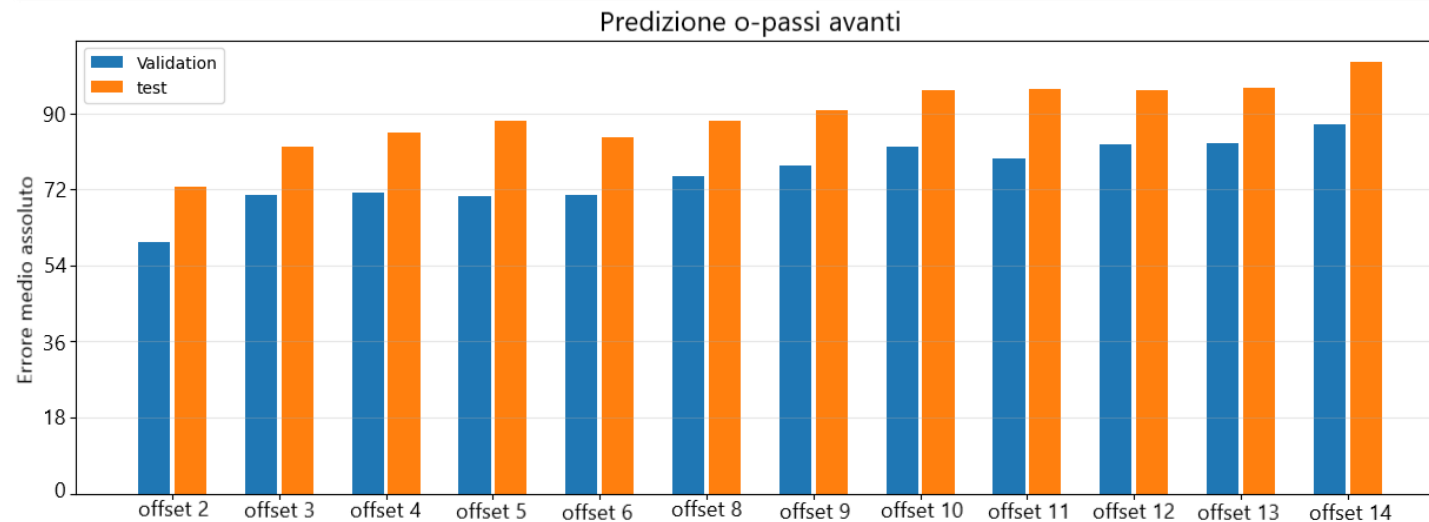
# Valutazioni dei Modelli



## Prestazioni sul Test set:

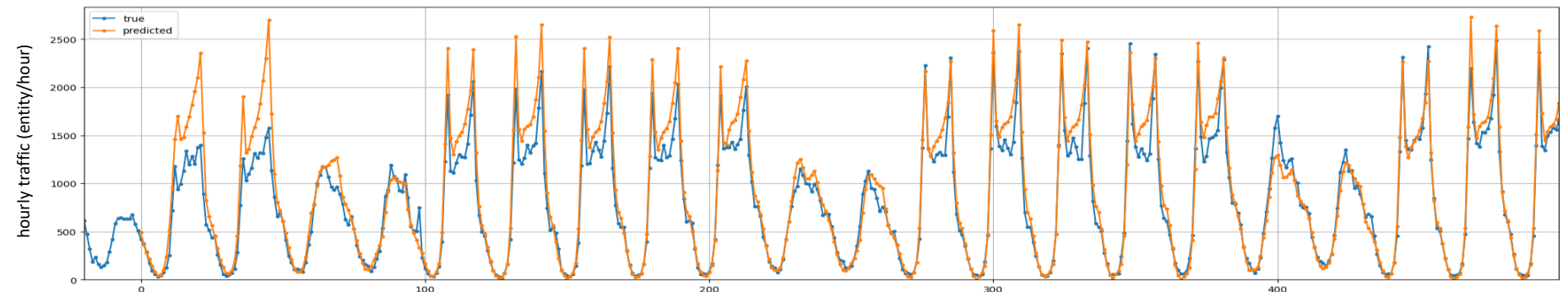
	Lineare	2-Dense	CNN 1D	LSTM semplice	GRU	Stacked LSTM
MAE	183.50	175.09	75.22	69.40	64.91	<b>60.09</b>
RMSE	263.56	252.49	110.77	101.61	93.16	<b>90.12</b>

# Predizione o-Passi Avanti



## Esempio 10-passi avanti:

- MAE = 95.41
- RMSE = 157.27



# Correzione Temporale

---

Correzione basata sulle sole caratteristiche correlate temporalmente alla serie.

In questo lavoro:

- Imputazione tramite free-run dei modelli RNN
- Fine tuning del modello per correggere un sito, diverso di quello del allenamento, ma della stessa categoria.

Fine tuning tramite allenamento di una piccola porzione (6 mesi) del nuovo sito

- Dei soli strati FC
- Di tutti il modello

# Previsione Free-Run

$M[m]$ :

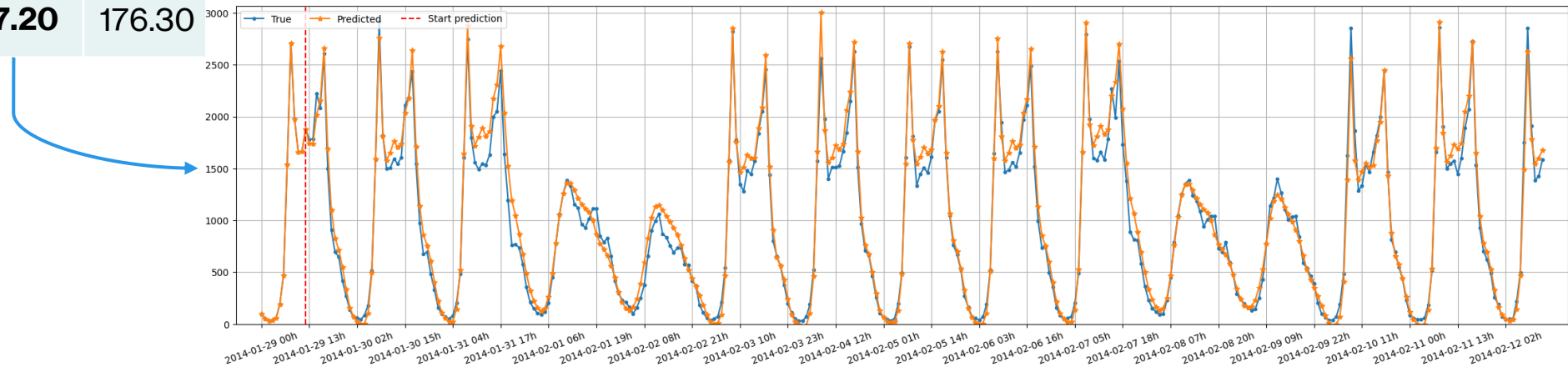
Timestamp	Traffico orario	...
...	...	...
m	$y_m$	...

$M[m + 1]$ :

Timestamp	Traffico orario	...
m	$y_m$	...
m+1	$\hat{y}_{m+1}$	...

$$f_1(M[m]) = \hat{y}_{m+1}$$

	LSTM semplice	GRU	Stacked LSTM
RMSE	168.54	<b>92.41</b>	118.59
MAE	128.02	<b>127.20</b>	176.30

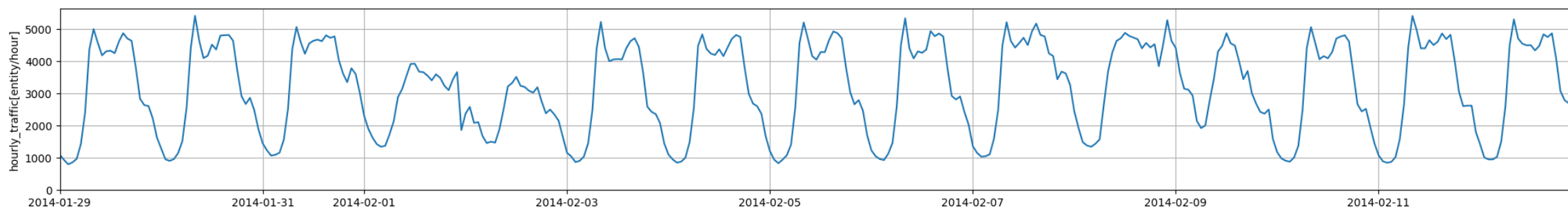


# C.T.: Sito da Adattare

Sito del allenamento originale 10

--->

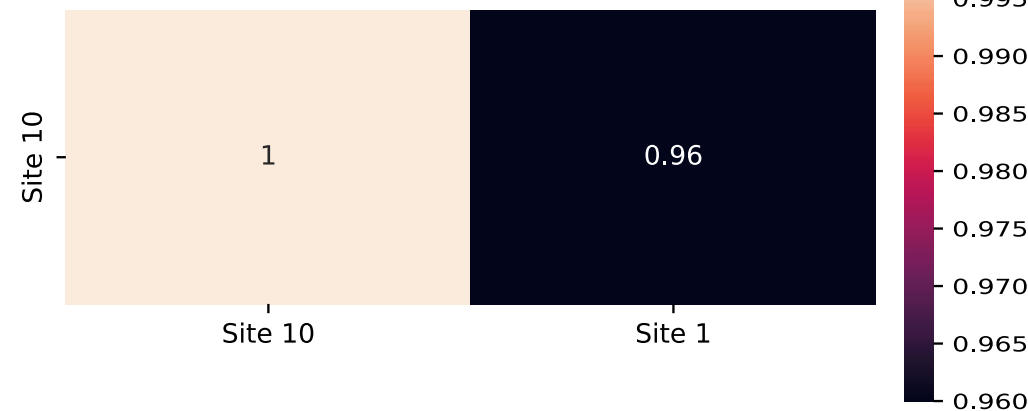
Sito da adattare 1 (Training 6mesi, Test 2settimane)



Heatmap w\Pearson's correlation coefficient

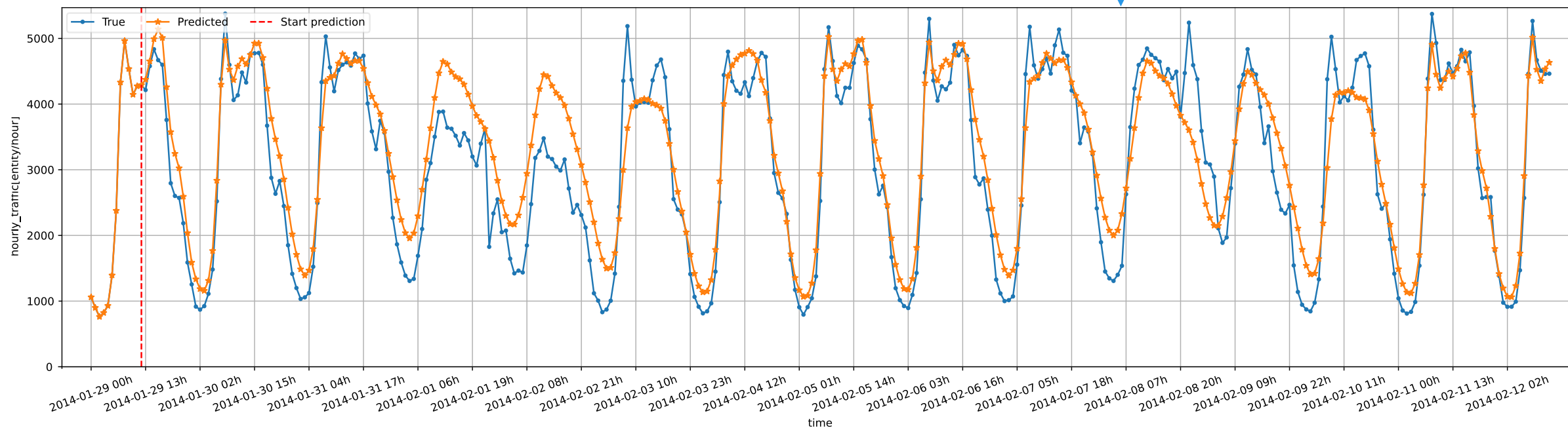


Heatmap w\Spearman rank correlation



# C. T. : Allenamento Soli FC

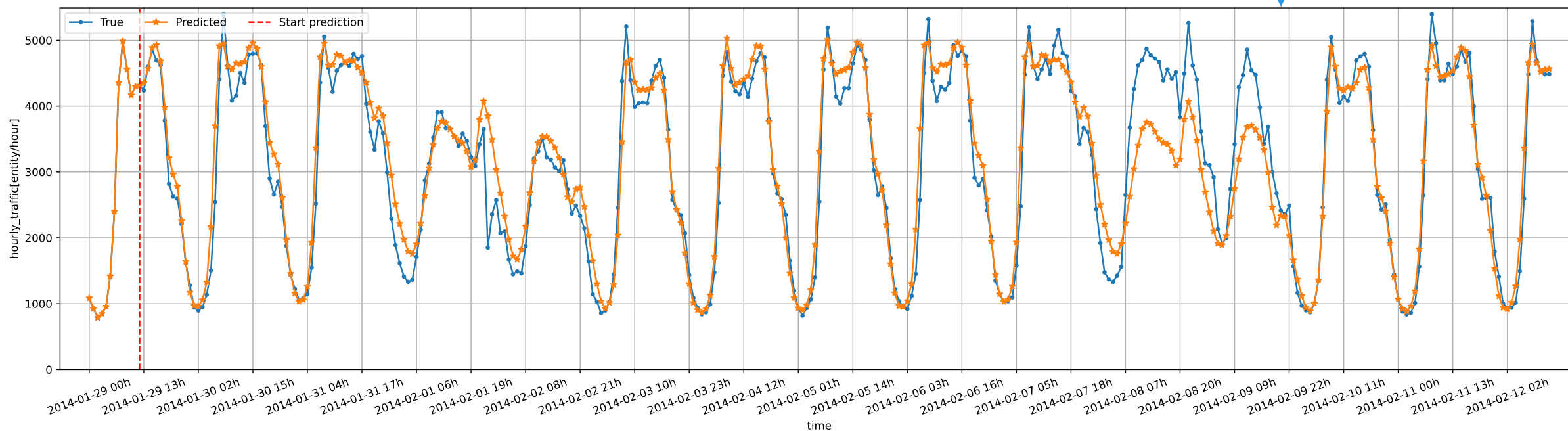
	LSTM semplice	GRU	Stacked LSTM
RMSE	827.91	<b>626.95</b>	630.17
MAE	647.08	<b>493.75</b>	511.73





# C. T. : Allenamento Completo

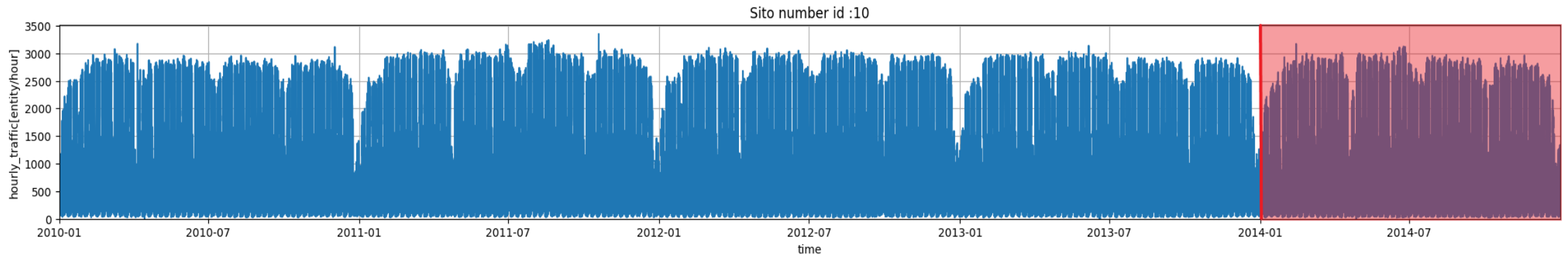
	LSTM semplice	GRU	Stacked LSTM
RMSE	561.76	427.22	<b>407.83</b>
MAE	440.44	303.09	<b>285.21</b>



# Correzione Spaziale

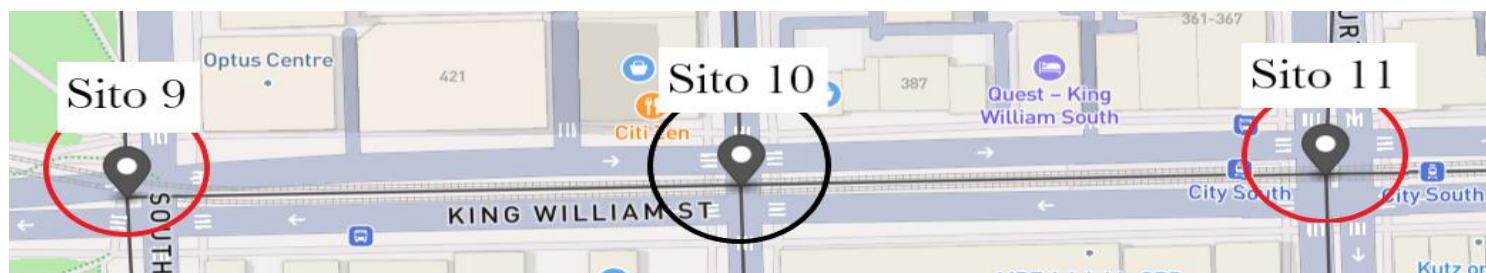
Problematica: Riparare la serie temporale che presenta delle anomalie

Obiettivo: Implementare una tecnica di 'imputazione' basata su l'utilizzo di modelli neurali addestrati per predire il traffico del sito che presenta anomalie basandosi sul traffico di siti limitrofi



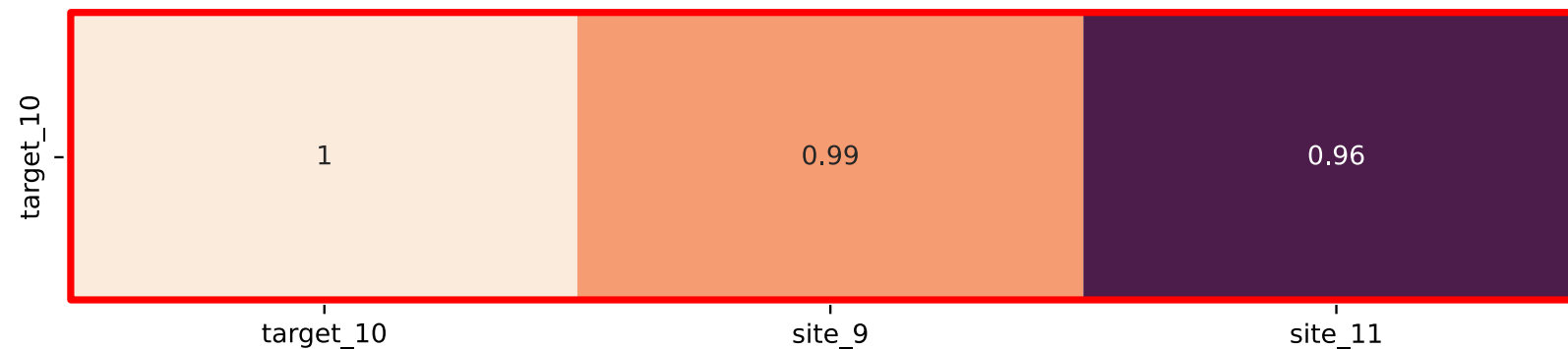
# Impostazione e Valutazione problema

Ipotesi: Siti limitrofi presentano una correlazione spaziale tra i loro flussi di traffico.



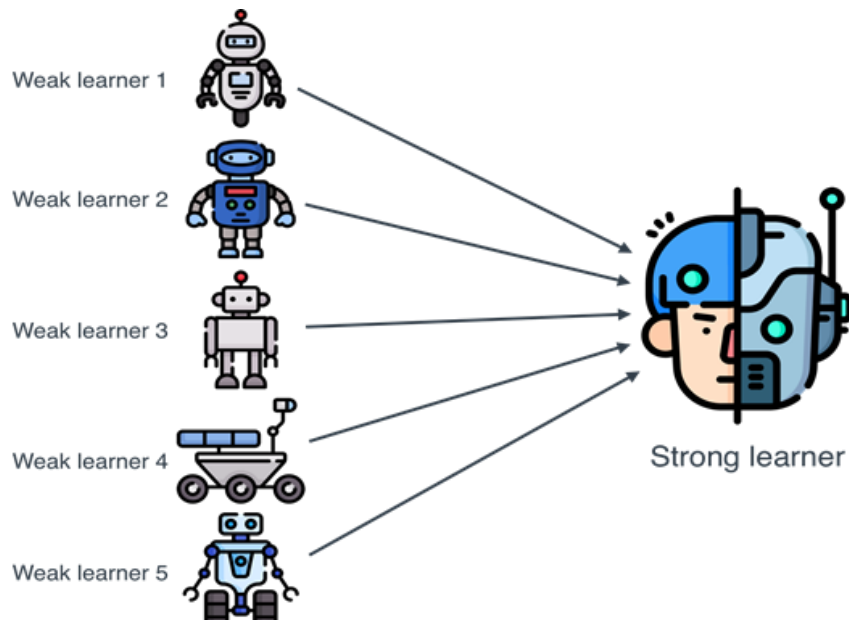
Heatmap w\Pearson's correlation coefficient

Correlazione tra siti:



# Ensemble Learning: apprendimento d'insieme

- L'Ensemble Learning combina diversi modelli di apprendimento per migliorare le prestazioni complessive.
- Esistono diversi tipi di ensemble learning, tra cui:
  - **Bagging:** Combina modelli addestrati su campioni di dati diversi con sostituzione.
  - **Boosting:** Combina modelli sequenzialmente, dove ogni modello cerca di correggere gli errori del modello precedente.
  - **Stacking:** Combina le predizioni di diversi modelli in un nuovo modello.
- Tecnica utilizzata in questo lavoro **Stacking ensemble**



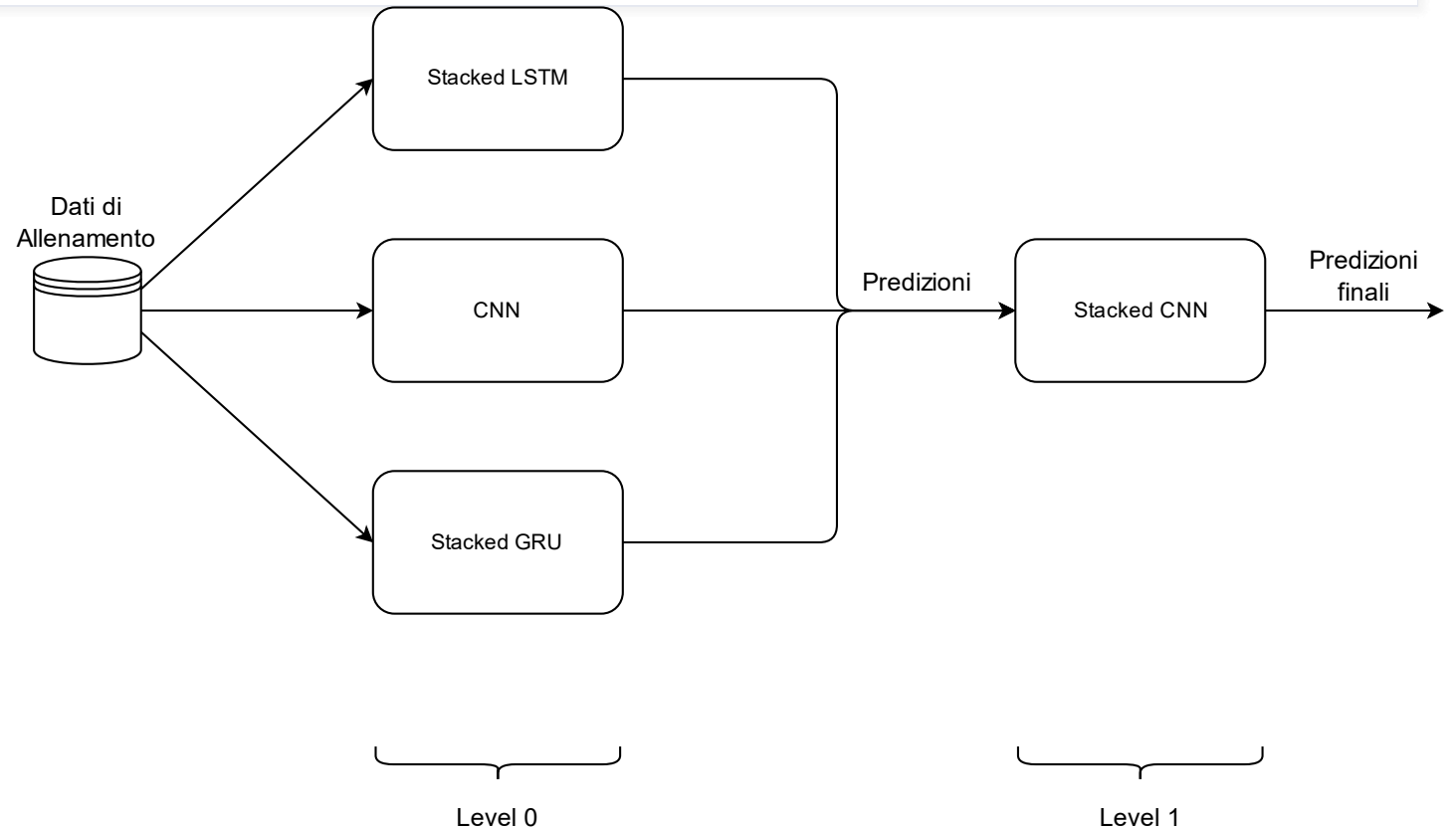
# Schema di Stacking ensemble utilizzato

## Livello 0: Modelli Preliminari

- Reti addestrate sul flusso del traffico dei siti 9 e 11 per predire il flusso del sito 10 (bucato)
- Dataset diviso in: training set (2010-2011), validation set (2012), test set (2013)

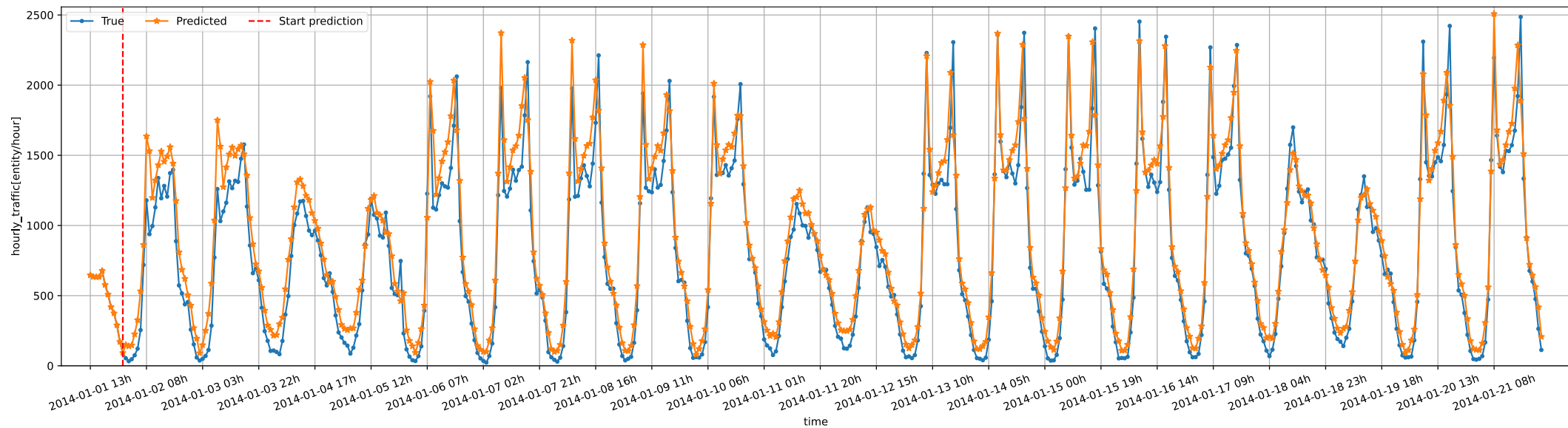
## Livello 1: Rete di Generalizzazione

- Integra le predizioni dei modelli preliminari per un'imputazione più robusta
- Addestrata utilizzando: training set (2012), validation set (2013), test set (2014)



# Valutazione dell'Approccio: Stacked Ensemble Learning

metrica	Stacked LSTM (level 0)	CNN (level 0)	Stacked GRU (level 0)	rete di generalizzazione
Test MAE	152.67	159.43	160.40	<b>117.55</b>



# Conclusione

## Risultati della ricerca:

- Utilizzo efficace di tecniche di ottimizzazione del modello
- Sviluppo di una tecnica di imputazione efficace a breve e lungo termine
- Esplorazione del free-run di RNN come imputazione temporale

## Sviluppi futuri:

- Predizione spazio-temporale estesa a più siti
- Utilizzo di approcci ibridi che combinano il Deep Learning con metodi model-based
- Raffinare l'approccio di correzione
- Espansione ad una previsione estesa a più siti



**Grazie per  
l'attenzione!**

---