

# Tecniche di manutenzione predittiva basate su LSTM in ambito Industria 4.0

Dario Bruneo, Fabrizio De Vita, Antonio Puliafito

Università di Messina - Dipartimento di Ingegneria

{dbruneo,fdevita,apuliafito}@unime.it

## Abstract

Gli aspetti relativi alla manutenzione sono diventati un problema cruciale soprattutto in quei settori in cui il guasto di un componente può compromettere il normale funzionamento di un sistema. Allo stato attuale, la maggior parte dei sistemi industriali permette solamente la rilevazione dei guasti (*run-to-failure*), causando inevitabilmente un blocco dell'intero sistema, o adotta tecniche di manutenzione preventiva ad intervalli temporali prefissati. Recentemente sono state proposte soluzioni di manutenzione *predittiva* che, facendo uso di tecniche di machine learning, permettono di analizzare lo *stato di salute* di un sistema a partire dai dati provenienti dai sensori. In questo lavoro si presentano tecniche di manutenzione predittiva basate su LSTM networks: un approccio all'avanguardia in grado di analizzare la *storia* di un sistema e di predire il suo *tempo di vita* al fine di avviare le procedure di manutenzione nel periodo temporale che ottimizza funzioni di costo specifiche.

## 1 Introduzione

In ambito industriale, la manutenzione predittiva è una delle tecniche più ambiziose per l'implementazione di meccanismi in grado di predire un guasto *prima* che questo avvenga [Aydin e Guldamlasioglu, 2017] a partire dai dati provenienti da svariati sensori (es. temperatura, vibrazione, rumore) connessi ai componenti del sistema in esame. In tal senso, la manutenzione predittiva analizza la storia di un sistema rappresentata dai dati raccolti nel tempo dai sensori al fine di stimarne il Remaining Useful Life (RUL) ed avviare per tempo le procedure di manutenzione, evitando di conseguenza lunghi periodi di blocco [Dong *et al.*, 2017].

Al giorno d'oggi non esiste una tecnica consolidata per una predizione accurata del RUL, tuttavia, negli ultimi anni, alcuni approcci stanno cominciando ad usufruire delle potenzialità del machine learning come tecnica risolutiva. In particolare, quando un sistema risulta essere troppo complesso la stima del RUL risulta essere difficile specialmente quando il numero di sensori e componenti è elevato. E' proprio in questo contesto che il machine learning risulta essere una scelta

vincente in quanto in grado di apprendere tutte le possibili correlazioni tra i dati provenienti dai sensori permettendo quindi una stima accurata del RUL.

## 2 LSTM networks

Grazie alla loro abilità di tenere in memoria lunghe sequenze temporali estraendo al tempo stesso complesse correlazioni tra i dati, le Long Short Term Memories (LSTM) networks si dimostrano un'ottima soluzione per la stima del RUL.

Da un punto di vista architetturale, una LSTM è una Recurrent Neural Network (RNN) che utilizza un algoritmo efficiente basato su gradiente in grado di mantenere costante l'errore durante la fase di apprendimento [Hochreiter e Schmidhuber, 1997]. La Figura 1 mostra una cella LSTM composta da tre porte di ingresso: l'*input gate*, il *forget gate* e l'*output gate* governate dalle seguenti equazioni:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C * [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t \quad (4)$$

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t \circ \tanh(C_t) \quad (6)$$

dove il  $\circ$  rappresenta il prodotto elemento per elemento,  $f_t$ ,  $i_t$ ,  $o_t$  il *forget gate*, *input gate* e *output gate*, e  $W_f$ ,  $W_i$ ,  $W_C$ ,  $W_o$  sono le corrispondenti matrici dei pesi.

Quando i dati entrano all'interno della LSTM, il primo step consiste nello stabilire quali informazioni devono essere tenute in memoria e quali no. Tale decisione viene effettuata tramite l'eq. (1) la quale prende in input un array  $x_t$  al generico istante di tempo  $t$  assieme all'output  $h_{t-1}$  proveniente dall'istante di tempo precedente. Il risultato ottenuto viene in seguito passato alla funzione di attivazione *sigmoide*  $\sigma$ , la quale ritorna un numero compreso tra 0 e 1 per ogni elemento della cella di stato  $C_{t-1}$  dove 0 significa che quell'elemento può essere dimenticato ed 1 significa che quell'elemento deve essere mantenuto nel tempo. Il secondo step viene a sua

