FMECA - On the Edge

# Progetto n° 22

User Requirements Specification Document

DIBRIS – Università di Genova. Scuola Politecnica, Corso di Ingegneria del Software 80154

**DATA : 23/04/2020**

**VERSION : 1.2**

**Autori**

Matteo Cardano

Lavaggi Stefano

**REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| Versione | Data | Autori | Note |
| 1.0 | 13/04/2020 | Cardano, Lavaggi | Prima stesura |
| 1.1 | 23/04/2020 | Cardano, Lavaggi | Dettagli progetto |
| 1.2 | 08/05/2020 | Cardano | Aggiornamento Dataset |
|  |  |  |  |

**Indice dei Contenuti**

1. [Introduzione 3](#_gjdgxs)
   1. [Scopo del Documento 3](#_30j0zll)
   2. Overview del Documento 3
   3. [Definizioni e Acronimi 3](#_1fob9te)
2. [Descrizione Generale del Sistema 4](#_tyjcwt)
   1. [Contesto 4](#_3dy6vkm)
   2. Motivazioni 4
   3. [Obiettivo del Progetto](#_1t3h5sf) 5
   4. Stakeholders 6
3. [User Requirement](#_2s8eyo1) 6

# Introduzione

## Scopo del Documento

Analisi della problematica cliente. Il documento è necessario affinché le componenti richieste del cliente (RINA) vengano rispettate dai progettisti.

## Overview del Documento

Nel documento è descritto il problema, le sue caratteristiche e alcune considerazioni tecniche. Viene fornita inoltre una descrizione della struttura generale.

## Definizioni e Acronimi

|  |  |
| --- | --- |
| **Acronimo-Nome** | **Definizione** |
| SE | Software Engineering |
| AI | Artificial intelligence |
| NN | Neural network |
| RB | RaspBerry PI |
| AR | Arduino |
| TF | TensorFlow |
| KR | Keras |

# Descrizione Generale del Sistema

Il progetto ha l’obiettivo di sviluppare una NN configurabile e adattabile a vari contesti applicativi, per poter gestire macchine utensili e altri strumenti produttivi; In modo da programmare ed effettuare una manutenzione mirata, evitando guasti e fermi macchina per rotture e anomalie; Il prodotto finale sarà testato e installato su HW embedded con risorse e capacità limitate come ad esempio un RB o AR.

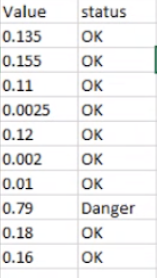
## Contesto

Il progetto è legato ad un contesto industriale. “Il case study” fa riferimento ad una piegatrice fornita di sensori (nel particolare accelerometri) per valutare vibrazione e oscillazioni durante il processo produttivo. Essi sono utilizzati per segnalare all’utilizzatore o al manutentore eventuali precursori di possibili malfunzionamenti o rotture, permettendo così la possibilità di effettuare una manutenzione predittiva.

## Motivazioni

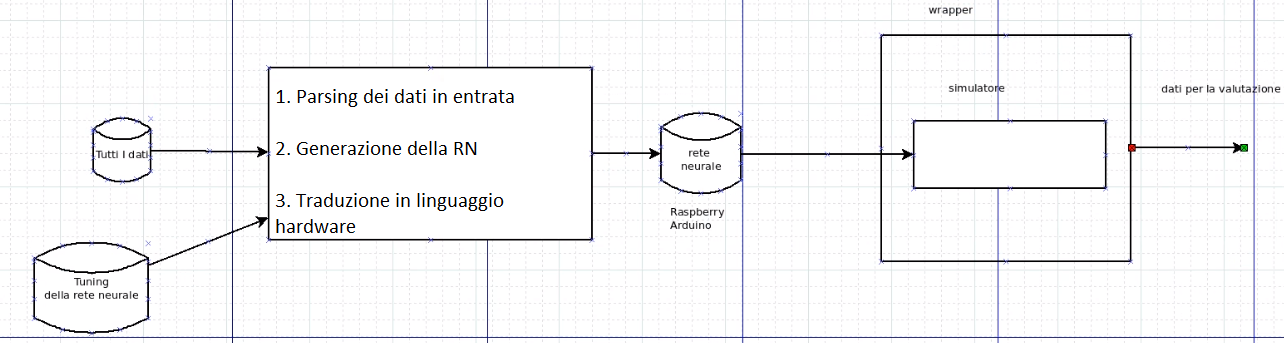
Attualmente è già implementa una soluzione di AI che sfrutta un approccio differente, sul bordo macchina si effettua solo la rilevazione dei campioni, l’allenamento e la valutazione dei rilevamenti viene effettuata completamente in ‘cloud’ tramite algoritmi specifici. Sono note solo le seguenti caratteristiche:

* 1 ‘layer’ di input
* 1 ‘hidden layer’ contenente 10 neuroni
* 1 ‘layer’ di output
* Funzione di attivazione (funzione Sigmoidale)

L’impianto preso in considerazione risulta dotato di 4 sensori disposti in vari punti. In una prima versione considereremo solamente uno di questi, perché i restanti al momento non fornisco dati significativi. I valori che vengono ricavati dai dispositivi hanno 2 possibili stati associati: - (status = ‘OK’), corretto funzionamento del sistema - (‘status = Danger’) presenza di malfunzionamenti o rotture (valori molto “piccoli” 0.001 - 0.011 indicano macchinario inattivo). Frequenza di campionamento 0.033 Hz.

## Obiettivo del Progetto

In seguito ai colloqui introduttivi e varie consultazioni abbiamo deciso di proporre una struttura di questo tipo :



Si vuole realizzare un software in grado di fare il parsing di un file contenente il dataset (es .csv, excel, ecc..ecc…) e possibili informazioni, che possono risultare utili per costruire campioni significativi. Il passo successivo sarà quello di realizzare una NN specifica consentendo all’utente di personalizzare il sistema fornendo i principali parametri caratteristici come ad esempio: il numero di nodi, ‘layer’ e la funzione caratteristica.

Successivamente si passerà alla parte di ‘training’ della NN. L’ allenamento verrà effettuato in locale. Da esso si fornirà il primo feedback qualitativo del sistema, attraverso indici caratteristici delle reti neurali. Da qua si cercherà di ottimizzare le prestazioni in modo automatico o attraverso l’intervento diretto dell’utente.

Appena si otterrà un risultato ritenuto ottimale si valuteranno le risorse necessarie al suo funzionamento, in modo tale da validare un eventuale portabilità su dispositivi HW. Nel caso specifico del RB si effettuerà un simulazione della NN ,‘arredata’ di tutti i valori ottimali in precedenza calcolati e opportunamente settata, tramite simulatore Raspberry Pi con S.O Raspbian, Attraverso la simulazione forniremo indici prestazionali legati ai tempi di funzionamento sull’ HW. Nel caso in cui i risultati soddisfino i requisiti utente il software fornirà in output le caratteristiche e il codice sorgente della rete. Nella versione fisica (installata direttamente sull’ HW) il sistema in caso di rilevamento di anomalia genererà messaggi di errore o segnali per permettere l’intervento della manutenzione.

Per realizzarlo sfrutteremo come linguaggio di programmazione python vista la compatibilità con l’HW e la disponibilità di strumenti come le librerie di KR e TS.

## Stakeholders

* Sviluppatore della rete e configuratore della rete neurale
* Manutentore del sistema
* Operatore d’impianto
* Industria manifatturiera
* Organizzazione di Sviluppo e Progettazione

# User Requirement

|  |  |
| --- | --- |
| **PRIORITA’** | **SIGNIFICATO** |
| **M** | Mandatory. Requisito obbligatorio. |
| **D** | Desiderabile. Requisito che dovrebbe essere inserito nel sistema, a meno che il costo per implementarla non sia troppo alto. |
| **O** | Optional. Una funzionalità marcata con O può essere inserita nel sistema, a discrezione del manager del progetto. Ad esempio se il tempo di sviluppo è minore di quello previsto oppure se il costo per implementarla non è troppo alto. |
| **E** | future Enhancement. Questo requisito viene lasciato per la prossima release. |

|  |  |  |
| --- | --- | --- |
| **ID** | **DESCRIZIONE** | **PRIORITA’** |
| 1 | ‘Parsing’ dei dati in entrata | M |
| 2 | Possibilità di scelta delle caratteristiche base della RN | M |
| 3 | Creazione RN | M |
| 4 | Allenamento RN | M |
| 5 | Prima valutazione dei risultati ottenuti | M |
| 6 | Ottimizzazione della RN | D |
| 7 | Possibilità di scelta dell HW | O |
| 8 | Generazione sorgente per HW scelto | M |
| 9 | Testing della RN sul simulatore | D |
| 10 | Portabilità sul HW | E |
| 11 | Gestione segnalazione errori | E |
| 12 | Controllo strumenti di segnalazione guasto | E |