Protocollo sicuro per l'elaborazione di dati cifrati mediante una rete neurale

NNSec - Neural Network Secure

Michele Caini

Università degli Studi di Firenze, Facoltà di Ingegneria

18 Dicembre 2007

Motivazioni

SPEED (**S**ignal **P**rocessing in the **E**ncrypt**E**d **D**omain)

Col progetto SPEED vengono avvicinati due mondi apparentemente distanti, come:

- **Tecniche di *signal processing*:**
  - Strumenti di classificazione dei dati in classi di appartenenza
  - Percettrone, reti neurali feed-forward multi-livello
- **Tecniche di crittografia:**
  - Sistemi dalle interessanti quanto utili proprietà omomorfiche
  - Cifrario di Paillier, generalizzazione di Damgård-Jurik

Il lavoro di tesi

Partendo dallo studio di un protocollo già esistente nella teoria:

- Algoritmi e procedure prendono vita sotto forma di classi e relazioni fra esse
- Viene realizzato un classificatore in grado di operare con dati cifrati
- Si ottengono reti neurali capaci di lavorare in dominio cifrato

Motivazioni

SPEED (**S**ignal **P**rocessing in the **E**ncrypt**E**d **D**omain)

Col progetto SPEED vengono avvicinati due mondi apparentemente distanti, come:

- **Tecniche di *signal processing***:
  - Strumenti di classificazione dei dati in classi di appartenenza
  - Percettrone, reti neurali feed-forward multi-livello
- **Tecniche di crittografia**:
  - Sistemi dalle interessanti quanto utili proprietà omomorfiche
  - Cifrario di Paillier, generalizzazione di Damgård-Jurik

Il lavoro di tesi

Partendo dallo studio di un protocollo già esistente nella teoria:

- Algoritmi e procedure prendono vita sotto forma di classi e relazioni fra esse
- Viene realizzato un classificatore in grado di operare con dati cifrati
- Si ottengono reti neurali capaci di lavorare in dominio cifrato

Motivazioni

SPEED (**S**ignal **P**rocessing in the **E**ncrypt**E**d **D**omain)

Col progetto SPEED vengono avvicinati due mondi apparentemente distanti, come:

- **Tecniche di *signal processing***:
  - Strumenti di classificazione dei dati in classi di appartenenza
  - Percettrone, reti neurali feed-forward multi-livello
- **Tecniche di crittografia**:
  - Sistemi dalle interessanti quanto utili proprietà omomorfiche
  - Cifrario di Paillier, generalizzazione di Damgård-Jurik

Il lavoro di tesi

Partendo dallo studio di un protocollo già esistente nella teoria:

- Algoritmi e procedure prendono vita sotto forma di classi e relazioni fra esse
- Viene realizzato un classificatore in grado di operare con dati cifrati
- Si ottengono reti neurali capaci di lavorare in dominio cifrato

Motivazioni

SPEED (**S**ignal **P**rocessing in the **E**ncrypt**E**d **D**omain)

Col progetto SPEED vengono avvicinati due mondi apparentemente distanti, come:

- **Tecniche di *signal processing***:
  - Strumenti di classificazione dei dati in classi di appartenenza
  - Percettrone, reti neurali feed-forward multi-livello
- **Tecniche di crittografia**:
  - Sistemi dalle interessanti quanto utili proprietà omomorfiche
  - Cifrario di Paillier, generalizzazione di Damgård-Jurik

Il lavoro di tesi

Partendo dallo studio di un protocollo già esistente nella teoria:

- Algoritmi e procedure prendono vita sotto forma di classi e relazioni fra esse
- Viene realizzato un classificatore in grado di operare con dati cifrati
- Si ottengono reti neurali capaci di lavorare in dominio cifrato

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Scenario

Gli attori. . .

- Bob vuole mettere a disposizione una rete neurale opportunamente allenata
- Alice vuole usufruire del servizio offerto da Bob per elaborare i propri dati
- Bob e Alice non si fidano l'uno dell'altro
- Non si può o non si vuole trovare ad una terza parte fidata per entrambi

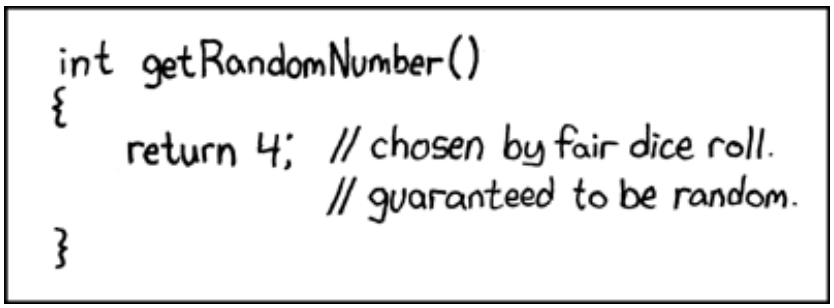
. . . E un caso concreto

Si immagini:

- Una rete neurale in grado di diagnosticare una malattia più o meno grave
- Un capo (o ex tale) di governo con sintomi particolari

La riservatezza dei dati acquista un valore inestimabile.

Requisiti



Sicurezza per Alice

Risiede in quella del cifrario sottostante:

- Protezione dei dati forniti in ingresso
- Protezione dei risultati ottenuti

Sicurezza per Bob

Consiste nel proteggere la struttura della rete neurale attraverso:

- Espansione tramite aggiunta di neuroni fittizi ai livelli intermedi
- Permutazione di neuroni in uno stesso livello intermedio
- Cambiamento dello stato del singolo neurone intermedio

Requisiti

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

Sicurezza per Alice

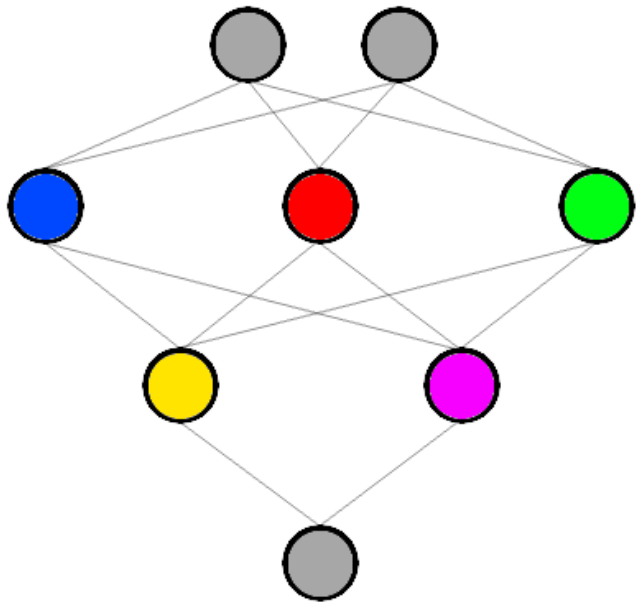
Risiede in quella del cifrario sottostante:

- Protezione dei dati forniti in ingresso
- Protezione dei risultati ottenuti

Sicurezza per Bob

Consiste nel proteggere la struttura della rete neurale attraverso:

- Espansione tramite aggiunta di neuroni fittizi ai livelli intermedi
- Permutazione di neuroni in uno stesso livello intermedio
- Occultamento dello stato del singolo neurone intermedio



Requisiti

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

Sicurezza per Alice

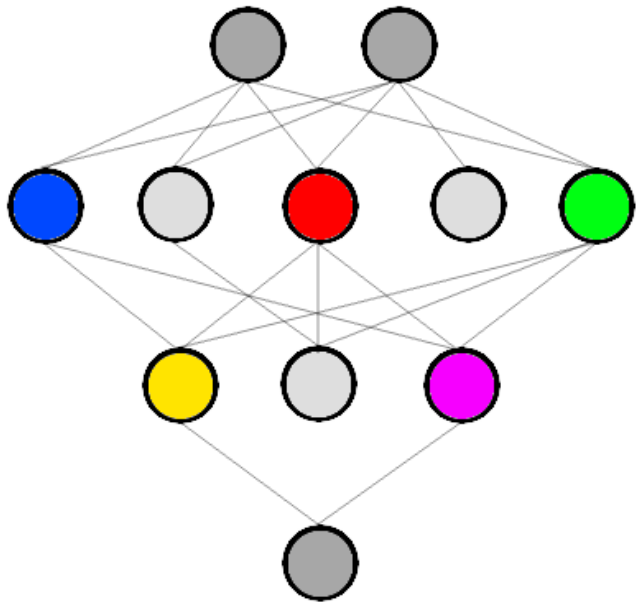
Risiede in quella del cifrario sottostante:

- Protezione dei dati forniti in ingresso
- Protezione dei risultati ottenuti

Sicurezza per Bob

Consiste nel proteggere la struttura della rete neurale attraverso:

- Espansione tramite aggiunta di neuroni fittizi ai livelli intermedi
- Permutazione di neuroni in uno stesso livello intermedio
- Occultamento dello stato del singolo neurone intermedio



Requisiti

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

Sicurezza per Alice

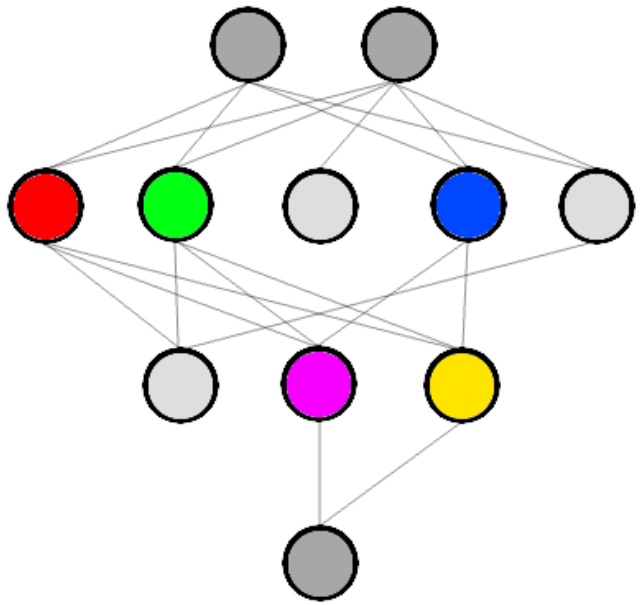
Risiede in quella del cifrario sottostante:

- Protezione dei dati forniti in ingresso
- Protezione dei risultati ottenuti

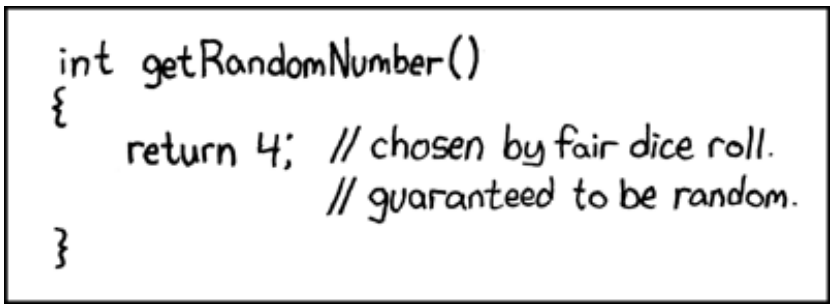
Sicurezza per Bob

Consiste nel proteggere la struttura della rete neurale attraverso:

- Espansione tramite aggiunta di neuroni fittizi ai livelli intermedi
- Permutazione di neuroni in uno stesso livello intermedio
- Occultamento dello stato del singolo neurone intermedio



Requisiti



Sicurezza per Alice

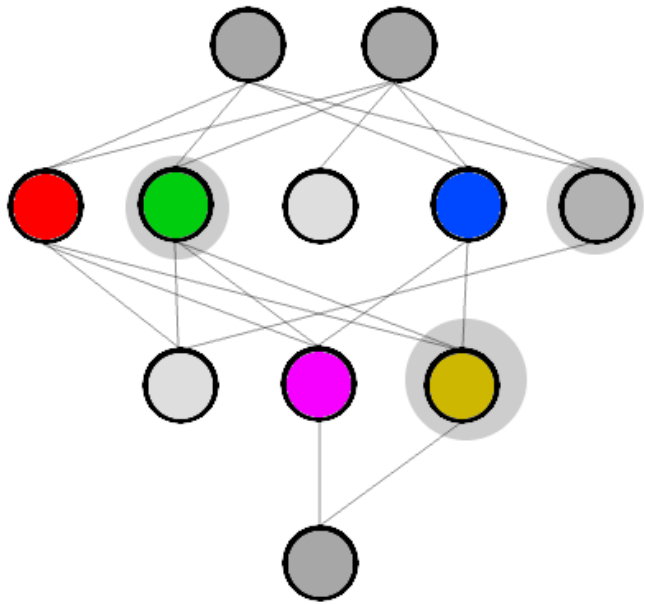
Risiede in quella del cifrario sottostante:

- Protezione dei dati forniti in ingresso
- Protezione dei risultati ottenuti

Sicurezza per Bob

Consiste nel proteggere la struttura della rete neurale attraverso:

- Espansione tramite aggiunta di neuroni fittizi ai livelli intermedi
- Permutazione di neuroni in uno stesso livello intermedio
- Occultamento dello stato del singolo neurone intermedio



Proprietà Omomorfiche

Il cifrario di Paillier ha caratteristiche molto utili e interessanti, in particolare:

Proprietà omomorfiche ...

Siano m_i messaggi in chiaro ($i = 1, \ldots, n$), $c_i = E(m_i)$ la loro versione cifrata (di conseguenza, $m_i = D(c_i)$), siano a_i un insieme di n valori interi, allora:

$$D\left(\prod_{i=1}^{n} c_i^{a_i}\right) = D\left(\prod_{i=1}^{n} E(m_i)^{a_i}\right) = \sum_{i=1}^{n} a_i \cdot m_i$$

...E reti neurali

Siano x un nodo nel j-esimo livello e $\bar{x}$ e $\bar{w}$ i vettori di nodi connessi e pesi associati (di lunghezza n), sia $\bar{c} = E(\bar{x})$. Il valore cifrato di attivazione d_x per x risulta da:

$$d_x = \prod_{i=1}^{n} c_i^{w_i} \quad \Longrightarrow \quad a_x = D(d_x) = \sum_{i=1}^{n} w_i \cdot x_i$$

Proprietà Omomorfiche

Il cifrario di Paillier ha caratteristiche molto utili e interessanti, in particolare:

Proprietà omomorfiche ...

Siano m_i messaggi in chiaro ($i = 1, \ldots, n$), $c_i = E(m_i)$ la loro versione cifrata (di conseguenza, $m_i = D(c_i)$), siano a_i un insieme di n valori interi, allora:

$$D\left(\prod_{i=1}^{n} c_i^{a_i}\right) = D\left(\prod_{i=1}^{n} E(m_i)^{a_i}\right) = \sum_{i=1}^{n} a_i \cdot m_i$$

...E reti neurali

Siano x un nodo nel j-esimo livello e $\bar{x}$ e $\bar{w}$ i vettori di nodi connessi e pesi associati (di lunghezza n), sia $\bar{c} = E(\bar{x})$. Il valore cifrato di attivazione d_x per x risulta da:

$$d_x = \prod_{i=1}^{n} c_i^{w_i} \quad \Longrightarrow \quad a_x = D(d_x) = \sum_{i=1}^{n} w_i \cdot x_i$$

Il Protocollo

Neuroni di ingresso

Per ogni neurone di ingresso i:

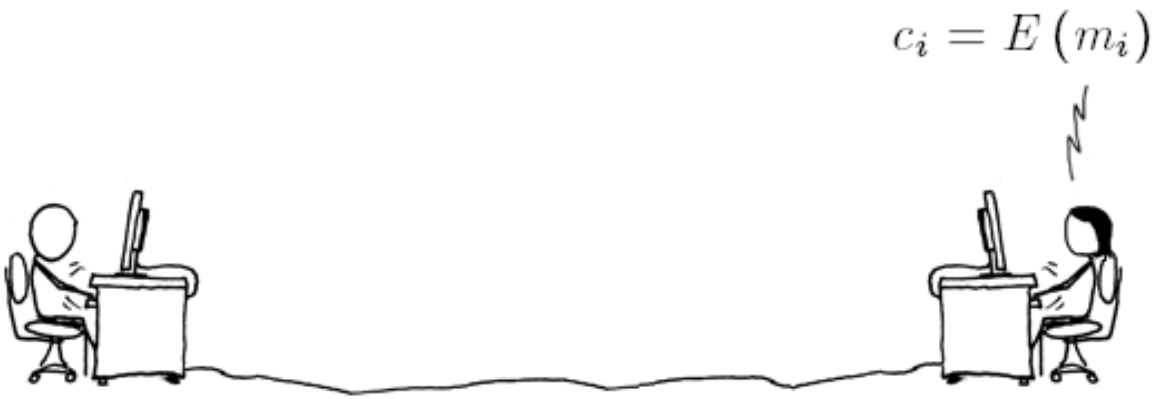
- Alice cifra il valore in ingresso m_i con la propria chiave pubblica
- Alice invia il valore cifrato $c_i = E(m_i)$ a Bob, il quale lo associa al corrispondente neurone di ingresso della rete neurale per l'elaborazione

Il Protocollo

Neuroni di ingresso

Per ogni neurone di ingresso i:

- Alice cifra il valore in ingresso m_i con la propria chiave pubblica
- Alice invia il valore cifrato $c_i = E(m_i)$ a Bob, il quale lo associa al corrispondente neurone di ingresso della rete neurale per l'elaborazione

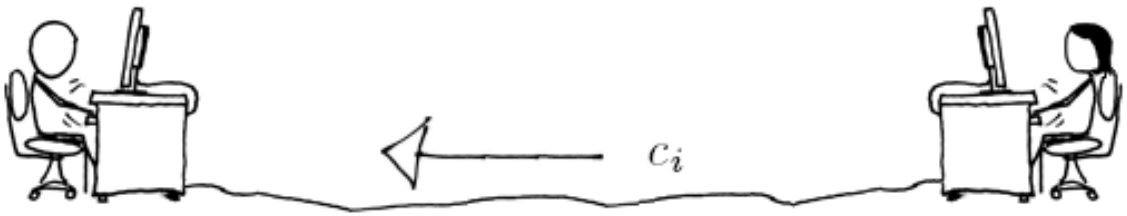


Il Protocollo

Neuroni di ingresso

Per ogni neurone di ingresso i:

- Alice cifra il valore in ingresso m_i con la propria chiave pubblica
- Alice invia il valore cifrato $c_i = E(m_i)$ a Bob, il quale lo associa al corrispondente neurone di ingresso della rete neurale per l'elaborazione



Il Protocollo

Neuroni intermedi

Per ogni neurone intermedio k, Bob ricava il valore di attivazione z'_k come segue:

- Calcola il valore cifrato $d_k = E(a_k)$ e genera in modo casuale $t_j \in \{-1, 1\}$: se $t_j = -1$ allora $d'_k = d_k^{-1} = E(-a_k)$, altrimenti $d'_k = d_k$
- Invia d'_k ad Alice, la quale computa e ritorna: $z_k = E(g(D(d_k)))$
  - $g(a)$ funzione non lineare di attivazione del neurone (segno o sigmoide)
  - Necessario ed unico punto di interazione fra le parti
- Se $t_j = -1$ allora $z'_k = E(1) z_k^{-1}$, altrimenti $z'_k = z_k$
  (segue dalle proprietà di anti-simmetria della funzione $g(a)$)

Il Protocollo

Neuroni intermedi

Per ogni neurone intermedio k, Bob ricava il valore di attivazione z'_k come segue:

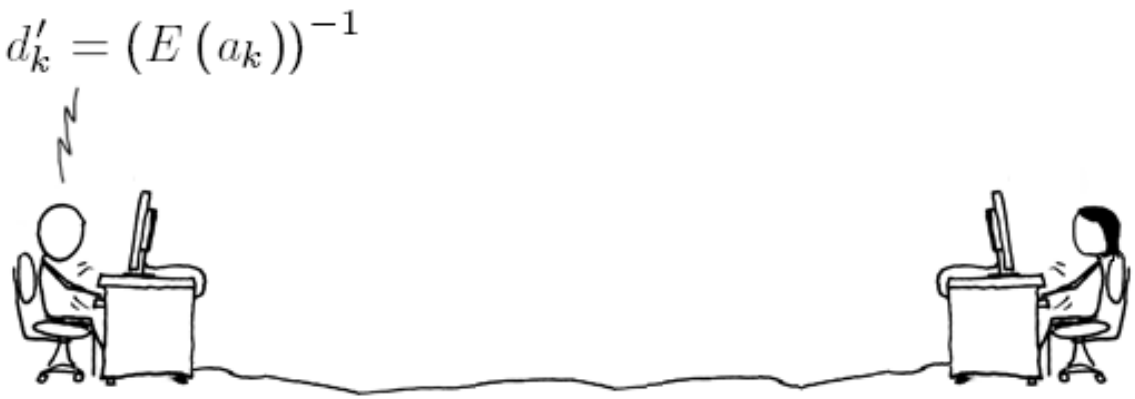
- Calcola il valore cifrato $d_k = E(a_k)$ e genera in modo casuale $t_j \in \{-1, 1\}$: se $t_j = -1$ allora $d'_k = d_k^{-1} = E(-a_k)$, altrimenti $d'_k = d_k$
- Invia d'_k ad Alice, la quale computa e ritorna: $z_k = E(g(D(d_k)))$
  - $g(a)$ funzione non lineare di attivazione del neurone (segno o sigmoide)
  - Necessario ed unico punto di interazione fra le parti
- Se $t_j = -1$ allora $z'_k = E(1)\, z_k^{-1}$, altrimenti $z'_k = z_k$ (segue dalle proprietà di anti-simmetria della funzione $g(a)$)

Il Protocollo

Neuroni intermedi

Per ogni neurone intermedio k, Bob ricava il valore di attivazione z'_k come segue:

- Calcola il valore cifrato $d_k = E(a_k)$ e genera in modo casuale $t_j \in \{-1, 1\}$: se $t_j = -1$ allora $d'_k = d_k^{-1} = E(-a_k)$, altrimenti $d'_k = d_k$
- Invia d'_k ad Alice, la quale computa e ritorna: $z_k = E(g(D(d_k)))$
  - $g(a)$ funzione non lineare di attivazione del neurone (segno o sigmoide)
  - Necessario ed unico punto di interazione fra le parti
- Se $t_j = -1$ allora $z'_k = E(1)\, z_k^{-1}$, altrimenti $z'_k = z_k$
  (segue dalle proprietà di anti-simmetria della funzione $g(a)$)

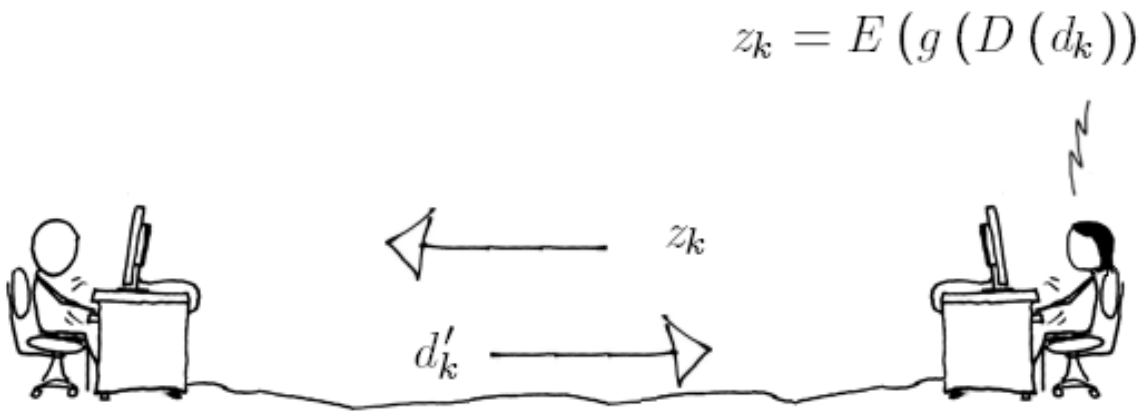


Il Protocollo

Neuroni intermedi

Per ogni neurone intermedio k, Bob ricava il valore di attivazione z'_k come segue:

- Calcola il valore cifrato $d_k = E(a_k)$ e genera in modo casuale $t_j \in \{-1, 1\}$: se $t_j = -1$ allora $d'_k = d_k^{-1} = E(-a_k)$, altrimenti $d'_k = d_k$
- Invia d'_k ad Alice, la quale computa e ritorna: $z_k = E(g(D(d_k)))$
  - $g(a)$ funzione non lineare di attivazione del neurone (segno o sigmoide)
  - Necessario ed unico punto di interazione fra le parti
- Se $t_j = -1$ allora $z'_k = E(1)\, z_k^{-1}$, altrimenti $z'_k = z_k$
  (segue dalle proprietà di anti-simmetria della funzione $g(a)$)

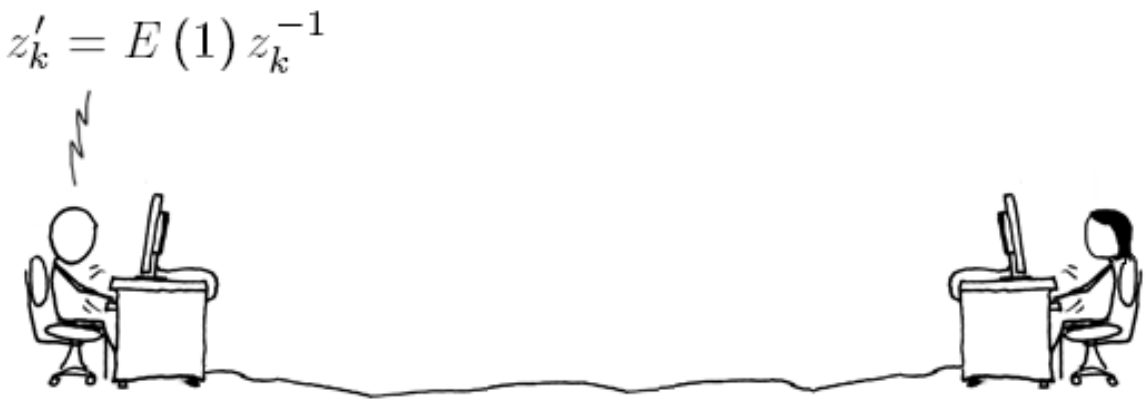


Il Protocollo

Neuroni intermedi

Per ogni neurone intermedio k, Bob ricava il valore di attivazione z'_k come segue:

- Calcola il valore cifrato $d_k = E(a_k)$ e genera in modo casuale $t_j \in \{-1, 1\}$: se $t_j = -1$ allora $d'_k = d_k^{-1} = E(-a_k)$, altrimenti $d'_k = d_k$
- Invia d'_k ad Alice, la quale computa e ritorna: $z_k = E(g(D(d_k)))$
  - $g(a)$ funzione non lineare di attivazione del neurone (segno o sigmoide)
  - Necessario ed unico punto di interazione fra le parti
- Se $t_j = -1$ allora $z'_k = E(1)\, z_k^{-1}$, altrimenti $z'_k = z_k$
  (segue dalle proprietà di anti-simmetria della funzione $g(a)$)



Il Protocollo

Neuroni di uscita

Per ogni neurone di uscita j:

- Bob computa il valore: $d_j = E(a_j)$, inviandolo poi ad Alice
- Alice ricava il valore di uscita del singolo nodo come: $z_j = g(D(d_j))$

Nota: I valori così ottenuti rappresentano il risultato della computazione tramite la rete neurale di Bob sui dati forniti in ingresso da Alice, avvenuta:

- Oscurando la rete neurale, preservandone la struttura interna
- Garantendo riservatezza per i dati di Alice

Il Protocollo

Neuroni di uscita

Per ogni neurone di uscita j:

- Bob computa il valore: $d_j = E(a_j)$, inviandolo poi ad Alice
- Alice ricava il valore di uscita del singolo nodo come: $z_j = g(D(d_j))$

Nota: I valori così ottenuti rappresentano il risultato della computazione tramite la rete neurale di Bob sui dati forniti in ingresso da Alice, avvenuta:

- Oscurando la rete neurale, preservandone la struttura interna
- Garantendo riservatezza per i dati di Alice

Il Protocollo

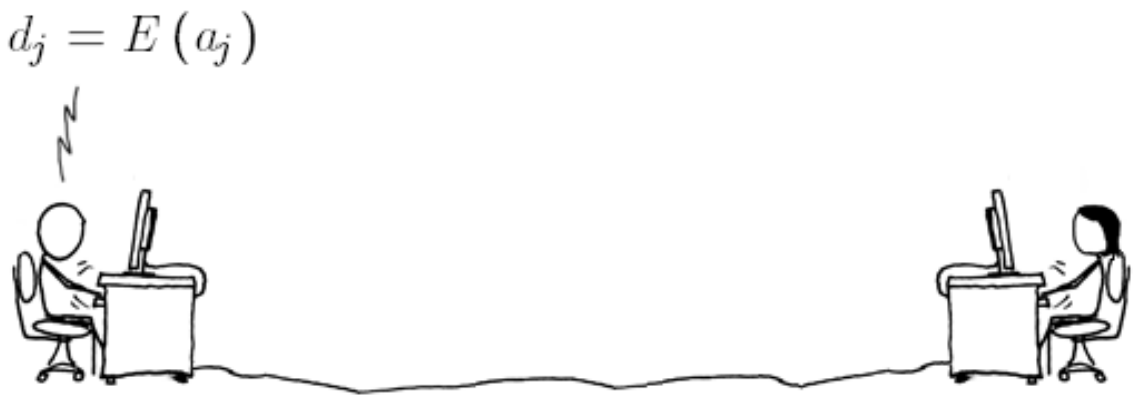
Neuroni di uscita

Per ogni neurone di uscita j:

- Bob computa il valore: $d_j = E(a_j)$, inviandolo poi ad Alice
- Alice ricava il valore di uscita del singolo nodo come: $z_j = g(D(d_j))$

Nota: I valori così ottenuti rappresentano il risultato della computazione tramite la rete neurale di Bob sui dati forniti in ingresso da Alice, avvenuta:

- Oscurando la rete neurale, preservandone la struttura interna
- Garantendo riservatezza per i dati di Alice



Il Protocollo

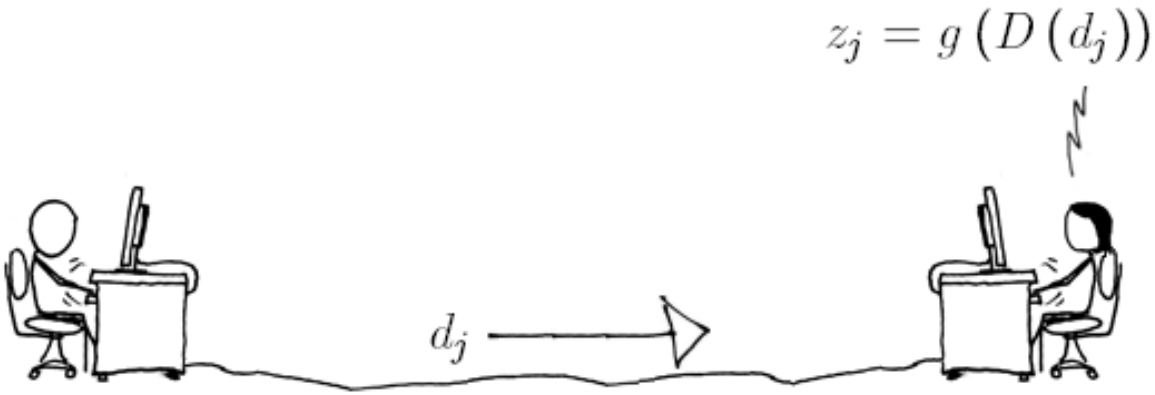
Neuroni di uscita

Per ogni neurone di uscita j:

- Bob computa il valore: $d_j = E(a_j)$, inviandolo poi ad Alice
- Alice ricava il valore di uscita del singolo nodo come: $z_j = g(D(d_j))$

Nota: I valori così ottenuti rappresentano il risultato della computazione tramite la rete neurale di Bob sui dati forniti in ingresso da Alice, avvenuta:

- Oscurando la rete neurale, preservandone la struttura interna
- Garantendo riservatezza per i dati di Alice



Il Protocollo

Neuroni di uscita

Per ogni neurone di uscita j:

- Bob computa il valore: $d_j = E(a_j)$, inviandolo poi ad Alice
- Alice ricava il valore di uscita del singolo nodo come: $z_j = g(D(d_j))$

Nota: I valori così ottenuti rappresentano il risultato della computazione tramite la rete neurale di Bob sui dati forniti in ingresso da Alice, avvenuta:

- Oscurando la rete neurale, preservandone la struttura interna
- Garantendo riservatezza per i dati di Alice

NNSec

Il software realizzato:

- Implementa praticamente il protocollo proposto
- Realizza un classificatore per dati cifrati

Inoltre, propone caratteristiche aggiuntive fra le quali:

Domanda. . .

- Parser integrato, linguaggio specifico
- Comunicazione fra le parti coinvolte
- Interazione multi-utente, accesso concorrente

. . . E risposta

NNSec

Il software realizzato:

- Implementa praticamente il protocollo proposto
- Realizza un classificatore per dati cifrati

Inoltre, propone caratteristiche aggiuntive fra le quali:

Domanda. . .

- Parser integrato, linguaggio specifico
- Comunicazione fra le parti coinvolte
- Interazione multi-utente, accesso concorrente

. . . E risposta

- Supporto attraverso:
  - **JFlex**
  - **JavaCUP**
- Realizzazione di un analizzatore sintattico/lessicale
- Ideazione di un linguaggio ad-hoc per la descrizione di reti neurali

NNSec

Il software realizzato:

- Implementa praticamente il protocollo proposto
- Realizza un classificatore per dati cifrati

Inoltre, propone caratteristiche aggiuntive fra le quali:

Domanda. . .

- Parser integrato, linguaggio specifico
- Comunicazione fra le parti coinvolte
- Interazione multi-utente, accesso concorrente

. . . E risposta

- Implementazione di un modello distribuito client-server
- Uso della tecnologia **RMI** (Remote Method Invocation)
- Comunicazione basata su messaggi scambiati fra oggetti

NNSec

Il software realizzato:

- Implementa praticamente il protocollo proposto
- Realizza un classificatore per dati cifrati

Inoltre, propone caratteristiche aggiuntive fra le quali:

Domanda. . .

- Parser integrato, linguaggio specifico
- Comunicazione fra le parti coinvolte
- Interazione multi-utente, accesso concorrente

. . . E risposta

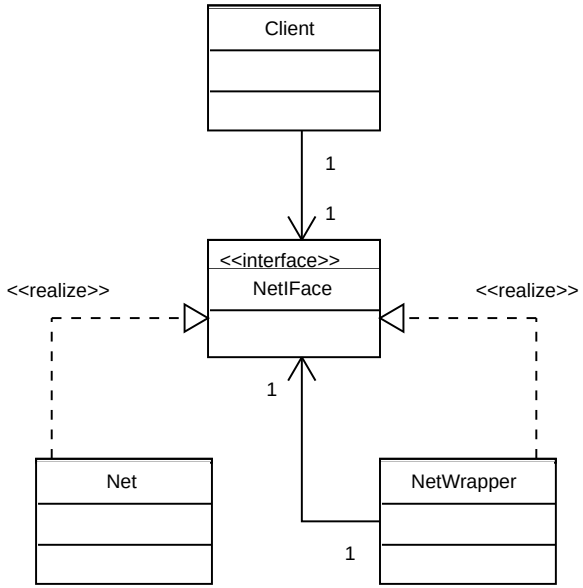
- Strutture dati e algoritmi provenienti dalla teoria dei sistemi operativi
- Pattern di progettazione presi in prestito dalle tecniche di ingegneria del software
- Uso degli strumenti messi a disposizione dal linguaggio

In `NNSec` le reti neurali sono gestite attraverso generici modelli unici.

Pattern Composite

Espansione e permutazione delle reti neurali:

- Espansione durante la fase di composizione
- Permutazione prima di ogni richiesta d'uso
- Interfaccia di base unica per il client



Gestore delle Reti Neurali

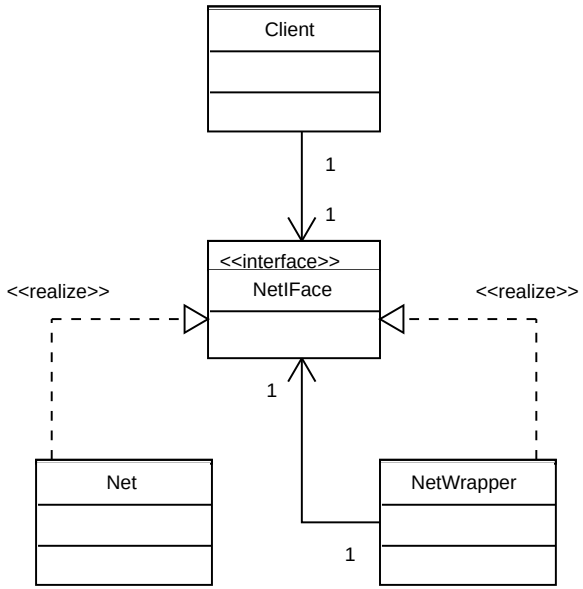
- Inserisce ogni rete neurale in un involucro che la espande
- Associa ad ogni rete neurale un semaforo che ne regola l'accesso concorrente
- Si preoccupa di forzare la permutazione dei neuroni
- Gestisce il recupero delle informazioni e le richieste d'uso

In `NNSec` le reti neurali sono gestite attraverso generici modelli unici.

Pattern Composite

Espansione e permutazione delle reti neurali:

- Espansione durante la fase di composizione
- Permutazione prima di ogni richiesta d'uso
- Interfaccia di base unica per il client



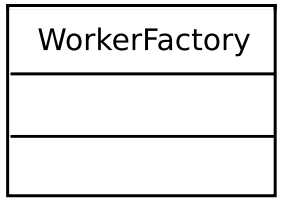
Gestore delle Reti Neurali

- Inserisce ogni rete neurale in un involucro che la espande
- Associa ad ogni rete neurale un semaforo che ne regola l'accesso concorrente
- Si preoccupa di forzare la permutazione dei neuroni
- Gestisce il recupero delle informazioni e le richieste d'uso

I Quattro Moschettieri

Il cuore di `NNSec`, oltre che dal gestore delle reti neurali, comprende:

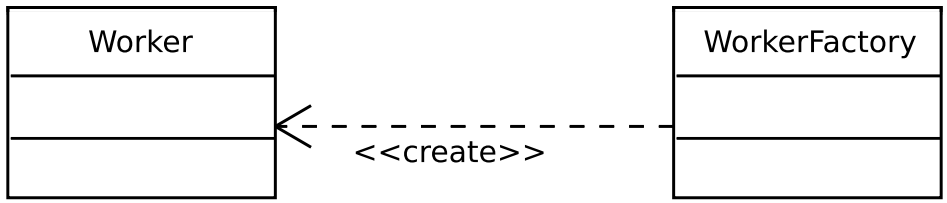
- Factory remota: Risponde alle necessità di interazione
- Lavoratori: Servono richieste diverse in modo indipendente e concorrente
- Modulo di comunicazione: Impostazione d'ambiente, inoltro di richieste
- Calcolatore: Risolve il problema del riferimento circolare



I Quattro Moschettieri

Il cuore di `NNSec`, oltre che dal gestore delle reti neurali, comprende:

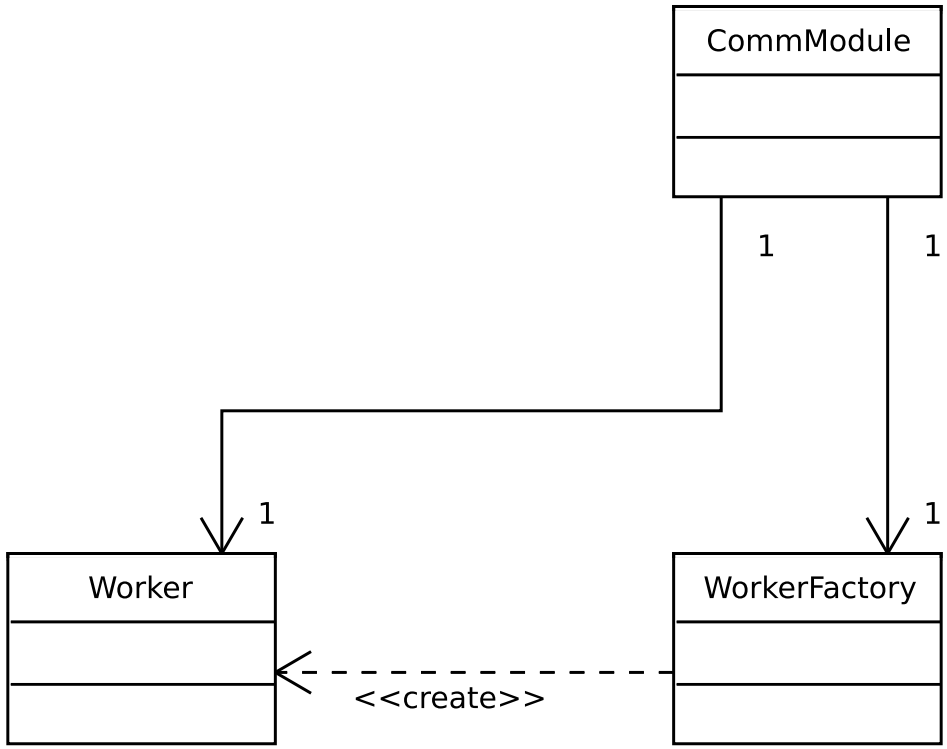
- Factory remota: Risponde alle necessità di interazione
- Lavoratori: Servono richieste diverse in modo indipendente e concorrente
- Modulo di comunicazione: Impostazione d'ambiente, inoltro di richieste
- Calcolatore: Risolve il problema del riferimento circolare



I Quattro Moschettieri

Il cuore di `NNSec`, oltre che dal gestore delle reti neurali, comprende:

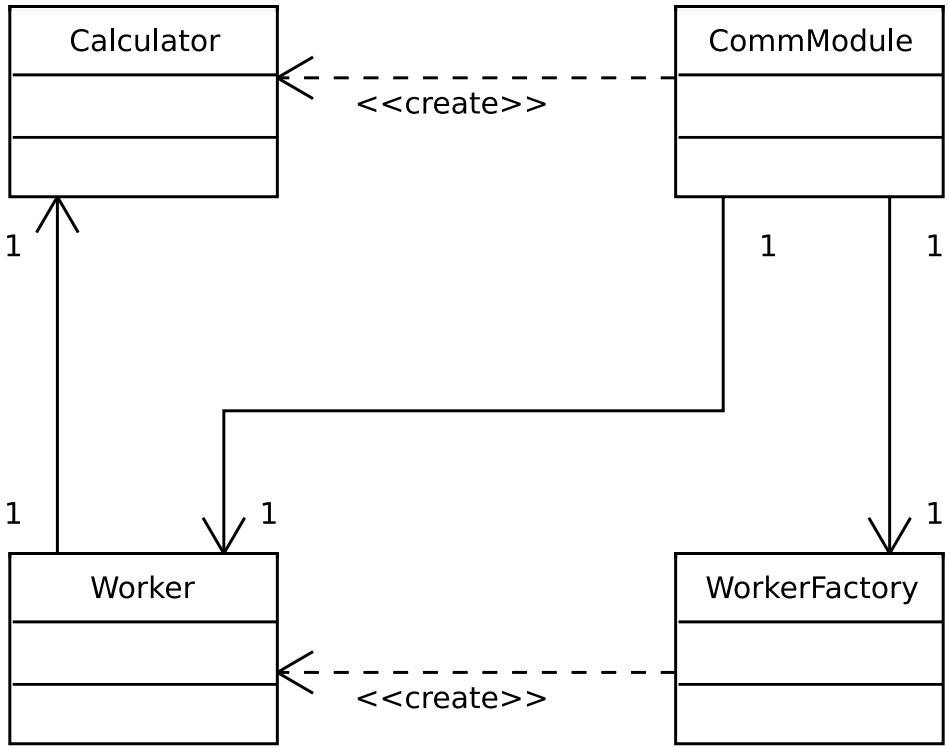
- Factory remota: Risponde alle necessità di interazione
- Lavoratori: Servono richieste diverse in modo indipendente e concorrente
- Modulo di comunicazione: Impostazione d'ambiente, inoltro di richieste
- Calcolatore: Risolve il problema del riferimento circolare



I Quattro Moschettieri

Il cuore di `NNSec`, oltre che dal gestore delle reti neurali, comprende:

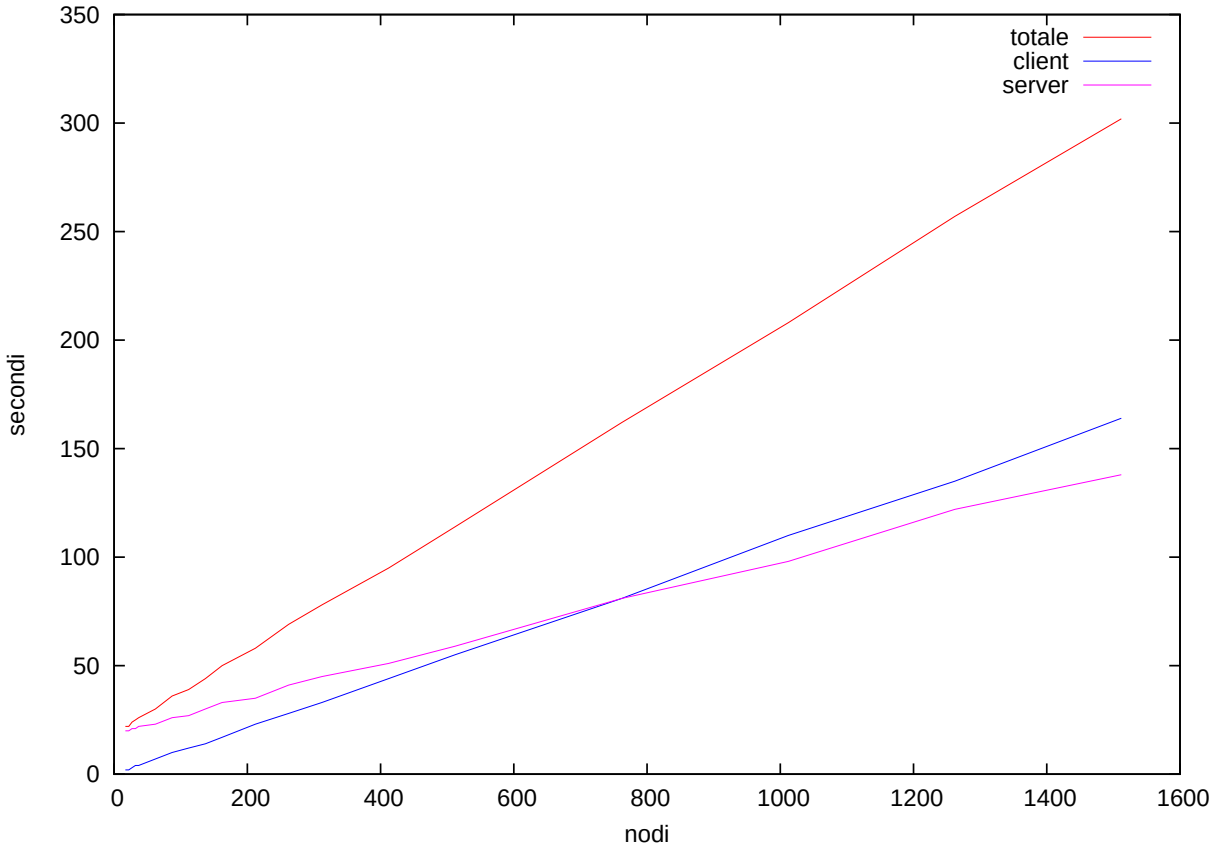
- Factory remota: Risponde alle necessità di interazione
- Lavoratori: Servono richieste diverse in modo indipendente e concorrente
- Modulo di comunicazione: Impostazione d'ambiente, inoltro di richieste
- Calcolatore: Risolve il problema del riferimento circolare



I Quattro Moschettieri

Il cuore di `NNSec`, oltre che dal gestore delle reti neurali, comprende:

- Factory remota: Risponde alle necessità di interazione
- Lavoratori: Servono richieste diverse in modo indipendente e concorrente
- Modulo di comunicazione: Impostazione d'ambiente, inoltro di richieste
- Calcolatore: Risolve il problema del riferimento circolare



Test

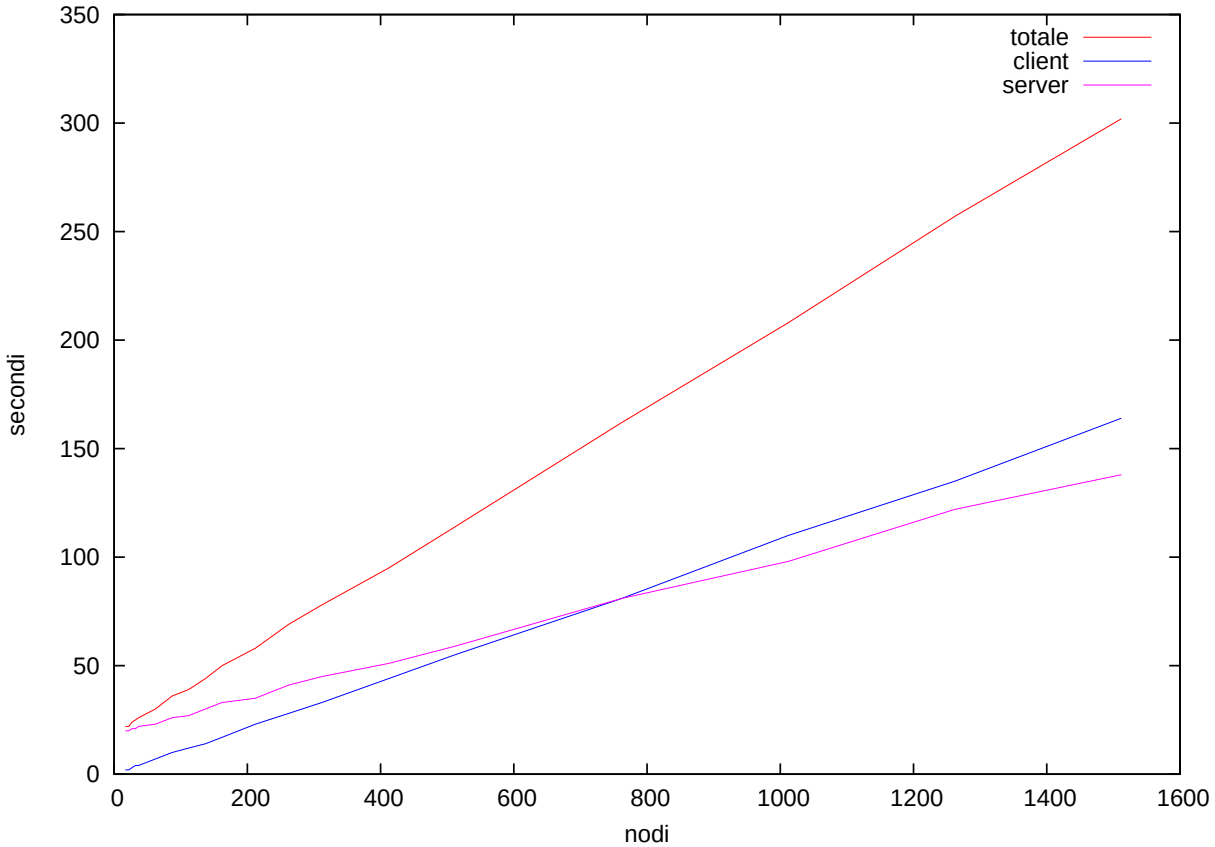
Preparazione:

- Rete neurale (overfitting sui dati)
- Numero neuroni intermedi variabile
- Chiave di lunghezza 1024 bit
- Processore Quad Core (2.40GHz) e 4Gb RAM

Risultati

I risultati ottenti determinano:

- Crescita lineare in base al numero di neuroni intermedi
- Distanza di un primo di taglio con un'interpolazione di ordine superiore
- Degenerazione più complessità lato client



Test

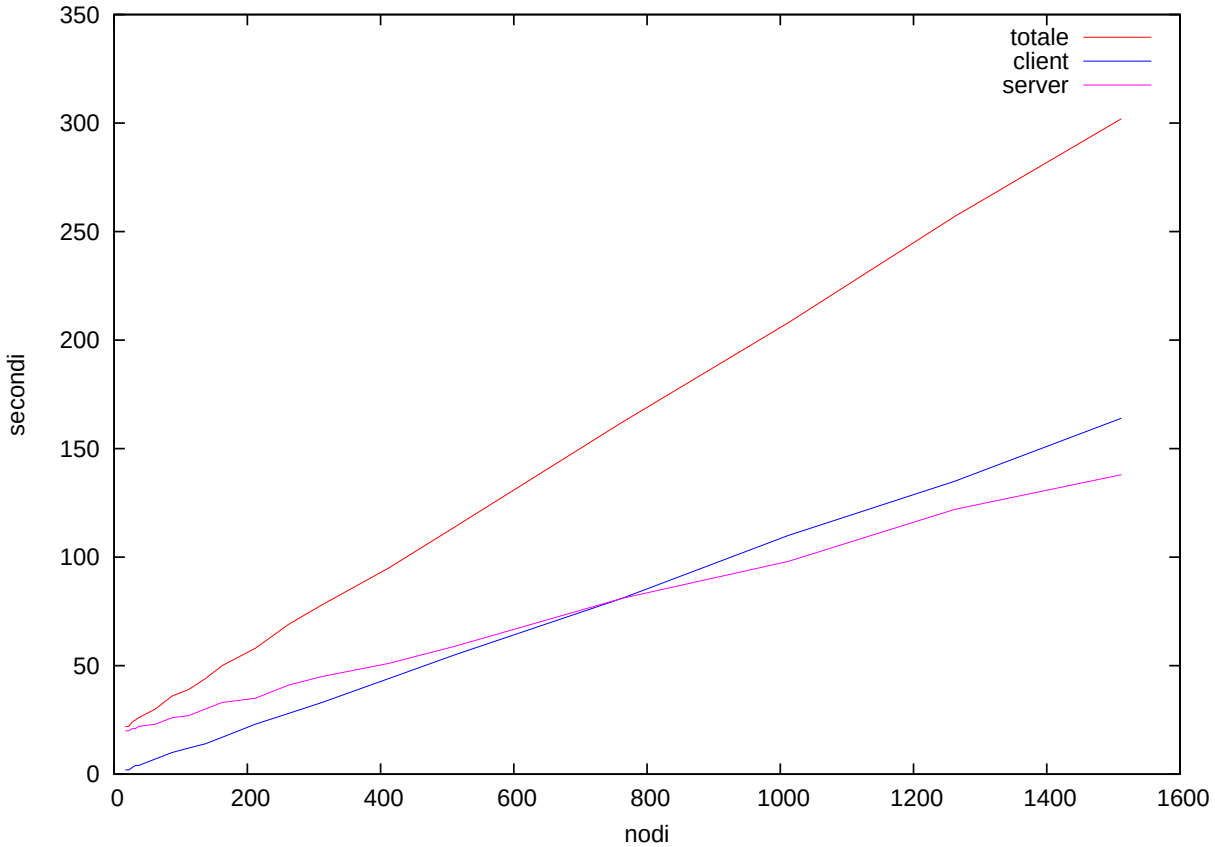
Preparazione:

- Rete neurale (overfitting sui dati)
- Numero neuroni intermedi variabile
- Chiave di lunghezza 1024 bit
- Processore Quad Core (2.40GHz) e 4Gb RAM

Risultati

I risultati ottenti determinano:

- Crescita lineare in base al numero di neuroni intermedi
- Esistenza di un punto di taglio con uguale distribuzione del carico di lavoro
- Degenerazione più consistente lato client



Test

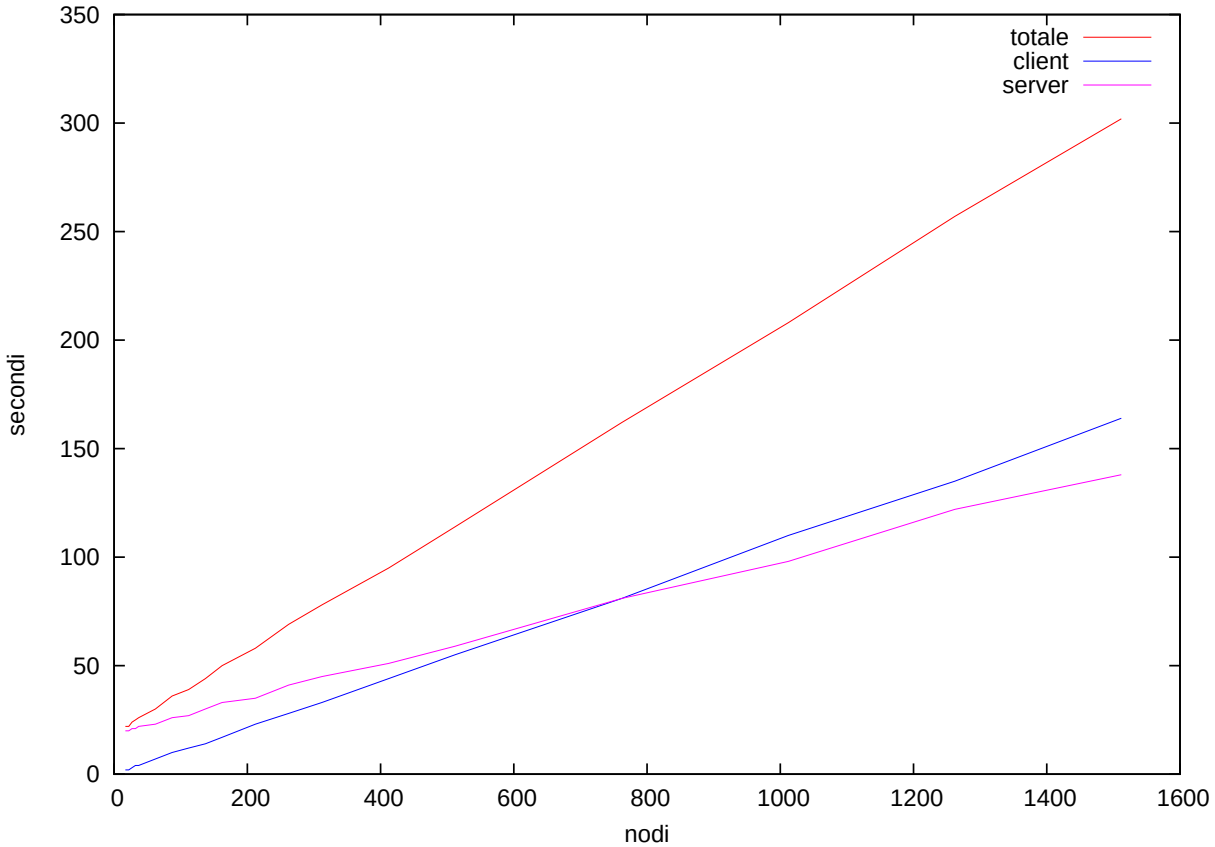
Preparazione:

- Rete neurale (overfitting sui dati)
- Numero neuroni intermedi variabile
- Chiave di lunghezza 1024 bit
- Processore Quad Core (2.40GHz) e 4Gb RAM

Risultati

I risultati ottenti determinano:

- Crescita lineare in base al numero di neuroni intermedi
- Esistenza di un punto di taglio con uguale distribuzione del carico di lavoro
- Degenerazione più consistente lato client



Test

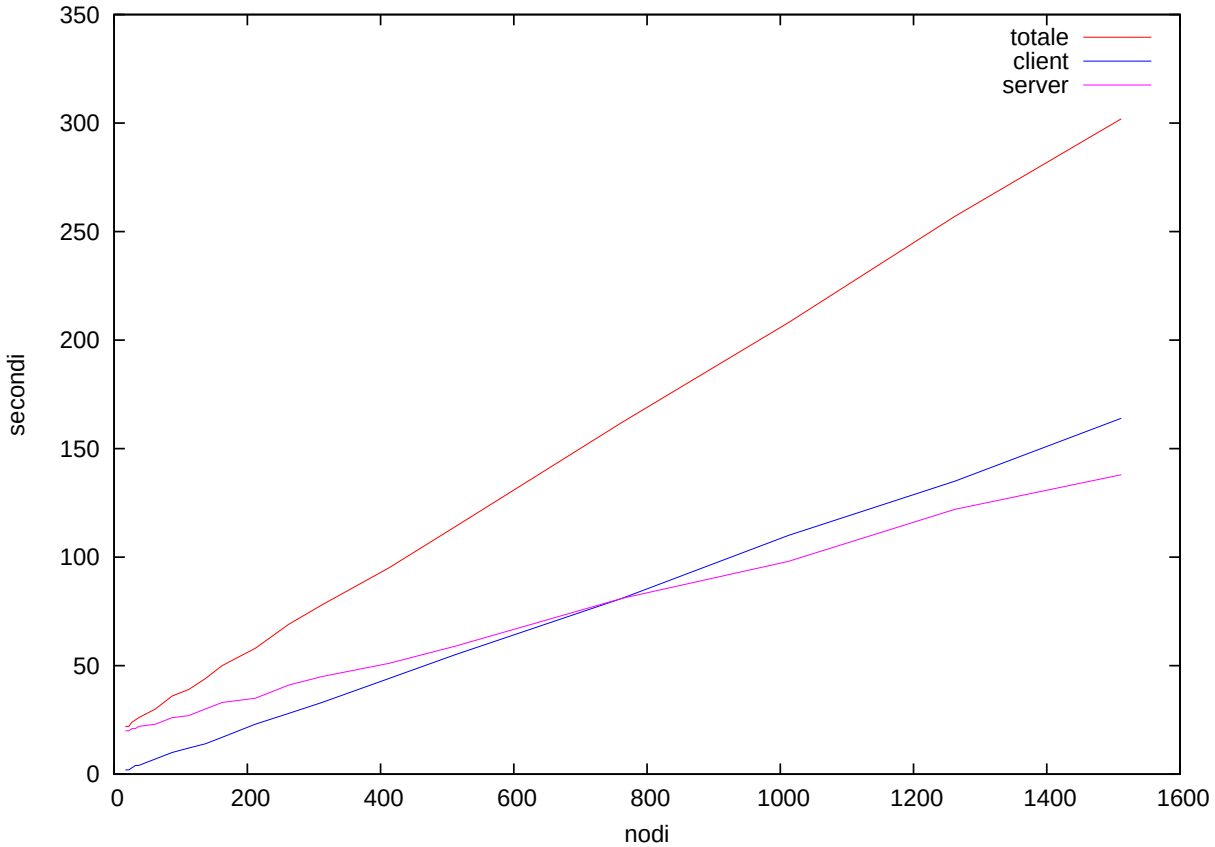
Preparazione:

- Rete neurale (overfitting sui dati)
- Numero neuroni intermedi variabile
- Chiave di lunghezza 1024 bit
- Processore Quad Core (2.40GHz) e 4Gb RAM

Risultati

I risultati ottenti determinano:

- Crescita lineare in base al numero di neuroni intermedi
- Esistenza di un punto di taglio con uguale distribuzione del carico di lavoro
- Degenerazione più consistente lato client



Test

Preparazione:

- Rete neurale (overfitting sui dati)
- Numero neuroni intermedi variabile
- Chiave di lunghezza 1024 bit
- Processore Quad Core (2.40GHz) e 4Gb RAM

Risultati

I risultati ottenti determinano:

- Crescita lineare in base al numero di neuroni intermedi
- Esistenza di un punto di taglio con uguale distribuzione del carico di lavoro
- Degenerazione più consistente lato client

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .

Approfondimenti

Un aspetto in particolare merita di essere approfondito: la **degenerazione**

Motivazioni possibili della degenerazione:

Lato Server

- Costo dovuto ad operazioni di moltiplicazione e potenze
- Numero di operazioni superiore. . .
- . . . Ma di complessità inferiore

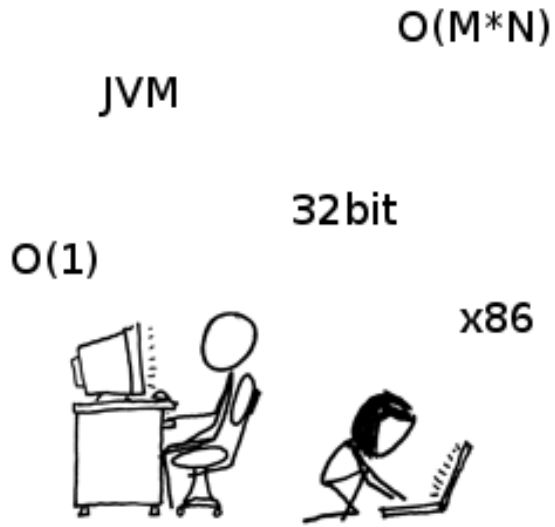
Lato Client

- Costo legato principalmente alle operazioni di decifratura/cifratura
- Numero di operazioni inferiore. . .
- . . . Ma di complessità superiore

Fattori

Alcuni dei fattori in gioco sono:

- Macchina virtuale (Java Virtual Machine)
- Costo in termini di operazioni macchina
- Architettura degli elaboratori
- Complessità di cifratura/decifratura
- . . .



Riassumendo

- L'obiettivo è quello di:
  - Avvicinare strumenti di classificazione e tecniche di crittografia
  - Realizzare un classificatore per dati cifrati
- La base di partenza:
  - Interessanti proprietà omomorfiche del cifrario
  - Un protocollo che sfrutti tali caratteristiche a suo favore
- Il risultato finale:
  - Un software sviluppato in Java che implementa il protocollo proposto
  - Multi-utenza, accesso concorrente alle risorse
- Le prove sperimentali hanno rivelato infine:
  - Prestazioni accettabili in ogni caso su hardware non datato e in genere per reti neurali con un numero di neuroni intermedi contenuto
  - Possibile previsione del comportamento in base al numero di neuroni
  - Applicabilità possibile (apparentemente) a scenari reali

Riassumendo

- L'obiettivo è quello di:
  - Avvicinare strumenti di classificazione e tecniche di crittografia
  - Realizzare un classificatore per dati cifrati
- La base di partenza:
  - Interessanti proprietà omomorfiche del cifrario
  - Un protocollo che sfrutti tali caratteristiche a suo favore
- Il risultato finale:
  - Un software sviluppato in Java che implementa il protocollo proposto
  - Multi-utenza, accesso concorrente alle risorse
- Le prove sperimentali hanno rivelato infine:
  - Prestazioni accettabili in ogni caso su hardware non datato e in genere per reti neurali con un numero di neuroni intermedi contenuto
  - Possibile previsione del comportamento in base al numero di neuroni
  - Applicabilità possibile (apparentemente) a scenari reali

Riassumendo

- L'obiettivo è quello di:
  - Avvicinare strumenti di classificazione e tecniche di crittografia
  - Realizzare un classificatore per dati cifrati
- La base di partenza:
  - Interessanti proprietà omomorfiche del cifrario
  - Un protocollo che sfrutti tali caratteristiche a suo favore
- Il risultato finale:
  - Un software sviluppato in Java che implementa il protocollo proposto
  - Multi-utenza, accesso concorrente alle risorse
- Le prove sperimentali hanno rivelato infine:
  - Prestazioni accettabili in ogni caso su hardware non datato e in genere per reti neurali con un numero di neuroni intermedi contenuto
  - Possibile previsione del comportamento in base al numero di neuroni
  - Applicabilità possibile (apparentemente) a scenari reali

Riassumendo

- L'obiettivo è quello di:
  - Avvicinare strumenti di classificazione e tecniche di crittografia
  - Realizzare un classificatore per dati cifrati
- La base di partenza:
  - Interessanti proprietà omomorfiche del cifrario
  - Un protocollo che sfrutti tali caratteristiche a suo favore
- Il risultato finale:
  - Un software sviluppato in Java che implementa il protocollo proposto
  - Multi-utenza, accesso concorrente alle risorse
- Le prove sperimentali hanno rivelato infine:
  - Prestazioni accettabili in ogni caso su hardware non datato e in genere per reti neurali con un numero di neuroni intermedi contenuto
  - Possibile previsione del comportamento in base al numero di neuroni
  - Applicabilità possibile (apparentemente) a scenari reali

Riassumendo

- L'obiettivo è quello di:
  - Avvicinare strumenti di classificazione e tecniche di crittografia
  - Realizzare un classificatore per dati cifrati
- La base di partenza:
  - Interessanti proprietà omomorfiche del cifrario
  - Un protocollo che sfrutti tali caratteristiche a suo favore
- Il risultato finale:
  - Un software sviluppato in Java che implementa il protocollo proposto
  - Multi-utenza, accesso concorrente alle risorse
- Le prove sperimentali hanno rivelato infine:
  - Prestazioni accettabili in ogni caso su hardware non datato e in genere per reti neurali con un numero di neuroni intermedi contenuto
  - Possibile previsione del comportamento in base al numero di neuroni
  - Applicabilità possibile (apparentemente) a scenari reali