

Generazione procedurale di contenuti per videogiochi

Renato Mainetti, Jacopo Essenziale, Riccardo Cantoni, Manuel Pezzera, Alessandro Tironi, N. Alberto Borghese

{renato.mainetti, jacopo.essenziale, manuel.pezzera, alessandro.tironi, alberto.borghese} @unimi.it

*Applied Intelligent Systems Laboratory - AISLab - Dip. Di Informatica,
Università degli Studi di Milano*

Perché ?

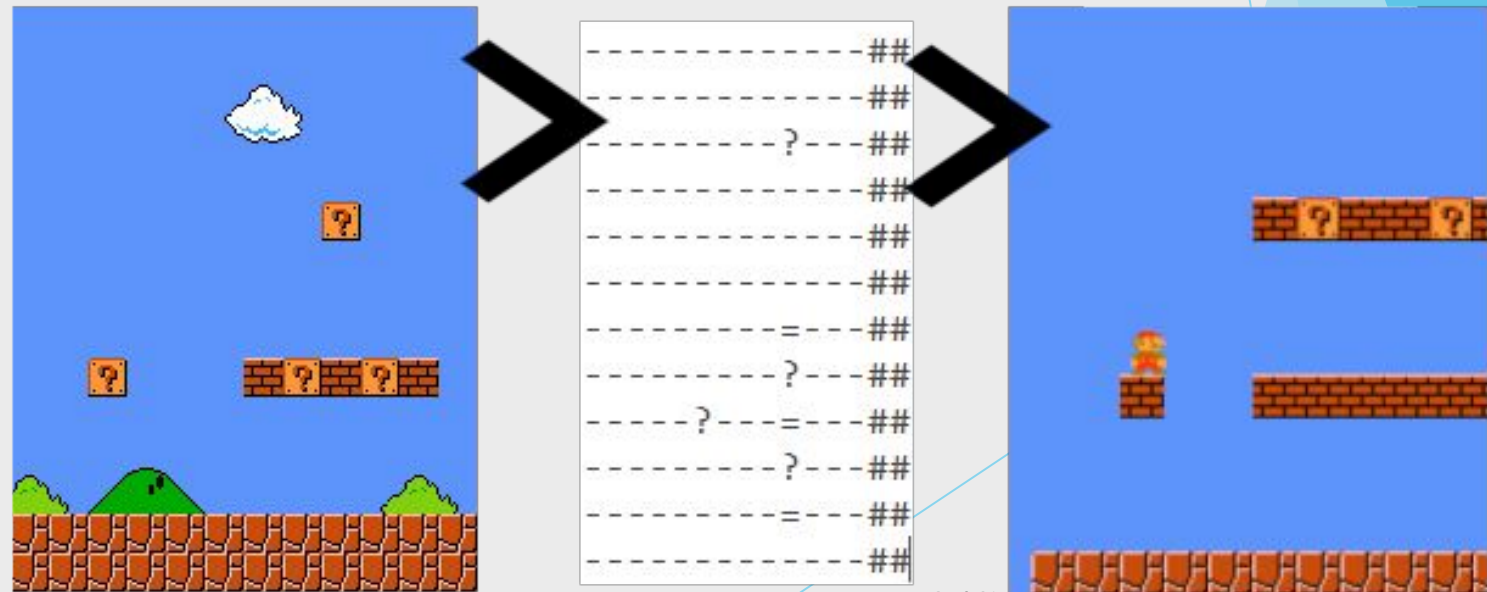
- ▶ In Italia nel 2018 il giro di affari videoludico è stato di quasi 1,5 miliardi di € (fonte AESVI), confermando ulteriormente l'andamento positivo di crescita del settore.
- ▶ L'industria videoludica riesce a produrre ogni anno un limitato numero di titoli di gioco, colmando solo in parte la crescente richiesta.
- ▶ La produzione di titoli di successo richiede infatti a professionisti la creazione di svariate tipologie di contenuti di alta qualità, incidendo così sui costi.
- ▶ Un aiuto può essere fornito da strumenti automatici, capaci di aiutare i professionisti, riducendo i tempi necessari alla produzione di contenuti digitali.

Cosa stiamo sperimentando:

Tecniche basate su IA per la generazione procedurale di contenuti necessari alla realizzazione di un videogioco (game design, 3D world, behaviour, narrative)

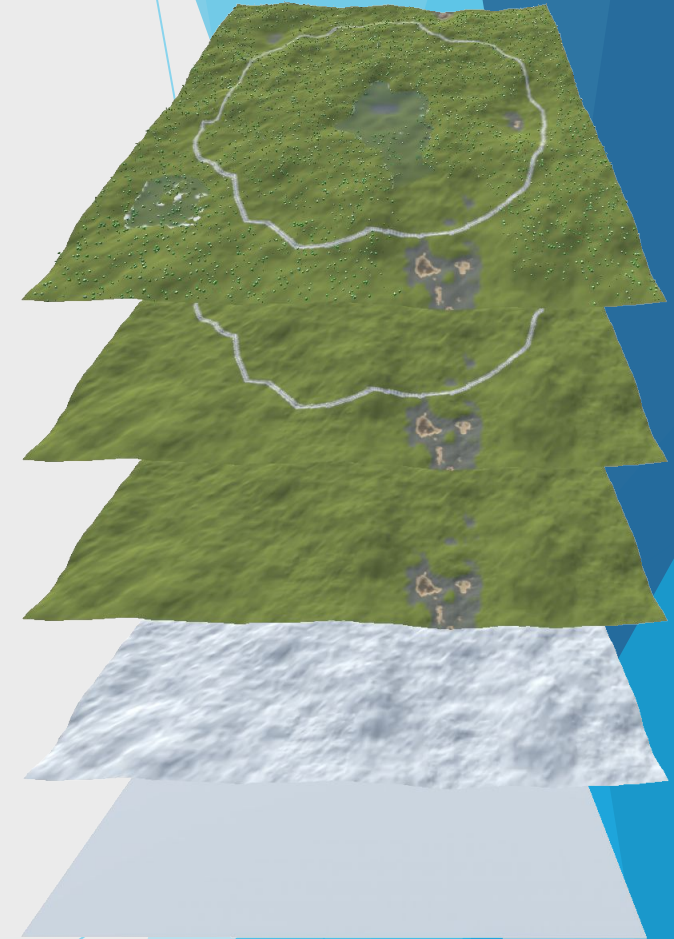
Level design basato su Reti Neurali Ricorrenti generative

- ▶ Abbiamo addestrato una Rete Neurale Ricorrente utilizzando come dataset 16 livelli di Super Mario Bros. codificati in testi i cui caratteri rappresentano i singoli tile del livello e definiscono implicitamente la grammatica seguita dal level designer.
- ▶ La rete addestrata si è dimostrata in grado di generare dei testi basati sulla grammatica appresa che, una volta decodificati in livelli di gioco, si sono dimostrati la maggior parte delle volte giocabili e divertenti.
- ▶ Un possibile sviluppo consiste nel realizzare un tool di validazione di giocabilità dei livelli generati.



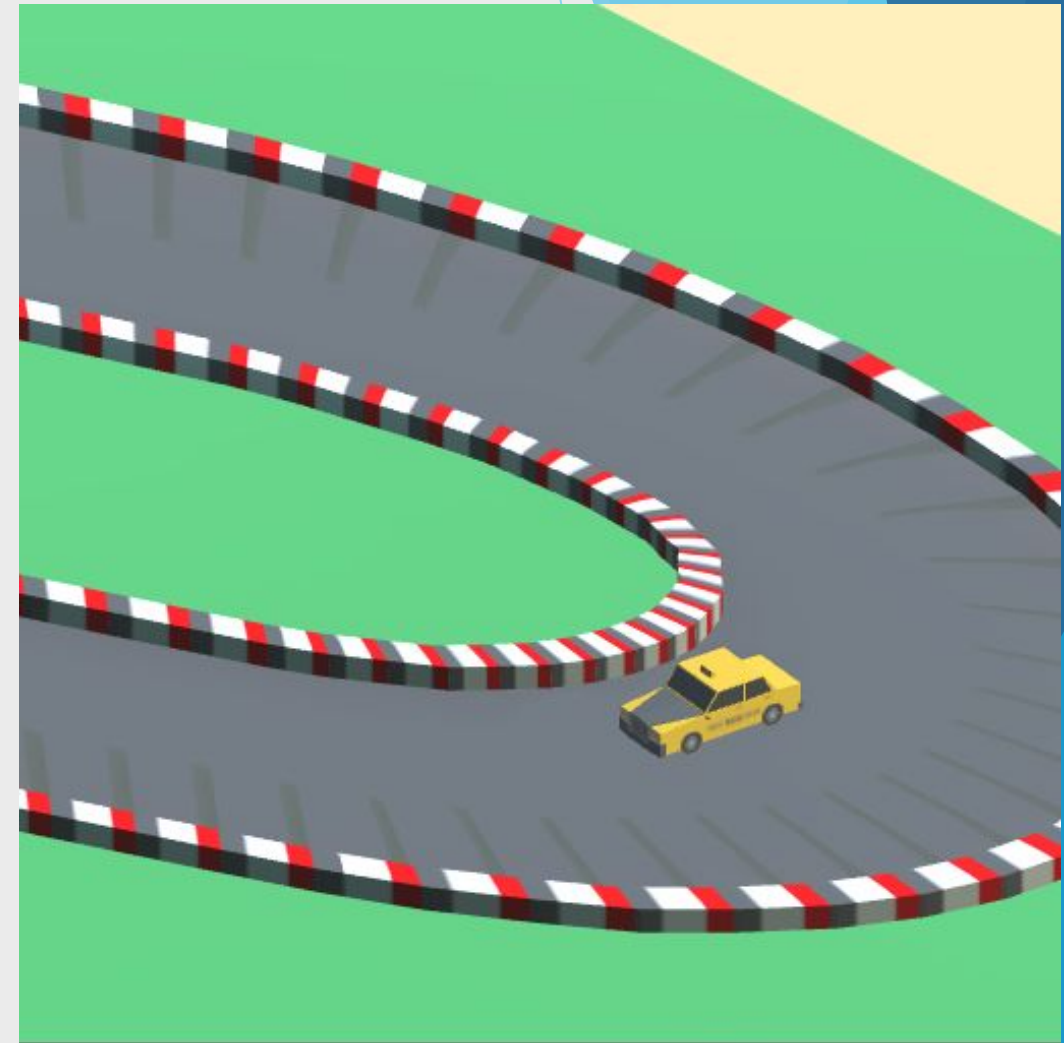
Generazione procedurale di scenari virtuali

- ▶ La generazione del mondo di gioco è tra i temi più trattati nella PCG: un mondo generato proceduralmente in ogni partita permette di ottenere un maggiore grado di varietà.
- ▶ Attraverso il Perlin noise generiamo un terreno realistico a cui applichiamo poi diverse texture (e.g., erba, neve, sabbia) in base alla topologia dell'ambiente.
- ▶ Successivamente aggiungiamo altri elementi decorativi come alberi, rocce, laghi, strade.
- ▶ La generazione procedurale può essere ulteriormente migliorata generando contenuti più complessi e maggiormente caratteristici, come città o fiumi, oppure attraverso la generazione di terreni con diversi biomi, per ottenere un mondo maggiormente realistico.



Procedural Behaviours con ML-Agents

- ▶ Impiegando tecniche di deep reinforcement learning è stato possibile addestrare un agente alla guida autonoma, capace di seguire la traiettoria ideale ed evitare collisioni.
- ▶ Sfruttando l'implementazione fornita dal package ML-Agents di Unity 3D, abbiamo addestrato un agente a guidare un'automobile all'interno di un circuito.
- ▶ Lo strumento ha permesso la generazione e conseguente integrazione, con uno sforzo minimo, di comportamenti intelligenti credibili all'interno di un videogioco.



Generazione Procedurale di Narrativa

- ▶ L'introduzione di una narrativa all'interno di videogiochi aiuta a migliorare il coinvolgimento e l'immersività dell'esperienza videoludica per l'utente.
- ▶ Abbiamo esplorato il dominio della “automatic narration” sviluppando un tool che, alimentato da una base di conoscenza contenente materiale narrativo, fosse in grado di produrre strutture narrative sempre nuove.
- ▶ Lo strumento utilizza elementi ereditati dai modelli di Propp e della Fabula per formalizzare i componenti della storia e introdurli in strutture dati che rappresentano le proprietà di un determinato evento della storia.
- ▶ Lo strumento permette di impostare dei vincoli nelle storie forzando la presenza di determinati eventi nel percorso generato.

Grazie per l'attenzione

Renato Mainetti, Jacopo Essenziale, Riccardo Cantoni, Manuel Pezzera, Alessandro Tironi, N. Alberto Borghese

{renato.mainetti, jacopo.essenziale, manuel.pezzera, alessandro.tironi, alberto.borghese} @unimi.it

*Applied Intelligent Systems Laboratory,
University of Milan, Department of Computer Science
Via Giovanni Celoria 18, 20133 Milan (MI)*