Machine Learning, “Deep Fake” ed i rischi in un mondo iperconnesso

A cura di: Andrea Pasquinucci 15 Giugno 2022

Negli ultimi anni l’importanza nella nostra società dell’informazione condivisa spesso in tempo reale, è cresciuta enormemente e con essa i rischi derivanti dalla diffusione di notizie incorrette e dalla manipolazione delle notizie. Ormai viviamo in un mondo iperconnesso ove le informazioni su quanto avviene sono condivise in tempo reale non solo tramite le agenzie di informazione e i giornali ma anche, e spesso soprattutto, tramite il passa parola, o meglio la condivisione sui Social Network.

Come esempi basta fare riferimento (purtroppo) ad eventi come la pandemia Covid-19 o la guerra in Ucraina per convincersi che siamo subissati da informazioni alle volte contrastanti e di cui è difficile capire la sorgente e l’affidabilità.

Lo scenario attuale è quindi quello in cui la condivisione delle informazioni è principalmente svolta tramite strumenti informatici che permettono al contempo sia una condivisione immediata e capillare sia una condivisione a livello globale (mondiale). Ormai la condivisione delle informazioni direttamente tra persone o cartacea,

preminente fino a una ventina di anni fa insieme a radio e televisione, si è ridotta notevolmente almeno rispetto ai canali informatici.

In questo scenario, come è possibile valutare l’autorevolezza, correttezza e anche liceità delle informazioni che riceviamo e condividiamo? Il problema delle così dette “Fake News” ovvero informazioni false ma a prima vista credibili per come proposte e formulate, è ben noto ed è discusso ampiamente.[1]

Gli algoritmi di Machine Learning (ML) contribuiscono già significativamente a questa problematica ed in questo articolo vogliamo fare una rapida rassegna degli aspetti positivi e negativi dell’utilizzo e dei possibili sviluppi degli algoritmi ML per le “Fake News” e i “Deep Fake”, ovvero la generazione di notizie false sfruttando i più avanzati modelli di ML.

Ricordiamo comunque che oltre ai modelli ML, vi sono molte altre tecnologie digitali che permettono di creare “Fake News” (anche un comune programma per modificare immagini o montare video può essere usato per creare “Fake News”), queste tecnologie sono tipicamente più economiche e facili da usare anche se spesso producono risultati più crudi e facilmente riconoscibili come falsi rispetto ai più avanzati modelli ML.[2]

In questo articolo ci limitiamo a considerare l’utilizzo dei modelli ML per creare “Fake News” e come primo passo dobbiamo ricordare velocemente come i modelli ML possono essere utilizzati in questo ambito.

Il “Classifier”

I primi, più importanti, più comuni e noti modelli di Machine Learning sono chiamati in generale *Classifier* in quanto capaci di riconoscere una informazione nell’input che gli è fornito. Possiamo fare tanti esempi quali:

- riconoscere un oggetto o una persona in un’immagine;
- riconoscere un oggetto od una persona in un video;
- riconoscere la lingua in cui è scritto un testo;
- attribuire un testo ad uno scrittore;
- attribuire una composizione musicale ad un compositore

e così via.

In pratica, qualunque informazione codificabile digitalmente può diventare ambito di un modello ML di tipo *Classifier*.

Il risultato del riconoscimento è sempre probabilistico ed il modello ML impara a distinguere le caratteristiche richieste utilizzando grandi data-set di informazioni già classificate.

In questo articolo facciamo riferimento come primo esempio ad un modello educativo un po’ particolare creato da Amazon e chiamato *Deep Composer* [Rif. 1]. Scopo finale (ed ideale) di questo modello ML è quello di comporre musica indistinguibile da quella di un compositore di riferimento. Il primo passo è quello di creare un modello ML in grado di discriminare le composizioni di un autore (ad esempio Mozart) da quelle di tutti gli altri autori. Il modello è istruito facendogli “sentire” tutte le opere di Mozart e moltissime altre opere musicali di autori diversi e di stili, epoche e tipi musicali diversi, indicandogli quali sono di Mozart e quali invece non lo sono. In Fig.1 è riportata la struttura ad alto livello del modello ML che riporta in output la probabilità che il brano in ingresso sia (in questo caso) di Mozart.

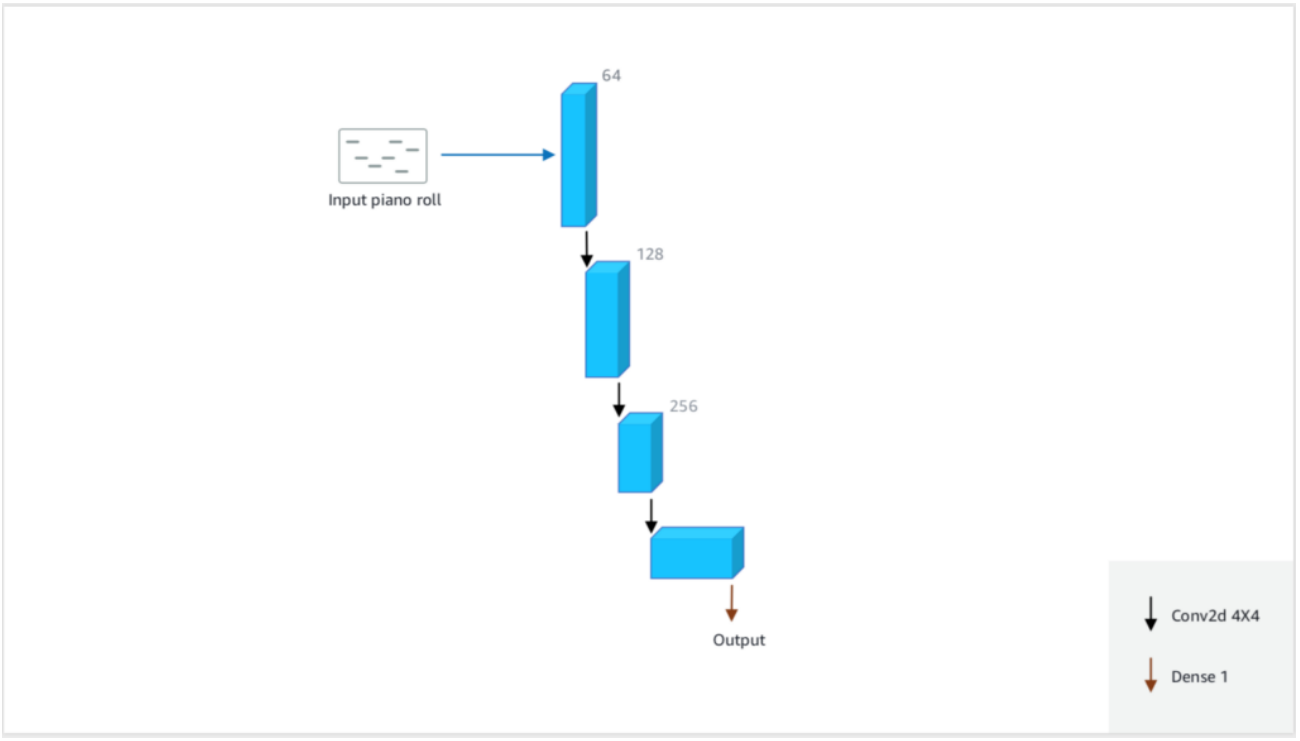

Fig. 1 Struttura di un Classifier/Discriminator di tipo Convolutional Neural Network utilizzato nel Deep Composer [Sorgente Rif. 2]

Utilizzare un “Classifier” per individuare “Fake News”

La maggior parte dei modelli ML sono dei *Classifier*, ovvero in grado di identificare o classificare i dati forniti in input. Questi modelli sono utilizzati ormai quotidianamente in moltissime applicazioni: dalla videosorveglianza, all’analisi degli attacchi informatici, alle campagne pubblicitarie, all’individuazione dello Spam nella posta elettronica ecc. ed anche per individuare “Fake News”.

L’approccio è molto semplice, dovrebbe essere sufficiente addestrare un modello ML con un data-set di informazioni, ad esempio testuali, già classificate come vere o false e poi utilizzare il modello per identificare testi falsi. Questo in realtà è quello che fanno già praticamente tutti i Social Network e tutti i grandi gestori di informazioni sia in forma testuale sia audio, immagini e video. Sono però ben noti sia i falsi positivi (si veda come un ben noto esempio [Rif. 3]) sia i falsi negativi, ovvero informazioni false presenti e disponibili ma non individuate dagli algoritmi ML.

I motivi per la mancanza di un completo successo, almeno come sperato, da parte di questi modelli ha molte cause, che non saranno analizzate in questa sede, ma una è sicuramente rilevante e di facile discussione. L’apprendimento di un modello ML dipende in modo cruciale dai dati utilizzati in questa fase nella quale l’algoritmo impara a individuare sia affermazioni false sia ragionamenti e modi di presentare le informazioni tali da ingannare le persone.

Ora, similmente ad esempio a quanto succede per l’identificazione dello Spam con modelli ML, si instaura un ciclo continuo tra

- chi inventa nuovi modi di formulare informazioni false in modo da non essere rilevato dal modello ML;
- l’identificazione manuale del nuovo attacco;
- la ri-esecuzione del processo di apprendimento del modello ML con un data-set a cui sono stati aggiunti i nuovi tipi di attacco.

A questo proposito va ricordato che l’addestramento di un modello ML è spesso un’operazione molto impegnativa (e costosa) dal punto di vista computazionale, ovvero ad ogni esecuzione può richiedere molte risorse HW/SW e molto tempo.

Il “Generator”

Modelli di Machine Learning possono essere utilizzati per creare nuovi contenuti. La principale idea è quella di creare un modello a molti livelli (da qui il nome “Deep”) che dato un input in un formato (ad esempio la codifica di un brano musicale), lo codifica nel formato interno per poi ritrasformarlo nel formato iniziale, ovvero in un brano musicale. Spesso il dato iniziale in input è puramente casuale e l’elaborazione nel modello ML viene rieseguita molte volte sino ad ottenere il risultato cercato. La struttura più semplice di questi modelli è chiamata *U-Net*.

U-Net è un’architettura che utilizza due *Convolutional Neural Network* e che prende il nome dalla sua forma a U. Il percorso di *downsampling/encoder* costituisce il lato sinistro della U e il percorso di *upsampling/decoder* costituisce la parte destra della U. La particolarità dell’architettura U-Net è che i livelli sul lato sinistro possono passare informazioni al lato destro senza attraversare l’intera rete ma saltando direttamente al livello corrispondente (Fig. 2).

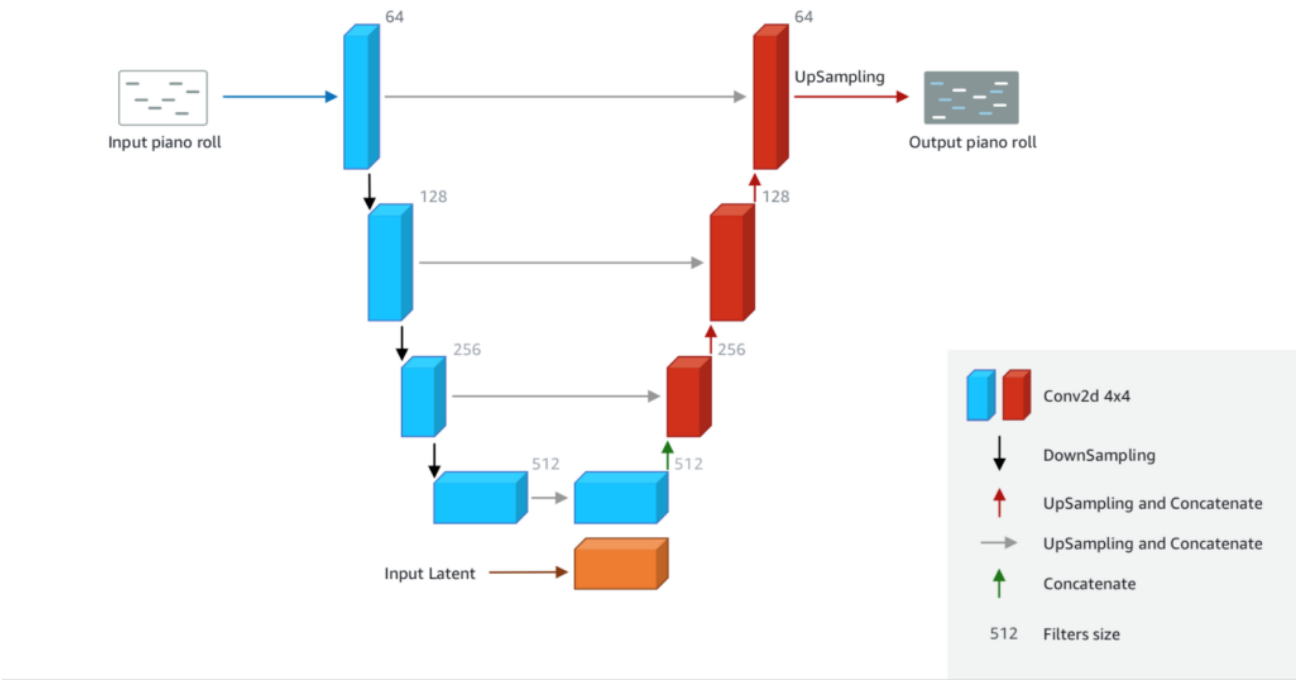

Fig. 2 Struttura di un U-Net composto da Convolutional Neural Network utilizzato come Generator nel Deep Composer [Sorgente Rif. 2]

Ovviamente, è necessario prima istruire un *Generator* in modo tale che sia in grado di produrre quanto ci aspettiamo.

Un “Generative Adversial Network” (GAN)

Il processo di istruzione di un *Generator* non è semplice, l’approccio più comune è quello di accoppiare un *Generator* ad un *Classifier* già opportunamente istruito. Un *Classifier* opportunamente accoppiato ad un *Generator* viene usualmente chiamato un *Discriminator*, e l’accoppiamento di un *Generator* con un *Discriminator* è chiamato un *Generative Adversial Network* – GAN (Fig. 3).

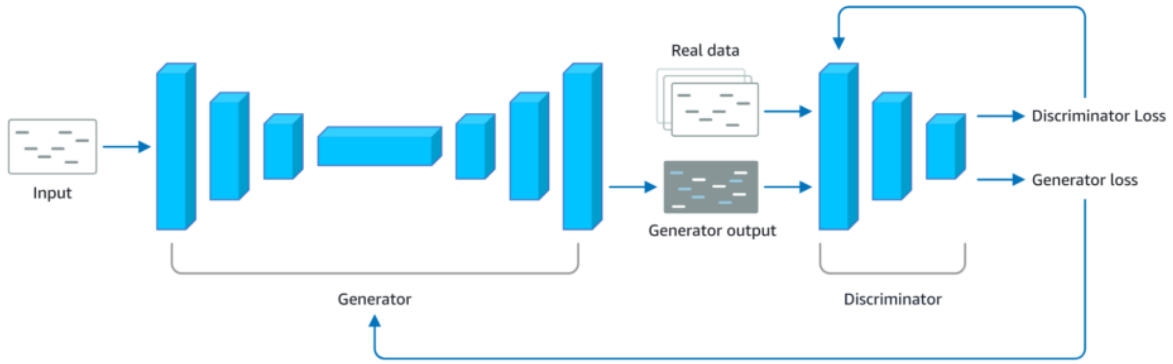

Fig. 3 Struttura di un GAN [Sorgente Rif. 2]

Il processo di istruzione di un *Generator* è, ad alto livello, semplice:

1. Si istruisce per primo il *Discriminator* con dati reali in modo che possa distinguere informazioni vere da “Fake”;
2. Si accoppiano il *Generator* ed il *Discriminator;*
3. Il *Generator* produce dei dati, il *Discriminator* verifica quanto reali siano e riporta al *Generator* il risultato in modo che questo possa apprendere e migliorare la produzione.

Il ciclo viene ripetuto sino a quando il *Generator* riesce a produrre dati classificati come reali dal *Discriminator*. Nell’esempio del *Deep Composer* che genera nuovi brani di Mozart, il ciclo di apprendimento continua sino a quando il *Discriminator* non ritiene che i brani prodotti dal *Generator* siano con un’ottima probabilità in realtà originali di Mozart.

Ovviamente il processo è molto più lungo e complesso di quanto traspare da questa veloce descrizione, richiede tipicamente moltissime risorse, grandi data-set ed è alle volte soggetto a problemi di convergenza o di collasso dei risultati (ovvero in grado di produrre sempre e solo gli stessi dati o le stesse modifiche ai dati ingresso).

Generare “Deep Fake”

Con un *Generator* opportunamente istruito, generare “Deep Fake” diventa abbastanza facile ed alla portata di molti, od almeno così sembrerebbe essere. Come già accennato, con l’opportuno *Generator* si possono creare in grandi quantità testi falsi ma del tutto convincenti (inclusi anche articoli scientifici) con i quali ad esempio è possibile supportare delle campagne di falsa-informazione; generare immagini e filmati realistici ma ovviamente falsi; modificare, anche in minima parte, immagini o filmati o audio in modo che questi descrivano qualcosa che non è avvenuto o il contrario della realtà. Di questo ultimo gruppo fanno parte anche programmi ormai quasi di uso quotidiano che ad esempio permettono di sostituire un viso in un’immagine o in un video.

Nella maggior parte dei casi, soprattutto per audio e video, i risultati sono divertenti, a prima vista realistici ma ad un’analisi un poco più precisa si rivelano subito come falsi (si veda ad esempio [Rif. 4]). E’ importante perciò considerare quali sono oggi i principali aspetti legati alla:

- generazione di “Deep Fake”;
- individuazione di “Deep Fake”.

Generare un vero “Deep Fake” oggi richiede:

1. conoscenze specialistiche molto approfondite;

2. enormi data-set disponibili per l’istruzione dei modelli;
3. grandi disponibilità di calcolo per la fase di istruzione del modello;
4. tempi sufficientemente lunghi (anche mesi) per la preparazione e istruzione del modello;
5. ed ovviamente fondi per pagare tutto ciò.

I risultati di tali modelli però sono veramente preoccupanti in quanto veramente difficili da distinguere da dati veri. In Fig. 4 riportiamo un esempio preso da [Rif. 5] di un testo generato con GPT-3 in cui le prime 4 righe sono originali della Divina Commedia di Dante mentre le successive righe sono state create da GPT-3 utilizzando come ingresso solo le prime 4 righe:

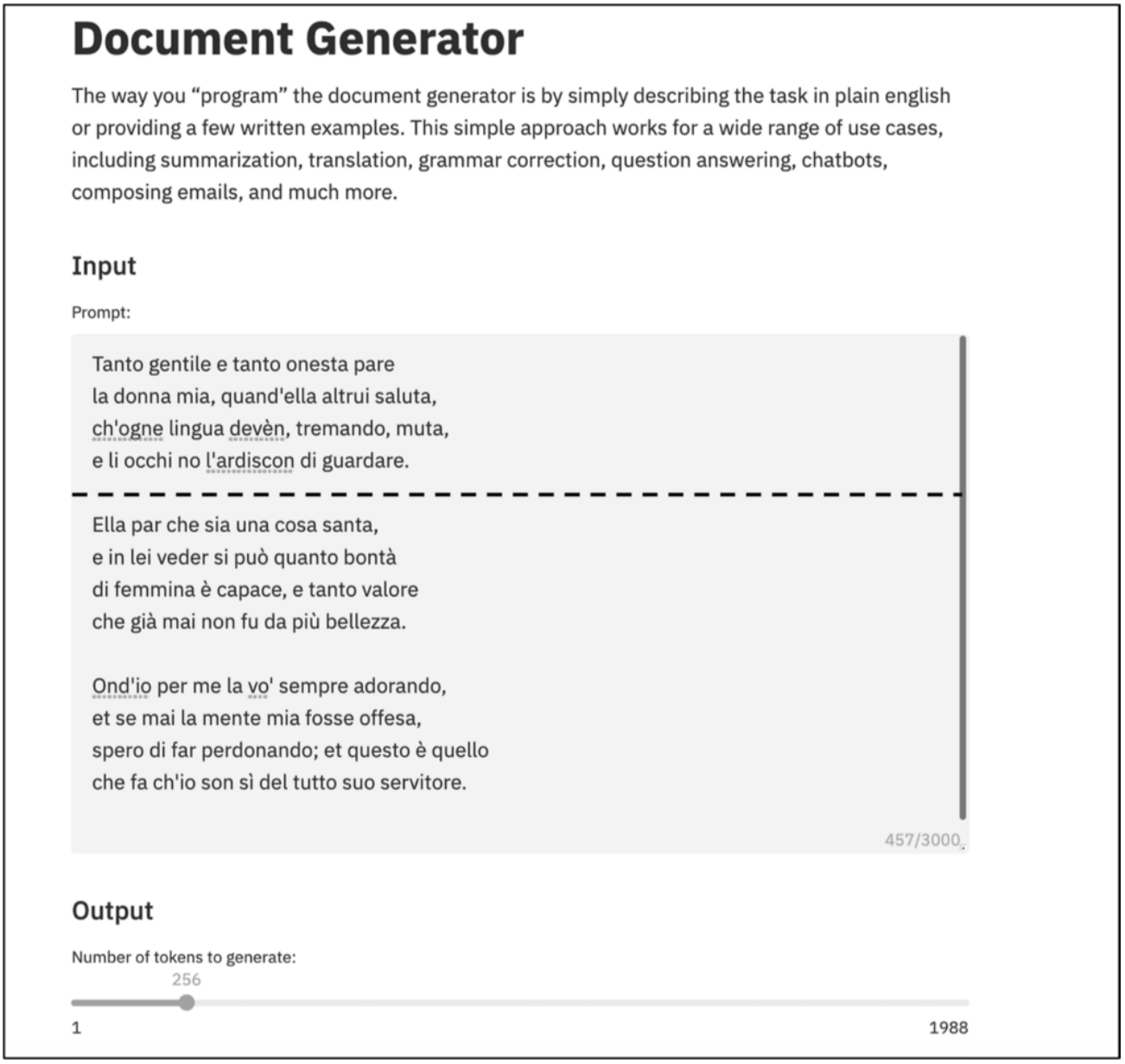

Fig. 4 GPT-3 e Dante, la linea tratteggiata divide il testo originale di Dante in alto, dal testo prodotto da GPT-3 in basso [Sorgente Rif. 5]

Come si vede, questi modelli sono potenzialmente di facile utilizzo per diffondere dati falsi, e questo rischio ha portato ad una profonda discussione [Rif. 6] in corso sulla correttezza etica di rendere disponibili pubblicamente gli ultimi modelli di generazione di testi quali OpenAI GPT-3, Eleuther AI, Meta OPT-175B [Rif. 7].

Al momento quindi, a meno di avere competenze, tempo e soldi disponibili, è improbabile (o almeno limitato a molto pochi) che si riesca a creare un vero “Deep Fake” di quasi impossibile individuazione. L’alternativa è quella di utilizzare modelli commerciali già pronti (“pre-trained”), ovvero già istruiti e disponibili immediatamente, a costi

ridotti o in alcuni casi nulli o quasi nulli. Utilizzare un modello commerciale per creare un testo o modificare audio, immagini o video, può dare dei buoni risultati. D’altra parte, visto che il modello non è stato disegnato ed istruito specificatamente per lo scopo per cui lo si vuole utilizzare, il risultato spesso non è di un livello qualitativo molto elevato. Inoltre molti di questi modelli presentano delle piccole stranezze nei loro prodotti, ad esempio dei rumori particolari di fondo, pixel di colore particolare nelle immagini o movimenti identici e poco spontanei nei video.

Individuare “Deep Fake”

Avendo a disposizione un opportuno modello ML è possibile generare enormi quantità di “Deep Fake” in tempi rapidissimi. Associando a questo la possibilità di condivisione immediata, capillare e globale dei “Deep Fake” tramite piattaforme digitali iperconnesse come i Social Network, ne segue che sono necessari strumenti automatici e molto efficienti per individuare e bloccare queste informazioni malevole.

Ma, come abbiamo già indicato, l’attività in cui i modelli ML sono più efficaci è proprio l’identificazione o classificazione (modelli *Classifier*) dei dati in input. Un possibile scenario è quindi quello in cui i “Deep Fake” sono generati da un attaccante con un modello *Generator* ed identificati da un difensore con un modello *Classifier*. Per entrambe le parti, il problema principale sono i tempi, le competenze, le risorse e i costi di istruzione del modello. In particolare un modello *Classifier* che deve identificare i “Deep Fake” può essere istruito solo con una sufficiente quantità di dati prodotti dal corrispondente *Generator*, il che rende difficile identificare in questo modo in tempi brevi i prodotti di nuovi *Generator*.[3] Inoltre nel caso di uso di *Classifier* pubblici da parte dei difensori, un attaccante può istruire un *Generator* modificando il *Discriminator* in modo che i dati prodotti non siano identificati come falsi dai *Classifier* utilizzati dai difensori. Si ripropone quindi l’usuale rincorsa tra attaccante e difensore in cui ognuno si aggiorna per superare le misure adottate dall’altro.

In pratica oggi lo scenario precedente si semplifica un poco tenendo conto che la grande maggioranza di “Deep Fake” sono generati con modelli *Generator* commerciali i cui prodotti sono più facilmente individuabili (alle volte anche senza la necessità di ricorrere ad un modello ML se non fosse per la quantità di dati e i tempi richiesti). In altre parole, “Deep Fake” generati con modelli *Generator* commerciali sono ad oggi identificabili anche se questo richiede un notevole impegno operativo e continuo aggiornamento dei *Classifier*. Invece veri “Deep Fake” generati con modelli *Generator* tipicamente ad-hoc sono oggi molto complessi, difficili e costosi da creare oltre a richiedere tempi lunghi per l’istruzione del modello, ma sono quasi impossibili da individuare automaticamente, comportandosi quasi come degli “Zero-Day Exploit”.

Una proposta che renderebbe ancora più semplice l’individuazione dei “Deep Fake” prodotti da *Generator* commerciali è quella di sviluppare, definire, standardizzare e poi richiedere che tutti i *Generator* commerciali includano la produzione di qualche forma di Watermark non rimovibile dai dati prodotti e facilmente individuabile dai *Classifier*. D’altra parte, nell’usuale rincorsa tra attaccante e difensore, questo porterebbe immediatamente alla creazione di un mercato alternativo o illegale di *Generator* che non producono i Watermark.

Infine, come già indicato, oltre alla difficoltà di creare *Classifier* in grado di identificare “Deep Fake” con alta probabilità e quindi con pochi falsi negativi, gli stessi *Classifier* devono anche classificare con bassa probabilità come “Deep Fake” i dati reali, ovvero con pochi falsi positivi. Il problema della gestione dei falsi negativi e falsi positivi negli strumenti di rilevazione è ben noto e spesso di difficile soluzione ottimale, ovvero spesso non è possibile minimizzare entrambi, e al momento questo problema sembra ancora più complesso da gestire nei modelli ML.

Concludiamo sottolineando che la ricerca in questo campo è molto attiva e gli sviluppi sono quasi quotidiani visto l’interesse generale al problema dei “Deep Fake” che ha anche spinto grandi aziende tra cui Facebook, Amazon e Microsoft, a sponsorizzare e partecipare al “Deepfake Detection Challenge” [Rif. 8]. E’ quindi possibile che in poco tempo gli scenari descritti in questo articolo si modifichino sostanzialmente.

Riferimenti Bibliografici

Rif. 1: AWS Deep Composer https://aws.amazon.com/it/deepcomposer/

Rif. 2: AWS Deep Composer Github: https://github.com/aws-samples/aws-deepcomposer-samples/

Rif. 3: “The innocuous photos banned by Facebook: Social media giant apologises after it blocks art gallery’s images of COWS, the England cricket team and a high-rise office block because they are judged ‘too sexy’”, MailOnline News, febbraio 2021, https://www.dailymail.co.uk/news/article-9249087/Facebook-apologises-blocks-art-gallerys-images-COWS-sexy.html

Rif. 4: “Deepfake video of Zelensky telling Ukrainians to surrender removed from social platforms“, New York Post 2022/03/17, https://nypost.com/2022/03/17/deepfake-video-shows-volodymyr-zelensky-telling-ukrainians-to-surrender/

Rif. 5: “GPT-3: Its Nature, Scope, Limits, and Consequences”, L.Floridi e M.Chiriatti, Minds and Machines vol. 30, pag. 681–694 (2020) Springer, https://link.springer.com/article/10.1007/s11023-020-09548-1

Rif. 6: si veda ad esempio “The Radicalization Risks of GPT-3 and Neural Language Models”, K.McGuffie and A.Newhouse. CTEC, https://www.middlebury.edu/institute/academics/centers-initiatives/ctec/ctec-publications/radicalization-risks-gpt-3-and-neural-language

Rif. 7: https://openai.com/ , https://www.eleuther.ai/ , https://ai.facebook.com/blog/democratizing-access-to-large-scale-language-models-with-opt-175b/

Rif. 8: Deepfake Detection Challenge – DFDC (Facebook, Amazon, Microsoft, PartnershipOnAI) https://ai.facebook.com/datasets/dfdc/ , https://partnershiponai.org/a-report-on-the-deepfake-detection-challenge/ , https://www.iqt.org/deepfake-detection-challenge/

Note

[1] Basta pensare agli studi ed investigazioni (alcune delle quali ancora in corso) sul ruolo ed effetto delle “Fake News” nelle campagne elettorali per le elezioni presidenziali Statunitensi del 2016 e 2020.

[2] Una “Fake News” può anche essere generata semplicemente estrapolando o utilizzando delle informazioni vere in un contesto non consistente.

[3] Va comunque ricordato che un *Classifier* è di solito in grado di individuare con buona probabilità prodotti di *Generator* sufficientemente simili a quelli sui quali è stato istruito.

Articolo a cura di **Andrea Pasquinucci**

Profilo Autore

Andrea Pasquinucci

PhD CISA CISSP
Consulente freelance in sicurezza informatica: si occupa prevalentemente di consulenza al top management in Cyber Security e di progetti, governance, risk management, compliance, audit e formazione in sicurezza IT.

Altri Articoli

- Intelligenza Artificiale / Machine Learning: tra Complessità e Sicurezza
- Sicurezza, Hardware e Confidential Computing – Parte 2
- Sicurezza, Hardware e Confidential Computing - Parte 1
- QUIC: un nuovo protocollo Internet e la sicurezza IT

#deep fake #Fake News #Machine Learning

← PRECEDENTE
MDR nella sanità. Cyber Security in ambienti critici

Articoli simili