Corso di Intelligenza Artificiale
Magistrale Informatica

Relazione Progetto d'Esame

ChatGPT e creatività: L'Arte della Poesia al sicuro dalle AI

Anno Accademico 2022/2023

Componenti del gruppo Deep Dreamers:
Gruppi Luca - Mat: 0001099394
(luca.gruppi@studio.unibo.it)
Raciti Gabriele - Mat: 0001102147
(gabriele.raciti2@studio.unibo.it)
Tamai Leonardo - Mat: 0001098711
(leonardo.tamai@studio.unibo.it)

0. INDICE

1. INTRODUZIONE

1.1. Problema

Nel corso degli ultimi due decenni, l'intelligenza artificiale ha fatto progressi straordinari che hanno cambiato profondamente il modo in cui interagiamo con la tecnologia e il mondo che ci circonda.
Infatti, con l'avvento di ChatGPT, un gran numero di persone è venuta a conoscenza delle grande possibilità e opportunità che l'intelligenza artificiale può offrire.
Essa adesso viene infatti utilizzata in una moltitudine di casi, come ad esempio la generazione di testo, di immagini, canzoni e molto altro.
Infatti, una delle evoluzioni più sorprendenti e rivoluzionarie dell'IA riguarda la capacità di generare testi in modo autonomo e coerente, con risultati che stanno sollevando importanti questioni sul futuro della comunicazione, della creatività e persino dell'identità umana.
L'intelligenza artificiale è infatti in grado di produrre testi che, in molti casi, sono potenzialmente indistinguibili da quelli scritti da esseri umani.
Questa capacità ha suscitato reazioni contrastanti: da un lato, è un progresso importante nell'automazione delle attività legate alla scrittura, dall'altro, solleva molti interrogativi etici sull'unicità e straordinarietà dell'essere umano.
Infatti, una delle principali preoccupazioni è che l'intelligenza artificiale stia gradualmente cambiando il modo in cui percepiamo e valutiamo la scrittura umana. La possibilità di generare testi di alta qualità in modo automatico ha infatti portato a dubitare dell'idea comune che i testi scritti siano l'espressione di pensieri, emozioni e creatività umane uniche e irripetibili.
Questo progetto è volto a capire se un'intelligenza artificiale, in particolare **ChatGPT-3.5** (d'ora in poi chiamato solo ChatGPT per semplicità), è realmente in grado di emulare lo stile di scrittura dei più grandi poeti italiani. Il nostro lavoro ci ha quindi portato a concentrarci sul riconoscimento dell'autenticità di una poesia per capire se realmente una IA è in grado di replicare lo stile di un poeta, rendendolo di fatto "nulla di eccezionale".

L’argomento è sicuramente molto ampio e la tecnologia si sta sviluppando molto rapidamente. È già stato annunciato l’arrivo di ChatGPT-4, che promette prestazioni superiori grazie a una rete più grande e un addestramento con una quantità ancora maggiore di dati. Questo potrebbe rendere il lavoro attuale meno rilevante, ma esso rimarrà come una testimonianza e una documentazione dello stato dell’arte fino a questo momento.
Ricapitolando quindi, la domanda alla quale vogliamo trovare risposta è: l’essere umano è davvero unico e inimitabile, in grado di produrre contenuti testuali ineguagliabili e straordinari? È possibile capire se una poesia di un determinato poeta è realmente stata scritta da lui o da una intelligenza artificiale?

1.1.1. Perchè è un problema

Il problema esiste e non riguarda un unico campo.
In primo luogo infatti, come anticipato sopra, questo vorrebbe dire non essere in grado di riconoscere se una poesia, o più in generale un testo, sia stato scritto da un’intelligenza artificiale o da un essere umano. Questo porterebbe sicuramente a delle conseguenze sul settore della scrittura professionale, con conseguenze per gli scrittori e i giornalisti.
Un altro motivo, puramente egoistico, sarebbe quello di natura umana. L’uomo ha sempre pensato di essere speciale e unico rispetto alle fredde macchine che, malgrado lo pseudonimo di “intelligenza artificiale”, compiono le loro azioni in base a dei pesi che sono stati modificati allenamento dopo allenamento.
Non riconoscere se una poesia è stata scritta da noi o da una IA vorrebbe dire ammettere che siamo prevedibili e che anche noi siamo in un certo senso delle macchine che compiono le loro decisioni tramite dei “pesi” che si sono modificati con le esperienze compiute, e quindi di fatto prevedibili.

1.1.2. Per chi è un problema

A cosa potrebbe mai servire più un poeta a quel punto?
Le poesie potrebbero essere generate a decine senza alcuno sforzo.
La figura del poeta risulterebbe essere obsoleta, proprio come quei lavori meccanici nelle catene di montaggio che sono già stati sostituiti dalle macchine.
Inoltre, se allarghiamo il discorso ai testi in generale, non poter sapere se un testo è stato scritto da una persona o da una macchina porterebbe a conseguenze sul mercato del lavoro, influenzando il settore della scrittura professionale che inevitabilmente cambierà in maniera irreversibile.
Allargando ancora di più il discorso, il problema riguarderebbe l'intera specie umana, che dovrà rassegnarsi al fatto che non ha nulla di speciale e unico rispetto alle macchine, essendo di fatto prevedibile e replicabile.

1.1.3. Che beneficio porterebbe una soluzione ipotetica al problema

Saper riconoscere se una poesia è stata scritta da un umano porterebbe ad una maggiore valorizzazione del lavoro dei poeti. Essi verrebbero infatti riconosciuti per la loro abilità unica e il loro contributo speciale.
Inoltre ciò riconoscerebbe la capacità umana di creare qualcosa che non può essere replicato artificialmente.

1.2 Soluzione proposta

La prima idea che ci è venuta in mente è stata quella di creare una neural network che, una volta allenata tramite Machine Learning Supervisionato, riconoscesse se la poesia è stata scritta da da un essere umano oppure da ChatGPT-3.5. Per fare ciò, abbiamo raccolto la maggior parte delle poesie scritte dagli autori italiani più famosi (Leopardi, Montale, Pascoli, ecc.) e abbiamo fatto in modo che ChatGPT generasse un numero simile di poesie a quelle a nostra disposizione, emulando lo stile di ogni singolo autore. Una volta fatto ciò abbiamo

estratto le caratteristiche testuali delle poesie raccolte e le abbiamo utilizzate per addestrare la neural network.
In più, dopo esserci resi conto delle risorse limitate in nostro possesso, abbiamo compiuto lo stesso studio utilizzando altri approcci di Machine Learning, quali Decision Tree, Random Forest e SVM, risultando essere più efficaci con dataset di dimensioni relativamente ridotte.

1.2.1. Come si intende affrontare la realizzazione di una soluzione

Il primo ostacolo da affrontare è stato la realizzazione del dataset. Infatti, non essendo riusciti a trovare un dataset già disponibile di poesie di poeti italiani, abbiamo dovuto costruirlo noi. In più, abbiamo dovuto fare la stessa cosa con le poesie che ChatGPT ha generato emulando i vari stili degli autori da noi scelti. Con alcuni script e del lavoro manuale coordinato siamo comunque riusciti ad ottenere una discreta quantità di poesie (più di 3000).
Successivamente, il secondo ostacolo è stato quello di capire come estrarre delle caratteristiche significative da queste poesie, da dare successivamente in pasto alla nostra neural network. Siamo riusciti a compiere questo task analizzando degli studi sul Natural Language Processing per capire cosa estrarre, implementando delle funzioni di analisi sintattica e utilizzando delle librerie python utili ad analizzare il POS (Part-of-Speech, categorizzazione delle parole all'interno di una frase in base al loro ruolo grammaticale e alla loro funzione nella struttura della frase) di ogni frase.
Successivamente abbiamo creato la nostra neural network, alla quale abbiamo dato in pasto le caratteristiche estratte precedentemente per addestrarla a riconoscere una poesia generata da un essere umano e una generata con ChatGPT. Naturalmente, abbiamo dedicato particolare attenzione all'ottimizzazione della nostra neural network, esplorando diverse configurazioni e strategie per garantire la massima efficienza e accuratezza.
Successivamente, per completezza e a causa delle dimensioni relativamente ridotte del nostro dataset, abbiamo utilizzato il dataframe

da noi creato per addestrare altri modelli di Machine Learning quali Decision Tree, Random Forest e SVM.
Abbiamo infine proseguito con una valutazione dei risultati ottenuti e esaminato le implicazioni delle nostre scoperte per il nostro progetto.

1.2.2. Quali sono le sfide (informatiche)

Le sfide principali sono state:

- Creazione del dataset
- Estrazione caratteristiche utili per addestramento
- Pre-processing dei dati
- Decisione approccio migliore da utilizzare
- Realizzazione neural network con relativa configurazione migliore.
- Realizzazione Decision Tree, Random Forest, SVM.
- Tuning dei parametri
- Analisi dei risultati

1.3. Revisione della letteratura

Gli articoli più rilevanti che abbiamo trovato sono due.
Il primo articolo è *Distinguishing Human Generated Text From ChatGPT Generated Text Using Machine Learning* di Niful Islam , Debopom Sutradhar , Humaira Noor, in collaborazione con altri autori. L'articolo è molto recente, è stato pubblicato il 26 Maggio 2023 e ha come scopo quello di creare una neural network in grado di distinguere testo generato da un essere umano da quello generato da ChatGPT-3.5.
La rete è stata allenata su un database di 10000 testi, 5204 scritte da umani e prese da news e social media. Il paper reclama dei risultati che si avvicinano al 77% di accuratezza.
I test sono stati fatti con svariati modelli: decision tree, logistic regression e altri. Il modello che da risultati migliori è Extremely Randomized Trees che per appunto presenta un valore di accuratezza di 77%. Precisione 74%, F1-score 76%. Il lavoro di pre-processing che è stato fatto partendo dal dataset è stato per prima cosa la tokenization delle singole parole e successivamente vettorializzare le parole in numeri (purtroppo non è specificata la dimensione vettoriale delle parole).

Il secondo articolo preso in considerazione è *Distinguishing Human-Written and ChatGPT-Generated Text Using Machine Learning.* Di Hosam Alamleh, Ali Abdullah S. AlQahtani, AbdElRahman ElSaid. In questo caso il lavoro di preprocessing è più simile a quello svolto da noi, hanno cercato di estrapolare delle features da usare come input della rete al posto di dare un valore alle singole parole della rete. Differentemente dal nostro studio, la versione di GPT utilizzata è la 2. Utilizzando il random forest arrivano a valori di accuratezza del 93%.

1.4. Suddivisione lavoro di gruppo e strumenti utilizzati per la condivisione

L'intero processo di condivisione del lavoro è stato gestito attraverso la piattaforma GitHub. La suddivisione del lavoro di gruppo è avvenuta tramite incontri in presenza o incontri online sulla piattaforma Discord, nei quali abbiamo assegnato e coordinato task individuali e collettivi all'interno del gruppo di lavoro.

1.5. Risultati ottenuti (in breve)

I risultati ottenuti hanno dimostrato che è possibile distinguere una poesia generata da un'intelligenza artificiale rispetto a una poesia scritta da un essere umano. In particolare, la neural network ha riportato risultati insoddisfacenti vista la ridotta dimensione del dataset, mentre le altre tecniche utilizzate, quali Decision Tree e Random Forest hanno performato molto bene. SVM ha fornito risultati leggermente meno significativi rispetto agli ultimi due modelli menzionati, ma rimane comunque senza dubbio favorevole alla nostra tesi.

2. METODO PROPOSTO

2.1. Ricerca della soluzione

Nella scelta della soluzione proposta sono state considerate diverse alternative. In particolare, la prima scelta è ricaduta su una neural network addestrata tramite Machine Learning Supervisionato che fosse in grado di discriminare tra poesie umane e poesie di ChatGPT. Successivamente, dopo esserci resi conto delle dimensioni limitate del nostro dataset, abbiamo optato per implementare altri meccanismi di Machine Learning, ovvero Decision Tree, Random Forest e SVM, essendo più funzionali con dataset di grandezza limitata. Come previsto questi meccanismi, in particolare Decision Tree e Random Forest, hanno ottenuto degli ottimi risultati. Anche SVM ha ottenuto degli ottimi risultati, ma leggermente inferiori rispetto a quelli descritti precedentemente.

2.2. Giustificazione della scelta

Abbiamo deciso di creare altri modelli oltre la neural network dopo aver capito il limite che essa aveva con il nostro dataset di dimensioni limitate. Dopo aver infatti analizzato i risultati ottenuti, ci siamo resi conto che la neural network non stava effettivamente “discriminando” le poesie, in quanto i dati a sua disposizione non erano abbastanza numerosi per farle comprendere la differenza tra una poesia umana e una generata artificialmente. Abbiamo quindi deciso di affidarci a modelli più “semplici”, che funzionano meglio in presenza di dataset di dimensione limitata. Abbiamo quindi scelto di replicare l’esperimento addestrando dei Decision Tree, Random Forest e SVM, ottenendo risultati ben migliori rispetto alla neural network, come previsto.

2.3. Creazione dell’esperimento

Nel paragrafo seguente, esamineremo il processo di creazione del nostro esperimento, a partire dalla costruzione del dataset.

La costruzione del dataset è stata attuata in modo parzialmente “manuale”, ovvero attraverso la ricerca di un numero significativo di poesie di alcuni poeti umani rilevanti, ovvero: Alda Merini, Giacomo Leopardi, Umberto Saba, Eugenio Montale, Giuseppe Ungaretti, Salvatore Quasimodo, Pier Paolo Pasolini, Francesco Petrarca e Giovanni Pascoli. La controparte dei dati, ovvero le poesie generate da ChatGPT, è stata creata utilizzando ChatGPT inviando come input la frase “crea 2/3/5 poesie emulando lo stile di questo autore.”. Il numero di poesie da generare (ovvero 2, 3 e 5) influisce sulla lunghezza delle stesse e di conseguenza lo abbiamo variato per ottenere risultati più eterogenei. Questo processo è stato messo appunto con uno script Python da noi realizzato che richiede la generazione delle poesie a ChatGPT a intervalli di tempo regolari. Successivamente, le suddette chat sono state scaricate attraverso il plugin per browser “Save ChatGPT”.
Per comodità, abbiamo realizzato due file di testo per ogni autore considerato: uno contenente le poesie originali e l’altro contenente le poesie generate da ChatGPT (suddivise tramite un separatore “*”, per comodità). Abbiamo successivamente utilizzato uno script in python per mettere assieme tutte le poesie raccolte e generare le label corrispondenti a ogni singola poesia (per poter distinguere le poesie generate da ChatGPT da quelle umane). In particolare, le poesie vengono raggruppate in un unico file chiamato “poesie.txt”, mentre le label corrispondenti alle singole poesie vengono raggruppate in un file “autori.txt”. Entrambi questi file sono generati automaticamente dal nostro script.
Una volta generato il nostro dataset, composto da 1496 poesie create da ChatGPT e 1457 poesie originali, siamo passati al pre-processing dei dati. Abbiamo identificato, selezionato ed estratto le caratteristiche testuali più rilevanti. Questo lavoro è stato fatto tramite metodi da noi implementati che analizzano le caratteristiche testuali e con l’aiuto di librerie python di Natural Language Processing.
Infine, una volta ottenuti i dati numerici risultanti e costruito il dataframe con uno script Python, abbiamo sviluppato e addestrato i nostri modelli di intelligenza artificiale sfruttando gli approcci precedentemente menzionati, ovvero Reti Neurali, Decision Tree, Random Forest e SVM.

2.4. Scelte del Preprocessing

Insieme alla creazione del dataset, anche il preprocessing ha rappresentato una grossa sfida data la rinomata difficoltà generale del trattamento del testo in problemi di Intelligenza Artificiale come quello trattato.
Abbiamo rilevato un numero significativo di caratteristiche fondamentali per descrivere una qualsiasi poesia. In particolare, le prime 8 features utilizzate sono:

0. **word_count**: il numero di parole di una singola poesia;
1. **sentence_count**: il numero di frasi di una singola poesia;
2. **comma_count**: il numero di virgole presenti in una singola poesia;
3. **exclamation_count**: il numero di punti esclamativi presenti in una singola poesia;
4. **unique_word_count**: il numero di parole uniche (non ripetute) in una singola poesie
5. **mean_words_phrases**: il numero medio di parole per frase di una singola poesia;
6. **polarity**: valore compreso fra -1 e 1, indica rispettivamente la positività o negatività del sentimento di una singola poesia;
7. **subjectivity**: valore compreso fra 0 e 1, indica rispettivamente la tendenza a essere un'opinione personale oppure un giuzidio oggettivo di una singola poesia;

Le successive 19 rappresentano i POS tag. Ognuno di questi valori è stato diviso per il numero totale di parole nella poesia, in modo da ottenere una sorta di misura della frequenza di ognuno. In particolare:

8. **ADJ** (pos0): il numero di aggettivi presenti in una singola poesia;
9. **ADP** (pos1): il numero di preposizioni/congiunzioni presenti in una singola poesia;
10. **ADV** (pos2): il numero di avverbi presenti in una singola poesia;
11. **AUX** (pos3): il numero di ausiliari presenti in una singola poesia;
12. **CONJ** (pos4): il numero di congiunzioni presenti in una singola poesia;
13. **CCONJ** (pos5): il numero di congiunzioni di coordinazione presenti in una singola poesia;
14. **DET** (pos6): il numero di articoli presenti in una singola poesia;

15. **INTJ** (pos7): il numero di interiezioni presenti in una singola poesia;
16. **NOUN** (pos8): il numero di nomi presenti in una singola poesia;
17. **NUM** (pos9): il numero di numeri presenti in una singola poesia;
18. **PART** (pos10): il numero di particelle presenti in una singola poesia;
19. **PRON** (pos11): il numero di pronomi presenti in una singola poesia;
20. **PROPN** (pos12): il numero di nomi propri presenti in una singola poesia;
21. **PUNCT** (pos13): il numero di simboli di punteggiatura presenti in una singola poesia;
22. **SCONJ** (pos14): il numero di congiunzioni di subordinazione presenti in una singola poesia;
23. **SYM** (pos15): il numero di simboli presenti in una singola poesia;
24. **VERB** (pos16): il numero di verbi presenti in una singola poesia;
25. **X** (pos17): il numero di parole prive di significato presenti in una singola poesia;
26. **SPACE** (pos18): il numero di spazi presenti in una singola poesia.

In particolare, il grafico sottostante ci fornisce informazioni riguardanti l'importanza delle feature elencate precedentemente nel processo di discriminazione, emerse dai modelli di Random Forest e similmente da Decision Tree.
Possiamo notare che la feature meno significativa è la numero 27 (SPACE), che risulta essere totalmente irrilevante.
La più significativa è invece la feature numero 4 (unique_word_count), ovvero la feature che tiene conto delle parole diverse utilizzate. Grazie a questo grafico riusciamo quindi a comprendere i primi limiti nella generazione di testo da parte di ChatGPT.

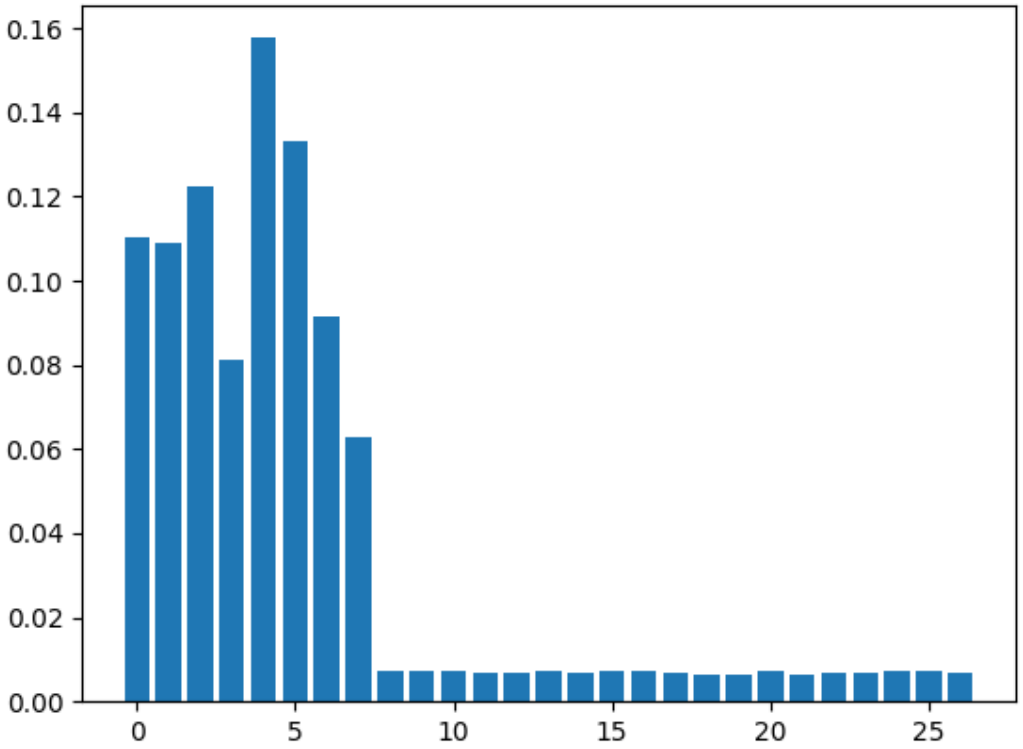

Fig. sull'asse X il numero della feature, sull'asse Y il contributo delle feature al modello.

2.5. Descrizione del metodo per la misurazione delle performance

La misurazione delle performance ottenute è stata misurata principalmente con il metodo dell'Accuracy. In particolare, abbiamo deciso di utilizzare l'Accuracy in quanto essa è la misura comune utilizzata principalmente per la classificazione binaria o multiclasse, come nel nostro caso. Essa rappresenta la percentuale di previsioni corrette fatte dal modello rispetto al numero totale di previsioni, ed è utile quando tutte le classi hanno la stessa importanza e non c'è un forte sbilanciamento tra le classi. Per completezza, abbiamo comunque deciso di calcolare anche la Precision, il Recall e l'F1-Score, in modo da avere un quadro il più completo possibile dei risultati. In particolare, le varie metriche sono definite come segue:

- **Accuracy**: $\frac{veri\ positivi + veri\ negativi}{veri\ positivi + falsi\ positivi + veri\ negativi + falsi\ negativi}$
- **Precision**: $\frac{veri\ positivi}{veri\ positivi + falsi\ positivi}$

- **Recall**: $\frac{veri\ positivi}{veri\ positivi + falsi\ negativi}$
- **Punteggio F1**: $2 * \frac{precision * recall}{precision + recall}$

Successivamente è stata riportata anche la **matrice di confusione**, dove ogni colonna della matrice rappresenta i valori predetti, mentre ogni riga rappresenta i valori reali. L'elemento sulla riga *i* e sulla colonna *j* è il numero di casi in cui il modello ha classificato la classe "vera" *i* come classe *j*.
Infine, i risultati della rete neurale forniscono anche la **loss**, ovvero la somma degli errori di classificazione accumulati, che ci permette di quantificare quanto le previsioni della rete di discostano dai valori reali.

3. RISULTATI SPERIMENTALI

3.1. Istruzioni per la dimostrazione

Avendo già creato in precedenza i modelli, le istruzioni per la dimostrazione dei risultati ottenuti riguardano l'esecuzione di un singolo script, ovvero *run.py*. Per eseguire questo script, dopo aver installato tutte le librerie necessarie indicate nel paragrafo successivo, basterà digitare da linea di comando *python run.py*. L'esecuzione di questo script mostrerà a schermo i risultati ottenuti dai nostri modelli. In particolare, per quanto riguarda la neural network verrà mostrato il numero di batch, il tempo trascorso durante un'epoca dell'addestramento, il valore della funzione di loss, l'accuracy, il tempo medio necessario per completare un'epoca di addestramento e il tempo medio impiegato per uno step durante l'addestramento. Invece, per quanto riguarda i modelli di Decision Tree, Random Forest e SVM, verranno mostrate per ogni modello delle metriche quali accuracy, precision, recall, f1-score e support.
In più, per questi ultimi modelli specificati precedentemente, verranno mostrati i parametri utilizzati durante il tuning dei modelli e la configurazione migliore ottenuta grazie alla combinazione di essi.

3.2. Elenco delle tecnologie usate per gli esperimenti per la riproducibilità

Mostriamo di seguito le tecnologie utilizzate per la costruzione del nostro esperimento:

- **Python** (versione 3.10.11): per la realizzazione dell'intero progetto e per l'auto compilazione delle domande fatte a ChatGPT.
- **ChatGPT** (versione 3.5): per la creazione di poesie simil reali.
- **TensorFlow** (versione 2.11.0): per la creazione e la gestione della rete neurale
- **Scikit-learn** (versione 1.3.1): per la realizzazione di modelli quali Decision Tree, Random Forest e SVM.

- **spaCy** (versione 3.5.2): per svolgere una vasta gamma di attività di analisi e comprensione del linguaggio naturale.
- **Pattern** (versione 3.6): per l'analisi del testo
- **Numpy** (versione 1.24.2): per manipolare e lavorare con array multidimensionali e matrici
- **Matplotlib** (versione 3.7.1): per la visualizzazione di dati e la creazione di grafici.
- **Pandas** (versione 1.5.3): per leggere il file csv e per creare il dataframe.
- **Joblib** (versione 1.2.0): per la serializzazione e il salvataggio dei modelli.
- **Save ChatGPT** (versione 3.1): per il salvataggio delle chat con ChatGPT.
- **Git**: per poter lavorare in team apportando modifiche allo stesso progetto.

3.3. Risultati configurazione migliore

Di seguito sono riportati i risultati delle configurazioni migliori dei 4 modelli: Neural Network, Decision Tree, Random Forest e SVM.

Configurazione Neural Network:

È composta da 5 livelli: il primo riceve tutti i dati di ingresso dopo il preprocessing delle poesie. Successivamente, 3 livelli interni composti rispettivamente da 1000, 500 e 1000 neuroni. Infine, l'ultimo livello composto da due uscite: la prima rappresenta il poeta umano e la seconda indica se la poesia è stata generata da ChatGPT.

Modello sequenziale		
Layer (tipo)	**Output shape**	**Numero parametri**
Denso	(1, 28)	841
Denso	(1, 1000)	30000
Denso	(1, 500)	500500
Denso	(1, 1000)	501000
Denso	(1, 2)	2002

Total params: 1,033,286
Trainable params: 1,033,286
Non-trainable params: 0
Loss: 0.4188
Accuracy: 0.7907
Tempistiche: 183ms/epoch - 6ms/step

Configurazione migliore Decision Tree:
Risultati train decision tree:

- criterion: entropy
- max_depth: 16
- min_samples_leaf: 2
- min_samples_split: 6
- splitter: best

Risultati accuracy test decision tree: 0.93058

DT	**Precision**	**Recall**	**F1-score**	**Support**
Poeti	0.90	0.94	0.92	435
ChatGPT	0.95	0.92	0.94	559
Accuracy			0.93	994

Confusion matrix:

	Poeti predetti	**ChatGPT predetti**
Poeti reali	410	25
ChatGPT reali	44	515

Configurazione migliore Random Forest:
Risultati train random forest:

- criterion: entropy
- max_depth: 20
- max_features: sqrt
- min_samples_split: 7
- n_estimators: 300

Risultati accuracy test random forest: 0.91851

	Precision	**Recall**	**F1-score**	**Support**
Poeti	0.91	0.91	0.91	435
ChatGPT	0.93	0.93	0.93	559
Accuracy			0.92	994

Confusion matrix:

	Poeti predetti	**ChatGPT predetti**
Poeti reali	395	40
ChatGPT reali	41	518

Configurazione migliore SVM:
Risultati train SVM:

- C: 7000
- gamma: 1
- kernel: poly

Risultati accuracy test SVM: 0.89034

	Precision	**Recall**	**F1-score**	**Support**
Poeti	0.90	0.84	0.87	435
ChatGPT	0.88	0.93	0.91	559
Accuracy			0.89	994

Confusion matrix:

	Poeti predetti	**ChatGPT predetti**
Poeti reali	365	70
ChatGPT reali	39	520

3.4. Studio di ablazione: comparazione tra diverse configurazioni

Di seguito mostriamo le varie strutture e configurazioni di tutti i modelli da noi realizzati.

Struttura della neural network:

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(28, activation='relu'),
    tf.keras.layers.Dense(1000, activation='relu'),
    tf.keras.layers.Dense(500, activation='relu'),
    tf.keras.layers.Dense(1000, activation='relu'),
    tf.keras.layers.Dense(2) ])
```

Abbiamo deciso di costruire la neural network con 5 livelli per dei motivi quali la non esagerata complessità del problema e la scarsità di dati disponibili. Inoltre, alternando livelli composti da più neuroni con livelli composti da meno neuroni, favoriamo l'estrazione di feature gerarchiche, riduciamo la complessità computazionale e aiutiamo a prevenire l'overfitting.

Parametri e valori utilizzati per la creazione del modello Decision Tree:

- 'criterion': ['gini','entropy','log_loss'], ovvero la funzione per misurare la qualità di uno split;
- 'splitter': ['best', 'random'], ovvero la strategia utilizzata per decidere lo split ad ogni nodo;
- 'max_depth': [10,12,14,16,18], ovvero la massima profondità dell'albero;
- 'min_samples_split':range(2,10), ovvero il minimo numero di samples richiesti per eseguire uno split su un nodo interno;
- 'min_samples_leaf':range(1,5), ovvero il numero minimo di samples richiesti per avere un nodo foglia.

Parametri e valori utilizzati per la creazione del modello Random Forest:

- 'n_estimators': [200,300], ovvero il numero di alberi nella foresta;
- 'max_features': ['sqrt', 'log2'], ovvero il numero di features da considerare al momento della creazione del migliore split;
- 'max_depth' : [16,18,20], ovvero la massima profondità degli alberi decisionali all'interno della Random Forest;
- 'criterion' :['gini', 'entropy'], ovvero la funzione per misurare la qualità di uno split;
- 'min_samples_split':[5,6,7], ovvero il minimo numero di samples richiesti per eseguire uno split su un nodo interno;

Parametri e valori utilizzati per la creazione del modello Support Vector Machine:

- 'C': [7000,6500], parametro di regolazione
- 'gamma': [1], coefficiente per il kernel (l'aggiunta di più valori possibili di questo parametro aumenta sensibilmente il tempo di computazione, di conseguenza, dopo aver effettuato dei test è stato lasciato a 1);
- 'kernel': ['rbf', 'poly', 'sigmoid'], specifica il tipo di kernel da utilizzare all'algoritmo.

3.5. Studi di comparazione: comparazione con alcune piattaforme anti-plagio

Per avere un'idea delle reali capacità del sistema sviluppato, abbiamo provato ad inserire alcune delle poesie create da ChatGPT su alcuni software online di riconoscimento di autenticità del testo, per poi darli in pasto al nostro sistema e comparare i risultati. In particolare, abbiamo generato 20 poesie chiedendo a ChatGPT di emulare lo stile di alcuni degli autori da noi scelti per la costruzione del dataset. I siti utilizzati come rilevatori di plagio online, che vengono utilizzato per rilevare il plagio di ChatGPT, sono due: ZeroGPT (https://www.zerogpt.com/) e Corrector App (https://corrector.app/it/rilevatore-contenuti-ia/). Prima di inserire le nostre poesie, abbiamo provato a testare i due siti con del normale testo generato da ChatGPT, ottenuto chiedendo di generare del testo per dei temi di italiano di vari argomenti. Per quanto riguarda questo tipo di input entrambi i siti si sono rivelati efficienti, ottenendo ottimi risultati e riconoscendo sempre la presenza dell'AI nel testo inserito.
Successivamente abbiamo provato a testare le nostre 20 poesie generate. Per un corretto riconoscimento e un esperimento il più preciso possibile, abbiamo fatto in modo che la lunghezza delle poesie variasse, inserendo poesie che vanno dai 130 ai 1400 caratteri. Inaspettatamente, i due rilevatori non sono stati in grado di identificare in NESSUNA delle poesie prese in input la presenza dell'intelligenza artificiale. Abbiamo ipotizzato che questo sia dovuto al fatto che i modelli che stanno alla base di questi rilevatori online non siano stati addestrati con delle

poesie, ma soltanto con del comune testo generato. Successivamente abbiamo testato le stesse poesie con il nostro modello, ottenendo risultati ottimi. Il nostro modello è stato in grado di riconoscere la generazione dei testi da parte di ChatGPT in 16 poesie su 20.
In sintesi quindi, l'esperimento condotto ha rivelato che gli strumenti online di riconoscimento di autenticità del testo mostrano una difficoltà non indifferente nel riconoscere la generazione di poesie da parte di ChatGPT. Il nostro modello invece è risultato essere più sensibile nel riconoscere la generazione di testi poetici da parte di ChatGPT. Questi risultati mettono in evidenza la necessità di continuare a sviluppare modelli di rilevamento del plagio specifici per ogni contesto, in modo da poterci "difendere" il più possibile dall'uso improprio di contenuti generati dall'intelligenza artificiale.

3.6 Studi di comparazione: comparazione con le soluzioni presenti in letteratura

Ricordiamo che le soluzioni presenti in letteratura che abbiamo trovato sono entrambe riguardanti dei modelli addestrati su testo generato e non su poesie come nel nostro caso. Mostriamo successivamente i risultati ottenuti dalle soluzioni presenti in letteratura.

Risultati accuracy *Distinguishing Human-Written and ChatGPT Generated Text Using Machine Learning*
Nel seguente articolo è stato svolto un interessante lavoro nella creazione del dataset: metà è composto da testi (Essay Prompts) scritti da ChatGPT-3.5 e da un gruppo di studenti sulla base di alcune domande con un limite di lunghezza, mentre l'altra metà è composto da quesiti di programmazione C e Python (Programming Prompts) riguardanti sia programmi semplici che codici più complessi con classi e funzioni multiple, sottoposti anch'essi sia a ChatGPT sia ai medesimi studenti. In particolare i risultati ottenuti sono i seguenti:

Modello	Essay Prompts	Programming Prompts	Combination
Neural Network	90.5%	86.5%	91.75%
Decision Tree	87%	87.5%	88%
Random Forest	93%	93.5%	92.5%
SVM	91.5%	91%	91%

Risultati *Distinguishing Human Generated Text FromChatGPT Generated Text Using Machine Learning*

La particolarità di questo studio, svolto su ChatGPT-2, è che tra gli approcci proposti è presente l'utilizzo di un Percettrone Multistrato: i neuroni artificiali sono il concetto principale del Perceptron multistrato (MLP). Questi neuroni sono un insieme di unità o nodi interconnessi che assomigliano vagamente ai neuroni di un cervello biologico. Come le sinapsi in un cervello biologico, ogni connessione ha la capacità di inviare un segnale ai neuroni vicini. L'output di ciascun neurone è calcolato da una funzione di attivazione.
Inoltre, in questo articolo è stato svolto un lavoro più simile al nostro per quanto riguarda il pre-processing dei dati. Nonostante infatti si parli di testo generato e non poesie come nel nostro caso, il lavoro svolto nel pre-processing è molto simile al nostro, cercando di estrapolare delle features da usare successivamente come input della rete. In particolare i risultati ottenuti sono i seguenti:

Modello	Accuracy	Precision	Recall	F1 score
Multi-layer Perceptron	0.72	0.73	0.72	0.72
Decision Tree	0.63	0.75	0.79	0.67
Random Forest	0.76	0.73	0.81	0.76
SVM	0.75	0.75	0.71	0.73

4. DISCUSSIONE E CONCLUSIONI

4.1. Discussione dei risultati ottenuti

I risultati che abbiamo ottenuto tramite lo svolgimento di questo progetto sono stati ottimi. Nonostante la difficoltà iniziale dovuta principalmente alla mancanza di un dataset già pronto da utilizzare, siamo riusciti a raggruppare un insieme di dati non indifferente, che ci hanno permesso di raggiungere dei risultati sorprendenti. I modelli che abbiamo realizzato hanno infatti raggiunto un accuracy superiore al 90%, dimostrando che il testo generato da un essere umano ha un valore inequivocabile e non è sostituibile da un testo generato da una intelligenza artificiale. In particolare, i test svolti su riconoscitori di plagio online hanno messo in evidenza la mancanza di un sistema che vada a coprire ogni ambito del riconoscimento della generazione del testo, mostrando concretamente la scarsa sensibilità nel riconoscimento del plagio nel campo delle composizioni poetiche. Coi risultati ottenuti la nostra tesi è confermata: l'essere umano è unico e la sua creatività non può (ancora) essere sostituita da una intelligenza artificiale. Siamo consapevoli che il campo dell'IA è in continua e costante crescita, ma questo lavoro rimarrà comunque come testimonianza e documentazione dello stato dell'arte fino a questo momento.

4.1.1. Performance migliori o peggiori delle attese

Le performance ottenute sono migliori di quello che speravamo di ottenere con lo svolgimento del nostro lavoro. Eravamo infatti convinti che si sarebbero potute distinguere le poesie generate da un essere umano con quelle generate da ChatGPT, ma non speravamo di raggiungere livelli di accuratezza così alti. Infatti, nonostante inizialmente avessimo pensato che il risultato non potesse essere quello da noi atteso a causa di un dataset relativamente limitato, siamo riusciti a realizzare dei modelli che grazie alla combinazione dei parametri utilizzati e combinati tra loro ci hanno permesso di raggiungere alti livelli di accuratezza.

4.2. Il metodo proposto rispetta le attese?

Il metodo proposto rispetta le attese che ci eravamo fatti e le supera. Come detto precedentemente infatti, eravamo convinti che saremmo riusciti a distinguere le poesie originali da quelle generate da un'intelligenza artificiale, ma non ci aspettavamo di raggiungere dei risultati così eccezionali, ottenendo valori di accuratezza superiori al 90%.

4.3. Limitazioni

Le principali limitazioni sono dovute alla costruzione del dataset. Infatti, non abbiamo trovato dei dataset già presenti online per quanto riguarda le poesie italiane e siamo stati costretti a crearne uno da zero.
Un altro limite è la lunghezza delle poesie generabili con ChatGPT. Essa infatti varia a seconda del numero di poesie che si chiedono di generare. Per questo motivo abbiamo cercato di essere più eterogenei possibili modificando il numero di poesie generate alla volta per ottenere un dataset il più possibile diversificato e realistico, garantendo così che la discriminazione non dipendesse dalla lunghezza delle poesie.
Mostriamo di seguito il grafico della lunghezza delle poesie in numero di parole, che come si nota non contiene particolari nette distinzioni fra i testi generati attraverso ChatGPT e quelle prese da autori umani:

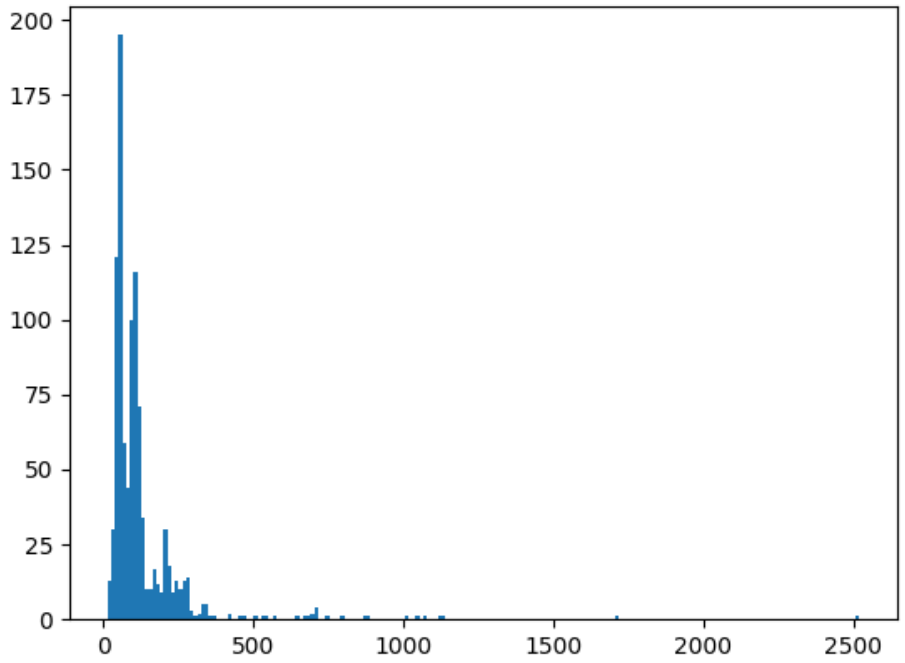

Fig. sull'asse X il numero di parole, sull'asse Y il numero di poesie.

Il nostro dataset è comunque accessibile per permettere ad altri di poter elaborare i propri modelli e metodi di preprocessing per ottenere risultati migliori.

4.3.1. Quali sono i limiti di applicabilità (bias)

Il limite di applicabilità principale è quello della lingua del dataset. Infatti, dato che eravamo interessati alla discriminazione di poesie di autori italiani, il dataset è esclusivamente italiano. Questo risulta essere il limite principale di applicabilità, in quanto i metodi di preprocessing da noi utilizzati tengono conto della lingua italiana e non inglese.

4.4. Lavori futuri

Lo scopo principale dello studio è principalmente quello di contribuire alla ricerca, quindi un possibile lavoro futuro potrebbe essere quello di continuare ad analizzare le continue nuove versioni delle intelligenze artificiali in grado di generare del testo, per poter studiare la capacità, si pensa crescente, che esse abbiano nel rendere il testo sempre più simile a quello di un essere umano. Inoltre, un possibile lavoro futuro potrebbe riguardare l'ambito della realizzazione di un software che permetta il riconoscimento del plagio di testo poetico utilizzando delle intelligenze artificiali. Infatti, come già descritto precedentemente, non è presente online un software che riesca a identificare la presenza dell'intelligenza artificiale all'interno di una poesia italiana generata tramite AI. Si potrebbe quindi pensare a un'applicazione commerciale di tale modello per poter fornire uno strumento affidabile e preciso per il riconoscimento dell'uso di AI in testi poetici, contribuendo così a preservare l'originalità e l'integrità delle opere letterarie.