Machine Learning

PROJECT

Documentazione Progetto Machine Learning

Versione	1.4
Data	*16/02/2024*
Destinatario	Dipartimento di Informatica dell'Università degli studi di Salerno
Presentato da	Cristian Porzio, Raffaele Monti

Revision History

Data	Versione	Descrizione	Autori
13/02/2024	1.0	Intestazioni del documento	Cristian Porzio, Raffaele Monti
13/02/2024	1.1	Prima stesura	Raffaele Monti, Cristian Porzio
14/02/2024	1.2	Aggiunta valori metriche di valutazione dei vari modelli	Raffaele Monti, Cristian Porzio
15/02/2024	1.3	Aggiunta conclusione	Raffaele Monti, Cristian Porzio
16/02/2024	1.4	Revisione completa	Raffaele Monti, Cristian Porzio

Ruoli e Contatti

Ruolo	Nome	Contatti
Docente	Giuseppe Polese	gpolese@unisa.it
Docente	Loredana Caruccio	lcaruccio@unisa.it
Studente	Raffaele Monti	r.monti2@studenti.unisa.it
Studente	Cristian Porzio	c.porzio3@studenti.unisa.it

Sommario

1. Introduzione

Negli ultimi anni, l'Intelligenza Artificiale (IA) ha conosciuto un'espansione senza precedenti, rivoluzionando numerosi settori e raggiungendo traguardi che fino a poco tempo fa sembravano ancora molto lontani. L'emergere di tecnologie come ChatGPT ha segnato un punto di svolta significativo, consentendo di compiere passi da gigante nell'ambito della generazione automatica di testo e di altre forme di contenuto creativo. Ciò che una volta poteva sembrare distante e irraggiungibile, ora è diventato una realtà quotidiana: generare immagini, codice, testi, musica, voci e molto altro è diventato all'ordine del giorno usando semplici prompt sui giusti modelli.

Tuttavia, insieme a questi incredibili progressi, sono emersi anche numerosi problemi e sfide. Tra questi, spiccano i cosiddetti "Deepfake", cioè video o immagini manipolati in maniera così realistica da risultare indistinguibili dall'originale, con potenziali conseguenze devastanti in termini di manipolazione dell'opinione pubblica, frode e violazione della privacy. Inoltre, l'ampia disponibilità di modelli di IA per la generazione di testo ha sollevato preoccupazioni riguardo all'etica e all'integrità accademica, poiché sempre più spesso vengono utilizzati per superare compiti ed esami senza sforzo.

Un'altra questione critica è rappresentata dalla diffusione di contenuti generati automaticamente su larga scala, che può comportare problemi di autenticità e originalità, minando la credibilità delle informazioni presentate e confondendo il confine tra ciò che è reale e ciò che è creato artificialmente. Questo fenomeno ha implicazioni profonde non solo nel campo dell'informazione e della comunicazione, ma anche in settori quali la cultura, l'arte e l'intrattenimento.

Questo solleva la questione fondamentale: come possiamo distinguere tra il lavoro umano e quello generato dall'IA? Ovviamente usando altra IA. Infatti, essa stessa può fornire soluzioni a questi problemi. Utilizzando altre tecnologie intelligenti, possiamo sviluppare sistemi di verifica in grado di riconoscere la provenienza dei contenuti e garantire l'autenticità delle informazioni. In questo modo, possiamo sfruttare l'intelligenza artificiale per combattere gli effetti negativi della stessa tecnologia, aprendo la strada a un utilizzo più etico e responsabile delle sue potenzialità.

Nota: Questo testo è stato generato dall'IA, evidenziando la facilità con cui è possibile creare contenuti utilizzando tali strumenti e la sfida nel distinguere tra testi generati artificialmente e quelli scritti da esseri umani.

2. Data Collection

I dati sono uno dei punti fondamentali per il corretto addestramento del nostro modello; infatti, per il corretto riconoscimento dei testi generati tramite IA, abbiamo bisogno di tantissimi esempi di testi scritti sia da persone reali sia generati tramite modelli di IA. Inoltre, per ogni esempio è necessario che sia noto l'autore.

Individuati i dati, prima del loro utilizzo vanno analizzati ed in alcuni casi manipolati e adattati.

2.1. Comprensione e Identificazione Dati Necessari

Al fine di produrre un modello di classificazione, il nostro dataset deve possedere:

- Un quantitativo sufficiente di dati etichettati in maniera consistente.
- Testi di lunghezza e complessità variabile.
- (Facoltativo) Contesto in cui il testo è stato scritto.
    - Esempio: il prompt sottomesso all'AI generatrice.
- (Facoltativo) Sorgente del testo scritto.
    - Esempio: quale modello di AI ha prodotto tale testo oppure da quale libro è stata presa tale citazione.

2.2. Dataset Individuati

Sono stati individuati due dataset su kaggle.com con le caratteristiche ricercate:

- Dataset 1
    - Questo dataset è composto da circa 500 mila testi, e rispetta i primi due punti sopracitati.
    - Inoltre, come l'autore "*Shayan Gerami*" descrive, questo dataset contiene dati da molteplici fonti e senza duplicati.
    - I testi sono generati o da "Human" o da "AI", senza informazioni inerenti al modello o al prompt usato per la generazione del testo.
- Dataset 2
    - Questo secondo dataset è composto da 800 mila testi, e rispetta tutti i punti sopracitati.
    - Infatti, oltre al testo e all'autore, sono presenti anche il nome dei modelli IA usati per la generazione e il prompt usato.
    - Infine, come l'autore "*Zachary Grinberg*" afferma, questo dataset è l'unione di molteplici dataset.

2.3. Data Exploration

2.3.1. Confronto dei Dataset [Grafici]

Tutti i grafici relativi al nostro modello e ai dataset utilizzati sono reperibili nel documento allegato, oppure, al seguente link.

2.3.2. Confronto dei Dataset

- Dataset 1
  - Numero di elementi di tipo 'AI': 181’438
  - Numero di elementi di tipo 'Human': **305’797**
  - Vi è una predominanza di testi compresi tra 10^2 e 10^4 parole.
  - Parola predominante nel dataset: 'students'
  - Parola predominante dei campioni umani: 'would'
  - Parola predominante dei campioni AI: ' students '
- Dataset 2
  - Numero di elementi di tipo 'AI': **441’230**
  - Numero di elementi di tipo 'Human': 347’692
  - Vi è una predominanza di testi compresi tra 10^2 e 10^4, scemando, vi sono testi di oltre 10^4 e fino a 10^6 parole.
  - Parola predominante nel dataset: 'also'
  - Parola predominante dei campioni umani: 'people'
  - Parola predominante dei campioni AI: 'also'

2.3.3. Considerazioni

- Il Dataset 1 possiede una predominanza di sample 'Human', precisamente 124’359 in più.
- Il Dataset 2 possiede una predominanza di sample 'AI', precisamente 93’538 in più.
- Il Dataset 2 risulta contenere frasi di lunghezza più variabile tra loro.
- Il Dataset 2 non identifica i testi semplicemente come scritti da IA bensì ne specifica l'IA che lo ha generato. Il 7% dei testi sono scritti con GPT-3.5, gli altri in percentuali inferiori al 2% con ulteriori IA, contribuendo alla varietà dei dati.
- Entrambi i dataset sono datati al Gennaio 2024 e non prevedono manutenzione od aggiornamenti in futuro.

3. Model Selection

Per poter dunque individuare l'origine del testo, oltre che i dati, dobbiamo scegliere un algoritmo in grado di poter imparare dai dati e classificare i testi per determinare se sono stati scritti da persone o generati da IA.

Dalla letteratura scientifica a riguardo, abbiamo individuato due varianti di modelli di Naïve Bayes che vengono utilizzati in tale ambito e, per curiosità abbiamo deciso di confrontare tali modelli "più efficaci" con un modello Tree Classifier che dovrebbe essere "svantaggiato". Di seguito verranno elencati i loro casi d'uso ed i punti di forza per questo task di classificazione in particolare:

3.1. Multinomial Naïve Bayes

È una variante dell'algoritmo Naïve Bayes che presuppone che le caratteristiche seguano una distribuzione multinomiale, ovvero una distribuzione di probabilità discreta su più categorie. Calcola la probabilità di una classe data una serie di caratteristiche applicando il teorema di Bayes. È particolarmente utile per i dataset in cui le caratteristiche possono essere contate, come le frequenze delle parole nei documenti di testo.

- **Assunzione di indipendenza**: Il termine "Naïve" di Naïve Bayes si riferisce all'ipotesi che le caratteristiche siano indipendenti l'una dall'altra, data l'etichetta della classe. Questo semplifica il calcolo delle probabilità e può portare a classificazioni accurate, soprattutto quando questa ipotesi è valida nel contesto di stili di scrittura umani o IA.
- **Caratteristiche discrete**: L'algoritmo funziona meglio con caratteristiche che possono essere contate, come la frequenza delle parole o dei caratteri, che si allinea bene con i dati di testo.
- **Classificazione del testo**: Multinomial Naïve Bayes è ampiamente utilizzato per compiti di classificazione del testo, come la categorizzazione dei documenti, che prevede l'assegnazione del testo a classi diverse in base al contenuto.
- **Caso d'uso**: L'algoritmo è stato progettato per compiti di classificazione di testi.

3.2. Complement Naïve Bayes

Questo classificatore è un'estensione del metodo Naïve Bayes progettato per gestire insiemi di dati sbilanciati. A differenza del Naïve Bayes standard, che seleziona la classe con la probabilità più alta, il CNB seleziona la classe con la probabilità più bassa. Ciò lo rende adatto a situazioni in cui una classe è molto meno frequente delle altre. Viene spesso utilizzato in compiti di classificazione di testi in cui la classe positiva (ad esempio, le e-mail di spam) è rara rispetto alla classe negativa (le e-mail non di spam).

- **Assunzione di indipendenza**: Il termine "Naïve" di Naïve Bayes si riferisce all'ipotesi che le caratteristiche siano indipendenti l'una dall'altra, data l'etichetta della classe. Questo semplifica il calcolo delle probabilità e può portare a classificazioni accurate, soprattutto quando questa ipotesi è valida nel contesto di stili di scrittura umani o IA.
- **Insiemi di dati sbilanciati**: CNB è progettato specificamente per gestire insiemi di dati sbilanciati, calcolando la probabilità che un elemento non appartenga a una certa classe, piuttosto che la probabilità che appartenga a quella classe. Questo approccio può contribuire a mitigare la distorsione verso la classe maggioritaria che spesso si verifica con i metodi Naïve Bayes tradizionali.
- **Approccio complementare**: Selezionando la classe con la probabilità più bassa (di non appartenervi), CNB sceglie effettivamente la classe che ha la probabilità più alta di essere la corrispondenza corretta, il che è utile quando la classe positiva è rara.
- **Prestazioni**: Secondo la letteratura, la CNB ha dimostrato di superare sia la Gaussiana Naïve Bayes che la Multinomiale Naïve Bayes in compiti di classificazione di testi, in particolare su insiemi di dati sbilanciati.
- **Caso d'uso**: La classificazione del testo, compresa la distinzione tra testo scritto dall'uomo e testo generato dall'intelligenza artificiale, è una delle aree in cui il CNB eccelle. È particolarmente utile quando il set di dati è sbilanciato, una situazione comune in questi problemi di classificazione.

3.3. Decisional Tree Classifier

Questo algoritmo costruisce un albero binario di decisioni basate sui valori delle caratteristiche. Suddivide i dati in modo ricorsivo in base all'importanza delle caratteristiche fino a quando non viene soddisfatto un criterio di arresto, come la profondità massima dell'albero o la dimensione minima dei nodi. Gli alberi decisionali sono interpretabili e possono catturare relazioni complesse tra le caratteristiche e la classe di riferimento. Sono anche in grado di gestire relazioni non lineari e interazioni tra le caratteristiche. Tuttavia, possono essere inclini all'overfitting, soprattutto quando l'albero è profondo.

- **Struttura ad albero**: Gli alberi decisionali sono facili da interpretare e visualizzare e possono essere utili per comprendere il processo decisionale del classificatore.
- **Gestione di dati sbilanciati**: Se il set di dati è sbilanciato, con molti più esempi di una classe (scritti dall'uomo o generati dall'intelligenza artificiale), l'albero decisionale potrebbe essere sbilanciato verso la classe maggioritaria, a meno che non si utilizzino tecniche adeguate come il sovracampionamento, il sottocampionamento o l'apprendimento sensibile ai costi.
- **Relazioni non lineari**: Sebbene gli alberi decisionali siano in grado di gestire relazioni non lineari in una certa misura, potrebbero non catturare modelli complessi con la stessa efficacia di altri classificatori come le Random Forest o le macchine di Gradient Boosting (altri modelli più performanti applicati al nostro task).
- **Caso d'uso**: L'uso di un classificatore ad albero decisionale per determinare se i testi sono scritti dall'uomo o generati dall'intelligenza artificiale può essere un approccio valido, anche se potrebbe non essere il metodo più efficiente o preciso rispetto ad altri classificatori.

4. Data Manipulation & Execution

4.1. Preprocessing & Feature-Selection

Fortunatamente, entrambi i dataset godono di un'altissima qualità, infatti contengono testi di svariate lunghezze, complessità, privi di valori sconosciuti e tutti etichettati. Dunque, ci siamo limitati a selezionare le feature che ci interessavano e li abbiamo adattati per l'esecuzione sullo stesso codice sorgente, andando a rinominare i campi delle colonne. In particolare:

- Nel Dataset 1, non avendo ulteriori campi oltre '*text*' e '*generated*' ci siamo limitati al rietichettamento dei valori '*0*' ed '*1*' della colonna *'generated'* rispettivamente in *'Human'* e *'AI'*.
- Nel Dataset 2, i campi superflui al training (prompt_id, text_length, word_count) sono stati eliminati. Inoltre, abbiamo provveduto a rietichettare come *'AI'* tutti i nomi di modelli generativi differenti da *'Human'* nella colonna *'source'*.
- Sono stati eseguiti ulteriori training sugli stessi dataset bilanciando le features in parti uguali, ottenendo dataset di rispettivamente 2*181'438 e 2*347'692 samples.
- Per ciascun dataset ottenuto, sono state fatte ignorare le stop words (particelle, punteggiature, congiunzioni...).

4.2. Feature-Extraction

È stata utilizzata la libreria "*TfidVectorizer*" di scikit-learn per effettuare la conversione dei testi presenti nei dataset in una matrice di feature TF-IDS (Term Frequency-Inverse Document Frequency). In particolare, la libreria esegue i seguenti passaggi:

1. **Tokenizzazione e conteggio delle parole**: Suddivide ogni testo in parole o token e conta quante volte appare ciascun token in ogni testo.
    - È possibile dichiarare di ignorare le stop words, come nel nostro caso.
2. **Calcolo del TF-IDF**: Calcola il valore TF-IDF per ogni termine in ogni testo.
   Il TF-IDF di un termine è una misura della sua importanza relativa in un testo rispetto a un insieme di testi. È composto da due componenti:
    - TF (Term Frequency): Valuta la frequenza con cui una parola appare in un testo. Più una parola appare spesso, più è rilevante per quel testo.
    - IDF (Inverse Document Frequency): Valuta quanto una parola è unica nell'insieme dei testi. Più una parola è unica, più alto sarà il suo peso. Parole comuni che compaiono in molti testi hanno un peso più basso.
3. **Normalizzazione dei vettori**: Normalizza i vettori delle caratteristiche per assicurarsi che siano comparabili e non influenzati dalla lunghezza dei testi.
4. **Costruzione della matrice dei TF-IDF**: Utilizza i valori TF-IDF calcolati per costruire una matrice in cui ogni riga rappresenta un testo e ogni colonna rappresenta un termine, con valori TF-IDF associati.

Ciò permette di poter gestire l'enorme quantità di testo contenuti nei dataset rendendo efficiente l'estrazione delle feature utili dai vari testi.

4.3. Pipeline

Sono state sviluppate 3 pipeline, una per ogni modello (Multinomial NB, Complement NB, Decision Tree Classifier), tutte addestrate sui dataset suddivisi seguendo il "Principio di Pareto", ovvero suddividendo il dataset in due porzioni, 80% per l'addestramento e 20% per il testing. Nella sezione successiva, sono presenti i risultati e le valutazioni che abbiamo tratto per ognuna delle pipeline.

Il link al notebook in cui è presente il codice del nostro modello è reperibile qui.

5. Evaluation ed Analisi Comparativa

5.1. Metriche di Valutazione

Per ciascuna esecuzione dei modelli verranno calcolati:

1. **Train Time**: misura quanto tempo è necessario per allenare il modello.
2. **Precision**: misura quanto bene il modello è in grado di evitare falsi positivi. È il rapporto tra i veri positivi (TP) e tutti i predetti positivi (TP + FP).
3. **Recall**: misura quanto bene il modello riesce a trovare tutti i casi positivi reali. È il rapporto tra i veri positivi (TP) e il numero totale di veri positivi e falsi negativi (FN). Recall è importante quando è necessario assicurarsi che non si perdano esempi positivi
4. **F1-Score**: è la media armonica di precisione e recall. Fornisce un equilibrio tra essere specifici ed essere completi, che è fondamentale quando si ha uno sbilanciamento delle classi.
5. **Support**: Support rappresenta il numero di istanze vere per ogni classe nel dataset. È utile per capire la distribuzione delle classi nel dataset e per confrontare i risultati tra modelli che gestiscono classi di dimensioni diverse.
6. **Accuracy**: è la proporzione di previsioni corrette rispetto al totale delle previsioni.
7. **Macro average**: calcola la media non pesata delle metriche per tutte le classi. Ogni classe viene trattata in modo indipendente e i risultati vengono poi mediati. È utile quando si desidera un'analisi delle prestazioni del modello per ogni singola classe.
8. **Weighted average**: calcola la media ponderata delle metriche per tutte le classi. Ogni classe viene pesata in base al suo numero di istanze nel dataset e i risultati vengono poi mediati. È utile quando si vuole dare più peso alle classi più numerose
9. **ROC Curve:** serve a valutare la performance di un modello di classificazione binaria. Mostra il trade-off tra il tasso di veri positivi (sensitivity o recall) e il tasso di falsi positivi (fall-out) per vari valori di soglia di classificazione.
10. **Confusion Matrix**: È una tabella che mostra le prestazioni di un modello di classificazione attraverso quattro metriche principali: veri positivi (TP), veri negativi (TN), falsi positivi (FP) e falsi negativi (FN). è utile per comprendere dove il modello sta commettendo errori e dove sta ottenendo risultati accurati.

Glossario Acronimi	
M-NB	Multinomial Naive Bayes
C-NB	Complement Naive Bayes
DTC	Decision Tree Classifier
DTC**[x]**	x = profondità dell'albero
H	Human
AI	Artificial Intelligence
ROC	Receiver Operating Characteristic

5.2. Risultati

5.2.1. Tabella 1

Tipo Dataset	Dataset 1 Non Bilanciato			Dataset 1 Bilanciato		
Modello	**M-NB**	**C-NB**	**DTC[5]**	**M-NB**	**C-NB**	**DTC[5]**
Train Δ	161s	155s	218s	118s	116s	156s
Precision	AI:97 H: 95	AI:96 H: 96	AI:87 H: 79	AI:96 H: 94	AI:96 H: 94	AI:84 H: 74
Recall	AI:91 H: 99	AI:93 H: 98	AI:58 H: 95	AI:94 H: 97	AI:94 H: 97	AI:70 H: 87
F1-Score	AI:94 H: 97	AI:95 H: 97	AI:69 H: 86	AI:95 H: 95	AI:95 H: 95	AI:76 H: 80
Support	36373 61074	36373 61074	36373 61074	36153 36423	36153 36423	36153 36423
Accuracy	95.59%	96.06%	80.99%	95.33%	95.32%	78.56%
Macro avg	0.96	0.96	0.83	0.95	0.95	0.79
Weighted avg	0.86	0.96	0.82	0.95	0.95	0.79

5.2.2. Tabella 2

Tipo Dataset	Dataset 2 Non Bilanciato			Dataset 2 Bilanciato		
Modello	**M-NB**	**C-NB**	**DTC[5]**	**M-NB**	**C-NB**	**DTC[5]**
Train Δ	385s	385s	573s	409s	413s	563s
Precision	AI:79 H: 84	AI:82 H: 80	AI:67 H: 87	AI:82 H: 79	AI:82 H: 79	AI:62 H: 89
Recall	AI:90 H: 70	AI:85 H: 76	AI:95 H: 42	AI:78 H: 83	AI:78 H: 83	AI:95 H: 42
F1-Score	AI:84 H: 77	AI:83 H: 78	AI:79 H: 56	AI:80 H: 81	AI:80 H: 81	AI:75 H: 57
Support	8045 69740	8045 69740	8045 69740	69668 69409	69668 69409	69668 69409
Accuracy	81.05%	81.18%	71.49%	80.51%	80.51%	68.61%
Macro avg	0.82	0.81	0.77	0.81	0.81	0.76
Weighted avg	0.81	0.81	0.76	0.81	0.81	0.76

Sono stati inoltre raccolti i dati per l'esecuzione del modello 'Decision Tree Classifier' a diversi livelli massimi di profondità, partendo dal minimo di 2 al massimo (permesso dalla libreria utilizzata) di 20:

5.2.3. Tabella 3

Tipo Dataset	Dataset 1 Non Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:90 H: 73	AI:87 H: 76	AI:86 H: 79	AI:82 H: 84	AI:86 H: 85	AI:97 H: 94
Recall	AI:38 H: 98	AI:49 H: 96	AI:58 H: 94	AI:71 H: 91	AI:73 H: 93	AI:89 H: 98
F1-Score	AI:54 H: 83	AI:63 H: 85	AI:69 H: 86	AI:76 H: 87	AI:79 H: 89	AI:92 H: 96
Support	36275 61172	36275 61172	36275 61172	36275 61172	36275 61172	36275 61172
Accuracy	75.49%	78.34%	80.70%	83.23%	85.58%	94.58%
Macro avg	0.81	0.82	0.82	0.83	0.86	0.95
Weighted avg	0.79	0.80	0.81	0.83	0.86	0.95

5.2.4. Tabella 4

Tipo Dataset	Dataset 1 Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:74 H: 69	AI:76 H: 74	AI:80 H: 74	AI:78 H: 85	AI:81 H: 88	AI:95 H: 94
Recall	AI:64 H: 78	AI:73 H: 76	AI:71 H: 83	AI:86 H: 76	AI:89 H: 79	AI:94 H: 95
F1-Score	AI:69 H: 73	AI:75 H: 75	AI:76 H: 78	AI:82 H: 80	AI:85 H: 83	AI:94 H: 95
Support	36138 36438	36138 36438	36138 36438	36138 36438	36138 36438	36138 36438
Accuracy	70.94%	74.98%	77.09%	81.16%	84.13%	94.45%
Macro avg	0.71	0.75	0.77	0.81	0.85	0.94
Weighted avg	0.71	0.75	0.77	0.81	0.85	0.94

5.2.5. Tabella 5

Tipo Dataset	Dataset 2 Non Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:63 H: 96	AI:65 H: 95	AI:65 H: 95	AI:69 H: 84	AI:70 H:85	AI:74 H:87
Recall	AI:99 H: 26	AI:99 H: 33	AI:99 H: 33	AI:93 H: 47	AI:93 H: 49	AI:93 H: 59
F1-Score	AI:77 H: 41	AI:78 H: 49	AI:78 H: 49	AI:79 H: 60	AI:80 H: 62	AI:83 H: 70
Support	88268 69517	88268 69517	88268 69517	88268 69517	88268 69517	88268 69517
Accuracy	66.88%	69.64%	69.64%	72.59%	73.77%	78.14%
Macro avg	0.80	0.80	0.80	0.76	0.77	0.81
Weighted avg	0.78	0.78	0.78	0.76	0.78	0.80

5.2.6. Tabella 6

Tipo Dataset	Dataset 2 Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:57 H: 97	AI:60 H: 96	AI:60 H: 97	AI:64 H: 86	AI:64 H: 88	AI:72 H: 82
Recall	AI:99 H: 26	AI:99 H: 33	AI:99 H: 33	AI:92 H: 48	AI:94 H: 48	AI:86 H: 66
F1-Score	AI:73 H: 41	AI:74 H: 49	AI:74 H: 49	AI:76 H: 62	AI:76 H: 62	AI:78 H: 73
Support	69561 69516	69561 69516	69561 69516	69561 69516	69561 69516	69561 69516
Accuracy	62.77%	65.81%	65.83%	70.14%	70.64%	76.01%
Macro avg	0.77	0.78	0.78	0.75	0.76	0.77
Weighted avg	0.77	0.78	0.78	0.75	0.76	0.77

NOTA: i grafici delle singole esecuzioni, sono presenti nel documento allegato, o in alternativa, sono stati allegati qui per questioni di leggibilità

6. Conclusioni

Abbiamo riesaminato tutte le tabelle dei risultati, evidenziando quali fossero, per ogni dataset e per ogni modello, in verde i risultati migliori, e in rosso i risultati peggiori per ogni metrica di valutazione. Per semplicità, tutte le metriche hanno lo stesso peso. Abbiamo infine sommato i punteggi ottenuti dai vari modelli nelle varie casistiche per ottenere un punteggio finale del modello:

Tipo Dataset	Dataset 1 Non Bilanciato			Dataset 1 Bilanciato		
Modello	**M-NB**	**C-NB**	**DTC[5]**	**M-NB**	**C-NB**	**DTC[5]**
Train Δ	161s	155s	218s	118s	116s	156s
Precision	AI:97 H: 95	AI:96 H: 96	AI:87 H: 79	AI:96 H: 94	AI:96 H: 94	AI:84 H: 74
Recall	AI:91 H: 99	AI:93 H: 98	AI:58 H: 95	AI:94 H: 97	AI:94 H: 97	AI:70 H: 87
F1-Score	AI:94 H: 97	AI:95 H: 97	AI:69 H: 86	AI:95 H: 95	AI:95 H: 95	AI:76 H: 80
Support	36373 61074	36373 61074	36373 61074	36153 36423	36153 36423	36153 36423
Accuracy	95.59%	96.06%	80.99%	95.33%	95.32%	78.56%
Macro avg	0.96	0.96	0.83	0.95	0.95	0.79
Weighted avg	0.86	0.96	0.82	0.95	0.95	0.79
Punti +	4	6	0	2	3	0
Punti -	0	0	3	0	0	7
Totale	4	6	-3	2	3	-7
Voti Finali	M-NB = 6		C-NB = 9		DTC[5] = -10	

Dalla Tabella 1:

- Dataset 1 non Bilanciato:
  - M-NB ha la precision più alta (97%) per l'intelligenza artificiale (AI), ma C-NB ha la precision più alta (96%) per l'uomo (H).
  - M-NB il recall più alta (99%) per H.
  - M-NB e C-NB hanno l'F1-Score più alto (97%) per H, mentre C-NB ha l'F1-Score più alto (95%) per AI.
  - C-NB ha l'accuracy più alta (96,06%).
- Dataset 1 Bilanciato:
  - M-NB e C-NB hanno il recall più alto (94%) per entrambi AI.
  - M-NB e C-NB hanno l'F1-Score più alto (95%) per entrambi AI.

- Conclusioni:
  - In generale, M-NB e C-NB si comportano in modo altamente migliore rispetto a DTC in entrambe le casistiche di dataset bilanciato e non bilanciato.
  - In definitiva, il C-NB si è dimostrato il miglior modello per il dataset 1.

Tipo Dataset	Dataset 2 Non Bilanciato			Dataset 2 Bilanciato		
Modello	**M-NB**	**C-NB**	**DTC[5]**	**M-NB**	**C-NB**	**DTC[5]**
Train Δ	385s	385s	573s	409s	413s	563s
Precision	AI:79 H: 84	AI:82 H: 80	AI:67 H: 87	AI:82 H: 79	AI:82 H: 79	AI:62 H: 89
Recall	AI:90 H: 70	AI:85 H: 76	AI:95 H: 42	AI:78 H: 83	AI:78 H: 83	AI:95 H: 42
F1-Score	AI:84 H: 77	AI:83 H: 78	AI:79 H: 56	AI:80 H: 81	AI:80 H: 81	AI:75 H: 57
Support	8045 69740	8045 69740	8045 69740	69668 69409	69668 69409	69668 69409
Accuracy	81.05%	81.18%	71.49%	80.51%	80.51%	68.61%
Macro avg	0.82	0.81	0.77	0.81	0.81	0.76
Weighted avg	0.81	0.81	0.76	0.81	0.81	0.76
Punti +	4	4	1	4	4	2
Punti -	0	0	4	1	1	6
Totale	4	4	-3	3	3	-4
Voti Finali	M-NB = 7		C-NB = 7		DTC[5] = -7	

Dalla Tabella 2:

- Dataset 2 non Bilanciato:
  - C-NB ha la precision più alta (82%) per AI.
  - DTC[5] ha il recall più alto (95%) per AI.
  - M-NB ha l'F1-Score più alto (84%) per AI.
  - C-NB ha l'accuracy più alta (81.18%)
- Dataset 2 Bilanciato:
  - M-NB e C-NB hanno la precision più alta (82%) per AI. Mentre DTC[5] ha la precision più alta (89%) per H.
  - M-NB e C-NB hanno il richiamo più alto (83%) per H.
  - M-NB e C-NB hanno l'F1-Score più alto (81%) per H.

- Conclusioni:
  - In generale, i modelli performano peggio sul Dataset 2 rispetto al Dataset 1.
  - La differenza di performance è più marcata per il Dataset 2 bilanciato.
  - C-NB e M-NB performano pressoché identica sia per dataset bilanciato che non bilanciato. Inoltre, risultano essere i migliori per il Dataset 2.

C-NB risulta essere in tutte le quattro le esecuzioni il modello con la miglior prestazione e richiede un tempo ragionevole per il training.

Per quanto riguarda invece i risultati inerenti al modello ad albero, sono qui riportati i risultati a diverse profondità massime (2, 3, 4, 6, 7, 20):

Dalla Tabella 3:

Tipo Dataset	Dataset 1 Non Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:90 H: 73	AI:87 H: 76	AI:86 H: 79	AI:82 H: 84	AI:86 H: 85	AI:97 H: 94
Recall	AI:38 H: 98	AI:49 H: 96	AI:58 H: 94	AI:71 H: 91	AI:73 H: 93	AI:89 H: 98
F1-Score	AI:54 H: 83	AI:63 H: 85	AI:69 H: 86	AI:76 H: 87	AI:79 H: 89	AI:92 H: 96
Support	36275 61172	36275 61172	36275 61172	36275 61172	36275 61172	36275 61172
Accuracy	75.49%	78.34%	80.70%	83.23%	85.58%	94.58%
Macro avg	0.81	0.82	0.82	0.83	0.86	0.95
Weighted avg	0.79	0.80	0.81	0.83	0.86	0.95
Punti +	1	0	0	0	0	9
Punti -	7	0	0	2	0	0
Totale	-6	0	0	-2	0	9

Dalla Tabella 4:

Tipo Dataset	Dataset 1 Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:74 H: 69	AI:76 H: 74	AI:80 H: 74	AI:78 H: 85	AI:81 H: 88	AI:95 H: 94
Recall	AI:64 H: 78	AI:73 H: 76	AI:71 H: 83	AI:86 H: 76	AI:89 H: 79	AI:94 H: 95
F1-Score	AI:69 H: 73	AI:75 H: 75	AI:76 H: 78	AI:82 H: 80	AI:85 H: 83	AI:94 H: 95
Support	36138 36438	36138 36438	36138 36438	36138 36438	36138 36438	36138 36438
Accuracy	70.94%	74.98%	77.09%	81.16%	84.13%	94.45%
Macro avg	0.71	0.75	0.77	0.81	0.85	0.94
Weighted avg	0.71	0.75	0.77	0.81	0.85	0.94
Punti +	0	0	0	0	0	9
Punti -	8	1	0	1	0	0
Totale	-8	-1	0	-1	0	9

Dalla Tabella 5:

Tipo Dataset	Dataset 2 Non Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:63 H: 96	AI:65 H: 95	AI:65 H: 95	AI:69 H: 84	AI:70 H:85	AI:74 H:87
Recall	AI:99 H: 26	AI:99 H: 33	AI:99 H: 33	AI:93 H: 47	AI:93 H: 49	AI:93 H: 59
F1-Score	AI:77 H: 41	AI:78 H: 49	AI:78 H: 49	AI:79 H: 60	AI:80 H: 62	AI:83 H: 70
Support	88268 69517	88268 69517	88268 69517	88268 69517	88268 69517	88268 69517
Accuracy	66.88%	69.64%	69.64%	72.59%	73.77%	78.14%
Macro avg	0.80	0.80	0.80	0.76	0.77	0.81
Weighted avg	0.78	0.78	0.78	0.76	0.78	0.80
Punti +	2	1	1	0	0	5
Punti -	6	1	1	3	1	1
Totale	-4	0	0	-3	-1	4

Dalla Tabella 6:

Tipo Dataset	Dataset 2 Bilanciato					
Modello	**DTC[2]**	**DTC[3]**	**DTC[4]**	**DTC[6]**	**DTC[7]**	**DTC[20]**
Precision	AI:57 H: 97	AI:60 H: 96	AI:60 H: 97	AI:64 H: 86	AI:64 H: 88	AI:72 H: 82
Recall	AI:99 H: 26	AI:99 H: 33	AI:99 H: 33	AI:92 H: 48	AI:94 H: 48	AI:86 H: 66
F1-Score	AI:73 H: 41	AI:74 H: 49	AI:74 H: 49	AI:76 H: 62	AI:76 H: 62	AI:78 H: 73
Support	69561 69516	69561 69516	69561 69516	69561 69516	69561 69516	69561 69516
Accuracy	62.77%	65.81%	65.83%	70.14%	70.64%	76.01%
Macro avg	0.77	0.78	0.78	0.75	0.76	0.77
Weighted avg	0.77	0.78	0.78	0.75	0.76	0.77
Punti +	2	3	4	0	0	5
Punti -	5	0	0	2	0	2
Totale	-3	3	4	-2	0	3

La migliore profondità risulta essere quasi sempre 20, tranne per il Dataset 2 bilanciato in cui 4 riesce per un punto a superarlo.

In conclusione, ci riteniamo soddisfatti dei risultati ottenuti. Grazie allo studio dei tre modelli realizzati, abbiamo sperimentato varie metodologie per classificare il testo e riconoscere quali fossero generati da IA. Senza ombra di dubbio però, il nostro studio è da considerarsi un piccolo successo per lo stato attuale delle IA. Non ci sembra impossibile pensare, infatti, che con il passare del tempo, saranno necessari metodi molto più sofisticati per riconoscere testo generato da IA, sempre se sarà possibile. Se ci proiettiamo nel futuro, sapendo come sia cambiata l'IA in questi pochi anni, ci sembra ragionevole pensare che diverrà così tanto sofisticata e realistica, ma rendersi essenzialmente indistinguibile dal testo umano. A fronte di ciò, l'unica conclusione che ci siamo dati è la speranza di risolvere il problema "alla base", ovvero la speranza che i modelli stessi inseriscano, durante la generazione del testo, uno o più valori nascosti, invisibili all'occhio del lettore, ma ampiamente riconoscibili tramite algoritmi, che permettano immediatamente l'identificazione dei testi generati. Inoltre, questa sfida va estesa anche per gli altri campi in cui l'IA cresce in modo esponenziale. Ad esempio, OpenAI Sora è un nuovo modello di generazione di video ancora in via di sviluppo. Visionando le demo video presentate, molte di queste risultano davvero indistinguibili dalla realtà. Sarà quindi, a nostro avviso, necessario e urgente un meccanismo efficace di riconoscimento in tutti i campi in cui l'IA generi contenuti, siano essi audio, video, codice, testo o qualsiasi altra cosa.

Grazie infinite dell'attenzione.