
ANONIMIZZAZIONE DI DATI SANITARI TRAMITE TECNICHE DI ML ED NLP

Progetto combinato: FDSML ed NLP

Authors

Consiglio Luigi (0522501894)
Ferrara Francesco (0522501959)
UNISA
2024-2025

Contents

1	Introduzione	4
1.1	Descrizione del problema	4
1.2	Research Question	4
2	Analisi Esplorativa del Dataset	5
2.1	Descrizione del dataset	5
2.2	Caricamento del dataset	6
2.3	Identificazione dei valori null e delle celle vuote	6
2.4	Identificazione di eventuali duplicati	6
3	Data Cleaning	6
3.1	Rimozione dei duplicati	6
3.2	Normalizzazione della colonna Name	6
3.3	Analisi di righe con una sola feature di differenza	7
3.3.1	Analisi della distribuzione delle etichette da predire	8
3.4	Rimozione della colonna Room Number	9
4	Statistica descrittiva	10
4.1	Statistiche delle variabili numeriche	10
4.2	Statistiche delle variabili categoriche	10
4.3	Boxplot Billing Amount ed Age	11
4.4	Encoding	12
4.4.1	Categorizzazione della colonna età	13
4.4.2	One Hot Encoding	13
4.4.3	Label Encoding	13
4.4.4	Feature Selection: ANOVA F-test	13
4.5	Statistiche sulle variabili dopo l'encoding	15
4.5.1	Grafici a barre	15
4.6	Statistica descrittiva multivariata	18
4.6.1	Risposta alla RQ1	18
5	Machine Learning	20
5.0.1	Creazione del test e validation set	20
5.1	Random Forest	20
5.2	Regressione Logistica	22
5.3	Support Vector Machine (SVM)	23
5.4	Naive Bayes	25
5.5	Grid Search	26
5.6	Risposta alla RQ2	27
6	Natural Language Processing	28
6.1	Creazione delle note cliniche	28
6.1.1	Generalizzazione di Age e Billing Amount	29
6.2	Anonimizzazione dei dati sensibili	29
6.2.1	Scelte applicate	29

6.3	Gold Dataset	30
6.4	Metriche utilizzate	30
6.5	Anonimizzazione tramite RegEx	31
6.5.1	Risposta alla RQ3	32
6.6	Anonimizzazione Tramite NER	32
6.6.1	Utilizzo di Spacy	32
6.6.2	Utilizzo di Bert	35
6.6.3	Utilizzo di Flair	36
6.7	Confronto tra i vari modelli	37
6.7.1	Risposta alla RQ4	38
6.8	Sviluppi futuri	38

1 Introduzione

La seguente documentazione è corredata dal file .ipynb, dov'è presente il codice python con il relativo output.

1.1 Descrizione del problema

La medicina sta diventando sempre più data-driven, con dati provenienti da cartelle cliniche elettroniche, imaging medico (radiografie, TAC, MRI), wearables e sensori remoti, e testi clinici non strutturati. Questi dati supportano diagnosi precoci, personalizzazione delle terapie, prevenzione e monitoraggio dei pazienti e ottimizzazione delle risorse sanitarie.

Il problema dell'**anonimizzazione dei dati** riguarda la trasformazione di dataset contenenti informazioni personali o sensibili in una forma che impedisca di risalire all'identità degli individui, pur mantenendo il più possibile l'utilità dei dati per analisi statistiche o applicazioni di machine learning.

La sfida principale sta nel bilanciare due esigenze spesso in conflitto:

- **Protezione della privacy** → evitare che dati come nome, indirizzo, codice fiscale o informazioni sanitarie possano essere ricondotti a una persona specifica.
- **Utilità dei dati** → preservare abbastanza dettaglio e struttura da permettere analisi accurate, senza distorsioni eccessive.

Anche dopo l'anonimizzazione, esiste il rischio di **re-identificazione** incrociando i dati con altre fonti. Per questo vengono usate tecniche come pseudonimizzazione, generalizzazione, randomizzazione.

L'**obiettivo** di questo progetto è analizzare e confrontare diverse tecniche di anonimizzazione dei dati sensibili e valutarne l'impatto sulle performance dei modelli predittivi.

1.2 Research Question

Questo progetto si propone di rispondere a 4 domande di ricerca principali:

Q RQ₁. *Quali fattori influenzano maggiormente l'esito dei test clinici?*

Q RQ₂. *Come si comportano i principali modelli di Machine Learning di fronte al dataset scelto?*

Q RQ₃. *Quanto sono efficaci le regular expression nell'anonimizzare automaticamente le note cliniche generate?*

Q RQ₄. *Quanto è efficace il Named Entity Recognition (NER) nell'anonimizzare automaticamente le note cliniche generate?*

2 Analisi Esplorativa del Dataset

2.1 Descrizione del dataset

Il dataset presente al link **Healthcare Dataset** è stato creato per essere una risorsa utile per gli appassionati di data science, machine learning e analisi dei dati. È progettato per imitare i dati sanitari del mondo reale, consentendo agli utenti di praticare, sviluppare e mostrare le loro abilità di manipolazione e analisi dei dati nel contesto del settore sanitario. Ogni colonna fornisce informazioni specifiche sul paziente, sul suo ricovero e sui servizi sanitari ricevuti, rendendo questo dataset adatto a diversi compiti di analisi dei dati e modellazione nel settore sanitario. Ecco una breve spiegazione di ciascuna colonna del dataset:

- **Name:** rappresenta il nome del paziente associato al record sanitario.
- **Age:** l'età del paziente al momento del ricovero, espressa in anni.
- **Gender:** indica il genere del paziente, "Maschio" o "Femmina".
- **Blood Type:** il gruppo sanguigno del paziente, che può essere uno dei gruppi comuni (es. "A+", "O-", ecc.).
- **Medical Condition:** specifica la principale condizione medica o diagnosi del paziente, come "Diabete", "Ipertensione", "Asma", ecc.
- **Date of Admission:** la data in cui il paziente è stato ricoverato presso la struttura sanitaria.
- **Doctor:** il nome del medico responsabile delle cure del paziente durante il ricovero.
- **Hospital:** identifica la struttura sanitaria o l'ospedale dove il paziente è stato ricoverato.
- **Insurance Provider:** indica il fornitore dell'assicurazione del paziente, tra diverse opzioni come "Aetna", "Blue Cross", "Cigna", "UnitedHealthcare" e "Medicare".
- **Billing Amount:** l'ammontare in denaro fatturato per i servizi sanitari ricevuti durante il ricovero, espresso come numero decimale.
- **Room Number:** il numero della stanza in cui il paziente è stato ospitato durante il ricovero.
- **Admission Type:** specifica il tipo di ricovero, che può essere "Emergenza", "Elettivo" o "Urgente", a seconda delle circostanze.
- **Discharge Date:** la data in cui il paziente è stato dimesso dalla struttura sanitaria, calcolata a partire dalla data di ricovero e un numero casuale di giorni entro un intervallo realistico.
- **Medication:** identifica un farmaco prescritto o somministrato al paziente durante il ricovero, ad esempio "Aspirina", "Ibuprofene", "Penicillina", "Paracetamolo" o "Lipitor".

- **Test Results:** descrive l'esito di un test medico effettuato durante il ricovero, con valori possibili come "Normale", "Anormale" o "Inconcludente", indicando il risultato del test.

2.2 Caricamento del dataset

La prima fase del progetto richiede il caricamento del dataset all'interno dell'ambiente di sviluppo. Inoltre, si andrà ad effettuare la pulizia dei dati in modo tale da averli pronti per le successive fasi di analisi del dataset.

2.3 Identificazione dei valori null e delle celle vuote

Non sono presenti valori mancanti, il che garantisce un'analisi totale fin dall'inizio.

2.4 Identificazione di eventuali duplicati

Sono stati riscontrati 534 record duplicati che andranno rimossi nella fase di data cleaning.

3 Data Cleaning

In questa fase avviene la prima pulizia del dataset, che comprende tra le altre cose la rimozione di valori duplicati, l'eventuale modifica delle feature e la loro standardizzazione.

3.1 Rimozione dei duplicati

Dopo l'identificazione dei record duplicati, si procede con la loro rimozione dal dataset.

3.2 Normalizzazione della colonna Name

La colonna Name presenta un formato non standardizzato, ad esempio i nomi sono di questo tipo: "**zaCHArY baLL**". La successiva modifica comporterà il passaggio al formato standard in cui la prima lettera della parola è maiuscola mentre le successive minuscole: "**Zachary Ball**".

Sono stati rilevati 24121 righe con nomi duplicati. Il dataset è composto da dati sintetici, che simulano il mondo reale, quindi è normale avere tutti questi duplicati con la colonna Name. Ma questo non basta per dire che ci troviamo di fronte alla stessa persona, poichè potremmo avere casi di omonimia. Inoltre una persona può comparire in diversi record poichè potrebbe essersi recata in ospedale più volte anche per patologie diverse.

Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital
Jeffery Johnson	40	Male	B-	Asthma	2021-10-24	Laura Sherman	Davis-Arroyo
Jeffery Johnson	42	Female	O+	Obesity	2023-09-26	William Johnson	Lutz Jackson Coffey, and
Jeffery Johnson	39	Male	B-	Asthma	2021-10-24	Laura Sherman	Davis-Arroyo
Jeffrey Henry	35	Male	AB+	Diabetes	2023-04-21	George Griffin	Walker Ltd
Jeffrey Henry	31	Male	AB+	Diabetes	2023-04-21	George Griffin	Walker Ltd
John Pugh	29	Male	O+	Hypertension	2023-03-24	Brian Miller	Fisher Ltd

Figure 1: Record ridondanti

Come mostrato nella tabella, ci sono casi in cui le variazioni tra omonimi sono insignificanti, come ad esempio Kelly Matthews: infatti si ha che, quando la paziente è donna l'unica variazione tra le due entry del dataset è relativa alla differenza di età, che nello specifico è di 1 anno. Una differenza così piccola indica che ci troviamo, di fatto, di fronte alla stessa persona.

3.3 Analisi di righe con una sola feature di differenza

Questa sezione mira a comprendere quanti record all'interno del dataset presentano una differenza di una sola colonna, e quindi di fatto mostrano lo stesso individuo duplicato diverse volte. Ciò è propedeutico per decidere se questi record in più possono essere lasciati oppure se possono essere eliminati.

L'analisi appena condotta si basa sulle caratteristiche che nel corso della vita sono soggette ad una variazione prossima allo 0: usare Name, Gender e Blood Type in maniera combinata serve per capire quante persone presentano un'uguaglianza sostanziale.

L'analisi condotta anche su altri fattori, come il medico di riferimento non avrebbe avuto lo stesso impatto in quanto è plausibile che una persona possa avere diverse patologie o problematiche e per questo decidere di farsi curare da un medico diverso.

Se all'analisi precedente si aggiunge anche la colonna dell'età, si ha che la variabilità cresce moltissimo, tanto da ridurre la possibilità di trovarsi di fronte alla stessa persona.

Aumentare il numero di colonne per effettuare il confronto farà ridurre ancora di più il numero di persone duplicate all'interno del dataset.

L'analisi sin qui condotta mostra come, basandoci sulle caratteristiche di Name, Gender e Blood Type, le persone che presentano tali caratteristiche in comune, ossia sono con una buona probabilità la stessa persona, e che hanno di dissimile solo una feature sono 4753.

Name	Age	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider
Aaron Archer	47	Female	B-	Cancer	2021-01-10	Cynthia Villanueva	Montes Case and Mendez,	Medicare
Aaron Archer	49	Female	B-	Cancer	2021-01-10	Cynthia Villanueva	Montes Case and Mendez,	Medicare
Aaron Carr	59	Female	O-	Asthma	2023-06-20	Diane Davis	Jones, Holmes Kelley and	Blue Cross
Aaron Carr	60	Female	O-	Asthma	2023-06-20	Diane Davis	Jones, Holmes Kelley and	Blue Cross
Aaron Dalton	25	Male	O+	Arthritis	2022-08-22	Sarah Adams	Schroeder PLC	Blue Cross
Aaron Dalton	26	Male	O+	Arthritis	2022-08-22	Sarah Adams	Schroeder PLC	Blue Cross
Aaron Davis	75	Male	O-	Asthma	2023-01-26	John Reyes	Lee-Brown	Blue Cross
Aaron Davis	77	Male	O-	Asthma	2023-01-26	John Reyes	Lee-Brown	Blue Cross

Figure 2: Differenze nella colonna Age

Da questo sample di dataset, si vede che le persone che presentano quelle 3 caratteristiche in comune e che hanno una sola feature di differenza, sono per la maggior parte, o la quasi totalità, persone che si differenziano esclusivamente per l'età.

3.3.1 Analisi della distribuzione delle etichette da predire

Prima di procedere con l'eliminazione delle righe appena analizzate, verifichiamo la distribuzione delle etichette per comprendere se il dataset risulta bilanciato oppure no.

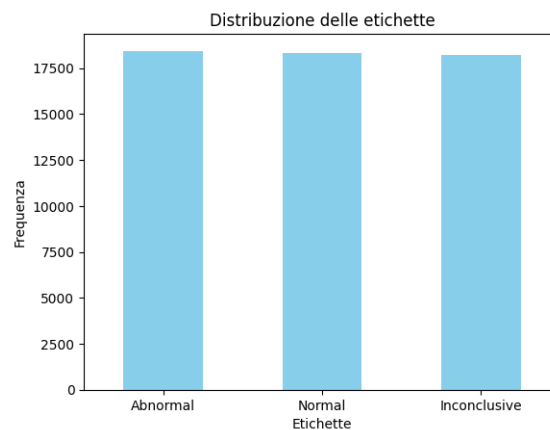


Figure 3: Differenze nella colonna Age

A fronte di questi dati, la distribuzione delle etichette risulta fortemente bilanciata. A questo punto si può procedere con la rimozione delle righe con una sola feature di differenza dal dataset.

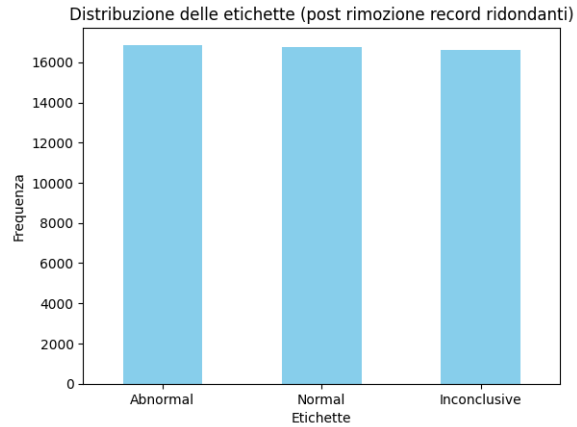


Figure 4: Differenze nella colonna Age

L'eliminazione delle righe che hanno in comune Name, Gender e Blood Type, ma che differiscono di una sola colonna, ha lasciato sostanzialmente invariata la distribuzione delle etichette della Y Test Results. Il dataset si mantiene quindi bilanciato.

3.4 Rimozione della colonna Room Number

La feature Room Number viene rimossa in quanto inutile ai fini dell'analisi e dell'addestramento di un modello di ML.

4 Statistica descrittiva

L'analisi statistica univariata rappresenta un passo fondamentale nella comprensione dei valori presenti nel dataset. In questa sezione vengono esaminate, da un punto di vista statistico, le varie feature presenti all'interno del dataset.

4.1 Statistiche delle variabili numeriche

	Age	Billing Amount
count	50213.000000	50213.000000
mean	51.575449	25554.091041
std	19.582698	14216.133950
min	13.000000	-2008.492140
25%	35.000000	13237.449118
50%	52.000000	25541.978243
75%	68.000000	37858.031471
max	86.000000	52764.276736

Figure 5: Describe su Age e Billing Amount

Age ha una media di circa 51 anni con un range 13-88 che è una distribuzione pressoché verosimile. Billing Amount ha valori negativi: questo è probabilmente dovuto a qualche errore di inserimento nei dati.

4.2 Statistiche delle variabili categoriche

	Name	Gender	Blood Type	Medical Condition	Date of Admission	Doctor	Hospital	Insurance Provider	Admission Type	Discharge Date	Medication	Test Results
count	50213	50213	50213	50213	50213	50213	50213	50213	50213	50213	50213	50213
unique	40235	2	8	6	1827	40341	39876	5	3	1856	5	3
top	Michael Smith	Female	AB+	Arthritis	2024-03-16	Michael Smith	LLC Smith	Cigna	Elective	2021-12-13	Lipitor	Abnormal
freq	23	25121	6313	8471	46	24	40	10133	16905	45	10077	16865

Figure 6: Describe su Age e Billing Amount

Molte variabili, tra cui Gender e Blood Type, hanno poche categorie. Verrà quindi utilizzato l'encoding per convertirle in numeri. Verranno modificate Gender, Blood Type, Medical Condition, Medication, Admission Type e Test Results. Inoltre sono state riscontrate 96 righe che presentano un Billing Amount negativo e che saranno per questo rimosse.

4.3 Boxplot Billing Amount ed Age

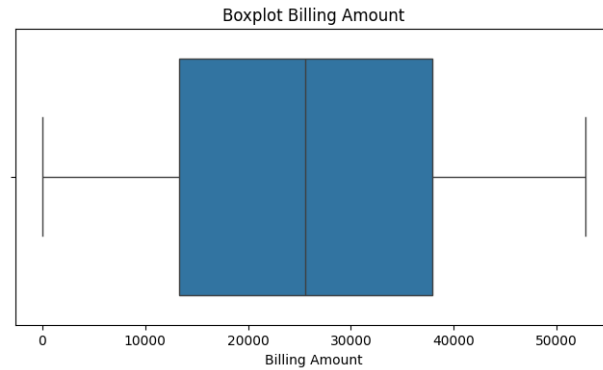


Figure 7: Boxplot Billing Amount

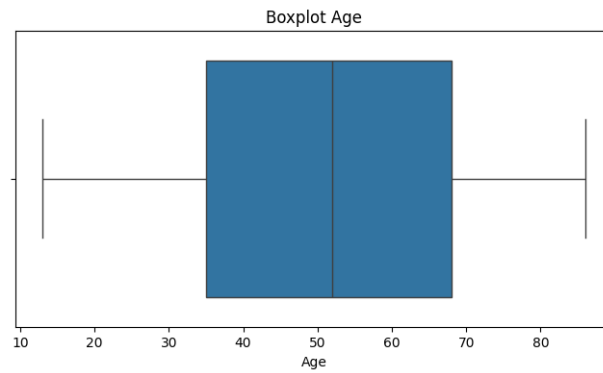


Figure 8: Boxplot Age

In entrambi i casi la mediana si trova approssimativamente al centro del box, indicando una distribuzione dei dati piuttosto simmetrica. I valori di "Billing Amount" sono distribuiti in modo abbastanza uniforme tra il minimo e il massimo, senza la presenza di outlier evidenti. Il boxplot di age risulta leggermente spostato verso destra. Questo potrebbe indicare una prevalenza di anziani.

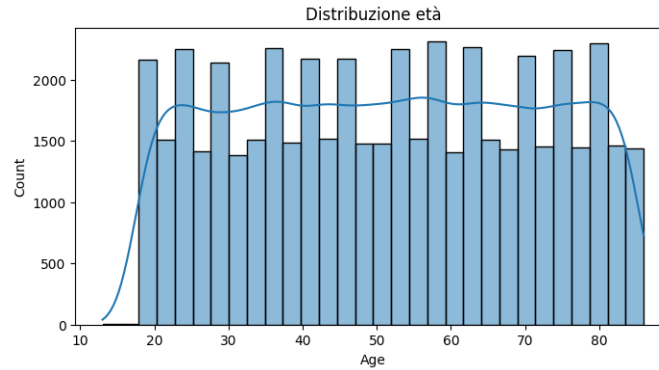


Figure 9: Distribuzione Age

L'età dei pazienti è distribuita in modo quasi uniforme tra 13 e 88 anni, con picchi piuttosto diffusi. L'andamento altalenante è dovuto anche al fatto che questa variabile è discreta e non continua.

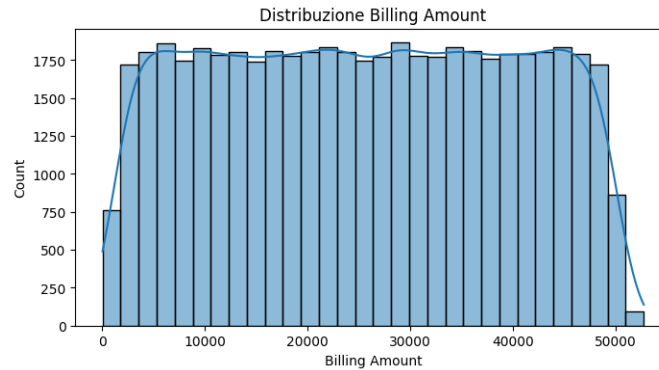


Figure 10: Distribuzione Billing Amount

La distribuzione di Billing Amount è abbastanza uniforme tra 0 e 50.000, senza evidenti picchi o concentrazioni. Questo indica che non ci sono outlier marcati e i valori sono ben distribuiti sull'intervallo.

4.4 Encoding

Le variabili Age, Gender, Blood Type, Medical Condition, Admission Type e Medication sono importanti per la predizione di Test Results, in quanto rappresentano caratteristiche cliniche e anagrafiche rilevanti e facilmente interpretabili.

Per poterle utilizzare nei modelli di machine learning, le variabili categoriche devono essere trasformate in numeri tramite encoding. Viene utilizzato il One Hot Encoding sulle feature categoriche scelte, perché hanno tutte poche categorie e non presentano un ordine naturale: questo metodo evita di introdurre relazioni spurie tra le categorie.

Per la variabile target (Test Results) viene usato, invece, il Label Encoding, così da fornire un'etichetta numerica univoca a ciascuna classe da predire.

4.4.1 Categorizzazione della colonna età

L'età è divisa ora nelle seguenti fasce:

- Da 13 a 24: Giovani
- Da 25 a 38: Adulti
- Da 39 a 59: Mezza età
- Da 60 a 86: Anziani

4.4.2 One Hot Encoding

Consiste nel dividere la colonna di interesse in più colonne binarie, ognuna delle quali rappresentativa di un valore specifico della colonna originaria. Verrà effettuato sulle feature: Age, Gender, Blood Type, Medical Condition, Admission Type e Medication.

Age_Giovani	Age_Adulti	Age_Mezza_Età	Age_Anziani	Gender
0	1	0	0	Male
0	0	0	1	Female
0	0	0	1	Male
0	1	0	0	Female
0	0	1	0	Male
...
0	0	0	1	Female
0	0	0	1	Male
0	1	0	0	Female
0	0	1	0	Male
0	0	1	0	Female

Figure 11: Age dopo il One Hot Encoding

4.4.3 Label Encoding

Verrà effettuato sulla variabile target Test Results. Consiste nel mappare i valori categorici della feature in un range numerico, in questo caso mapperemo Inconclusive a 0, Normal a 1 e Abnormal a 2.

4.4.4 Feature Selection: ANOVA F-test

Il test ANOVA (Analysis of Variance) è un metodo statistico che serve a confrontare le medie di due o più gruppi. Nel contesto della feature selection, si usa per capire se c'è una relazione significativa tra una variabile numerica (la feature) e una variabile categorica (il target).

Nel nostro caso viene utilizzato il test di ANOVA per capire quali colonne hanno la maggiore forza predittiva nei confronti dell'attributo target Test Results.

Feature	F-Value	p-Value
Medical Condition_Asthma	3.614737	0.026931
Medical Condition_Arthritis	2.398591	0.090856
Age_Adulti	2.045781	0.129290
Medical Condition_Hypertension	1.529547	0.216644
Age_Giovani	1.406111	0.245104
Blood Type_B+	1.391399	0.248737
Age_Mezza_Età	1.259454	0.283818
Blood Type_AB-	1.152926	0.315720
Medication_Lipitor	1.091579	0.335694
Age_Anziani	0.714408	0.489487
Blood Type_A+	0.695544	0.498808
Gender_Female	0.580098	0.559847
Gender_Male	0.580098	0.559847
Medication_Aspirin	0.577082	0.561538

Figure 12: Parte del risultato di ANOVA

L'F-value è un valore che varia nel range $[0; \infty)$. Un valore prossimo allo 0, o comunque molto basso, indica che la variabile in esame non ha forza predittiva nei confronti della Y. Il p-value invece è un valore che varia nell'intervallo $[0;1]$ e rappresenta la probabilità che l'associazione sia dovuta al caso. Nel caso specifico i valori risultanti sono bassissimi per l'F value.

Il p-value è statisticamente significativo solo quando è inferiore a 0.05. In questo caso risulta un solo valore che supera tale controlla, e questo è indice del fatto che la maggior parte dei dati a disposizione non saranno discriminanti nella predizione della Y. Nel caso specifico, il fatto che ci sia una sola colonna statisticamente significativa, mentre le altre non lo sono, è molto probabilmente un'eccezione casuale alla situazione generale del dataset.

4.5 Statistiche sulle variabili dopo l'encoding

	Age_Giovani	Age_Adulti	Age_Mezza_Età	Age_Anziani	Gender_Female	Gender_Male	Blood Type_A+	Blood Type_A-
count	50117.000000	50117.000000	50117.000000	50117.000000	50117.000000	50117.000000	50117.000000	50117.000000
mean	0.103119	0.204601	0.310693	0.381587	0.500170	0.499830	0.124888	0.125367
std	0.304117	0.403414	0.462782	0.485781	0.500005	0.500005	0.330595	0.331137
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000
75%	0.000000	0.000000	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Figure 13: Confronto sulla Deviazione Standard

Osservando la deviazione standard, si vede come la maggior parte delle persone rientra nelle categorie "Mezza Età" e "Anziani". Per quanto riguarda il resto delle colonne, invece, la situazione è perfettamente bilanciata.

4.5.1 Grafici a barre

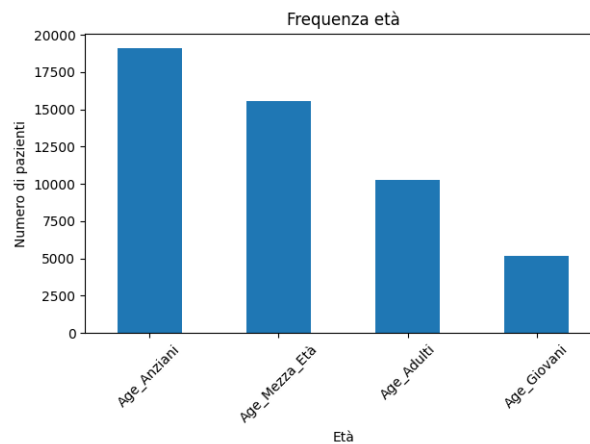


Figure 14: Barplot fasce di età

Il grafico è ancora più esplicativo della situazione, rappresentando quella che è la natura di molti dati sanitari dove la maggioranza dei pazienti ha un'età più avanzata.

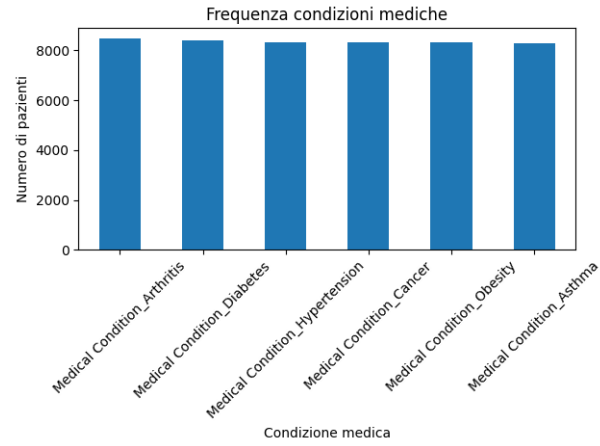


Figure 15: Barplot Clinical Condition

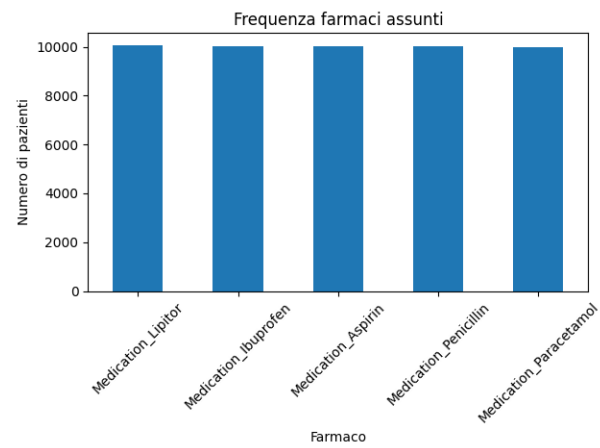


Figure 16: Barplot Farmaci Assunti

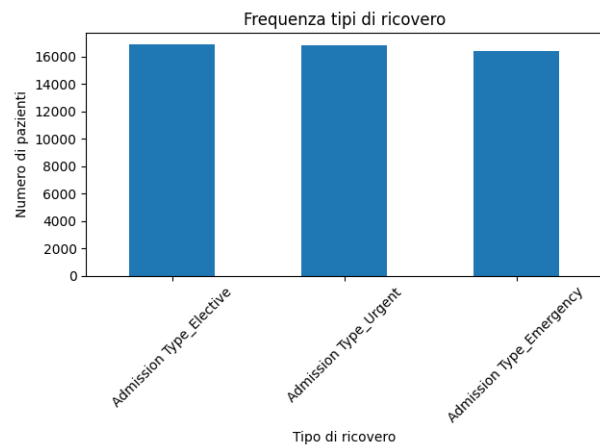


Figure 17: Barplot Tipi di Ricovero

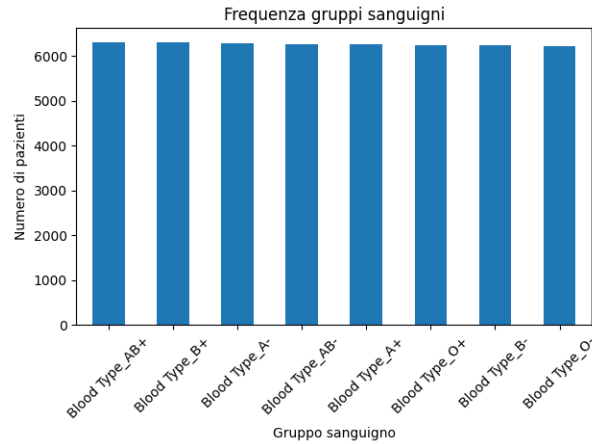


Figure 18: Barplot Gruppi Sanguigni

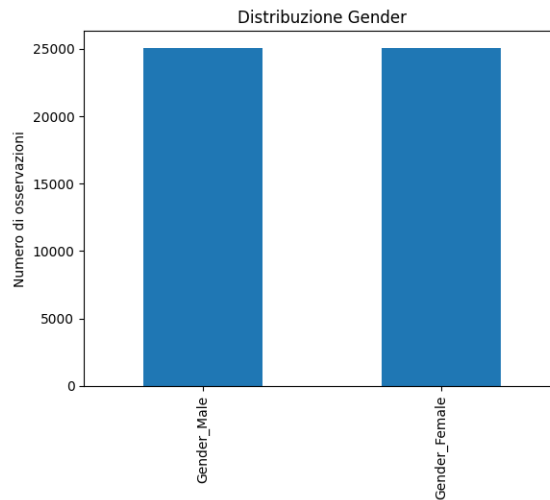


Figure 19: Barplot Gender

Dall'analisi statistica delle variabili numeriche dopo l'encoding, emerge che il dataset è ben bilanciato e privo di valori mancanti. In particolare, la distribuzione per genere è pressoché equa, come si evince dalle medie delle colonne Gender.Female e Gender.Male, entrambe prossime a 0.5. Lo stesso bilanciamento si riscontra tra le diverse tipologie di ricovero (Admission Type.Elective, Emergency, Urgent), ognuna delle quali rappresenta circa un terzo delle osservazioni.

Le variabili relative ai gruppi sanguigni e alle condizioni mediche/farmaci sono state codificate in formato binario. Le medie di queste colonne esprimono la proporzione della presenza di ciascuna caratteristica all'interno del dataset. Ad esempio, circa il 16,5% degli individui è affetto da obesità, mentre ogni gruppo sanguigno e ogni farmaco presentano percentuali simili tra loro, senza evidenziare forti squilibri.

Complessivamente, il dataset si presenta bilanciato in ogni variabile. Come evidenziato dal test di ANOVA, questa caratteristica però non è necessariamente un aspetto positivo.

4.6 Statistica descrittiva multivariata

In questa sezione analizzeremo la matrice di correlazione per valutare le relazioni tra le variabili in esame. Il nostro obiettivo è individuare eventuali associazioni tra le variabili, verificando se esistono legami significativi che possano influenzare l'interpretazione dei dati. Questo ci permetterà di comprendere meglio la struttura del dataset e di trarre spunti utili per le analisi successive.

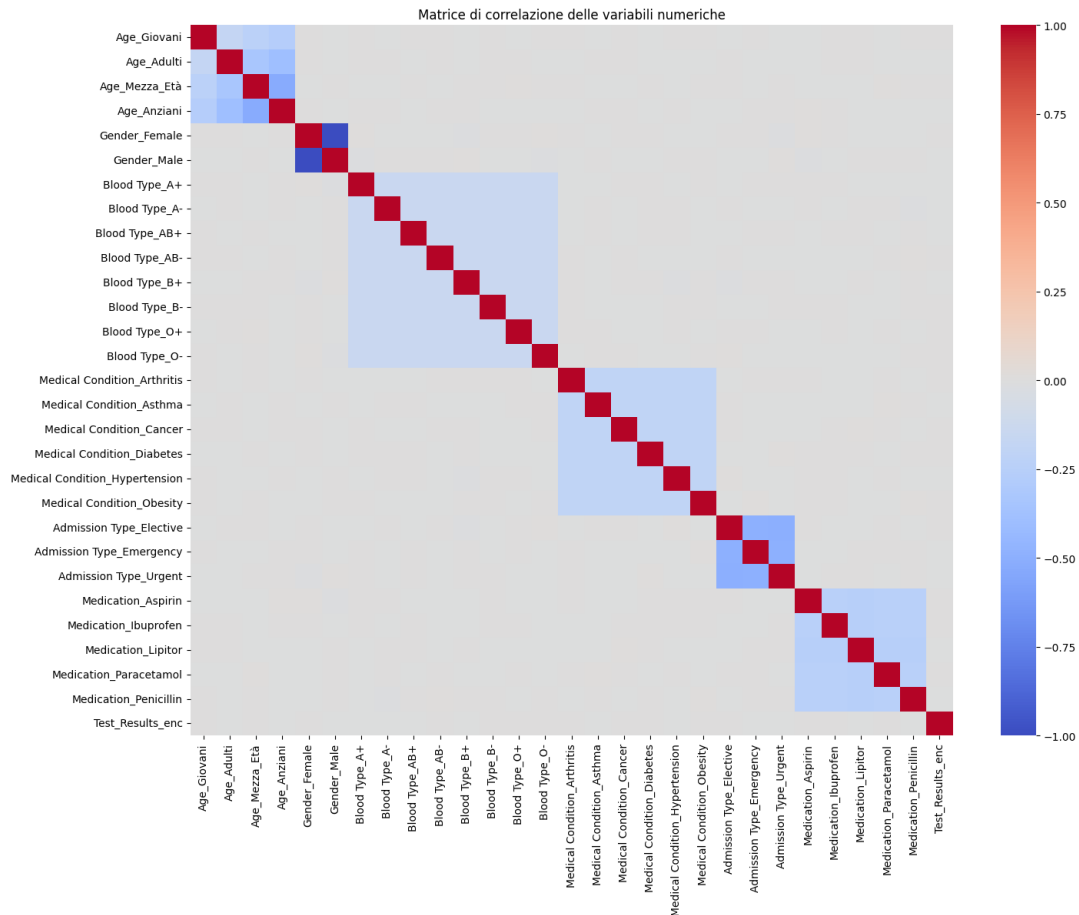


Figure 20: Heatmap

4.6.1 Risposta alla RQ1

Q RQ₁. Quali fattori influenzano maggiormente l'esito dei test clinici?

La heatmap mostra che la maggior parte delle variabili numeriche non presenta forti correlazioni tra loro: la matrice è quasi tutta grigia, cioè vicina a zero.

Le uniche correlazioni evidenti (in blu) sono tra variabili mutuamente esclusive (ad esempio tra diverse fasce di età o tra i diversi gruppi sanguigni), che mostrano una correlazione negativa, come è atteso dopo l'encoding.

Non si osservano correlazioni rilevanti tra le feature e la variabile target ("Test_Results_enc"). In sintesi: non ci sono relazioni forti o particolari pattern tra le variabili del dataset.

Test_Results_enc	
Test_Results_enc	1.000000
Medical Condition_Arthritis	0.004781
Admission Type_Elective	0.003921
Medical Condition_Obesity	0.003816
Blood Type_AB-	0.003738
Blood Type_O+	0.003530
Medication_Aspirin	0.003253
Age_Giovani	0.001872
Blood Type_O-	0.001849
Medical Condition_Diabetes	0.001741
Medication_Paracetamol	0.001610
Blood Type_B-	0.001163
Medication_Ibuprofen	0.000929
Age_Mezza_Età	0.000715
Medical Condition_Cancer	0.000525

Figure 21: Focus sui valori dell' Heatmap

La correlazione nei confronti della variabile target è sostanzialmente nulla. Questa valutazione conferma quanto detto in precedenza con il test di ANOVA.

5 Machine Learning

In questa fase, usiamo algoritmi come la Random Forest e la Logistic Regression che imparano dai dati esistenti a riconoscere i pattern e a fare una previsione su "Test Results". L'obiettivo è costruire un modello che, una volta addestrato, sia in grado di indovinare correttamente i risultati anche su dati che non ha mai visto.

5.0.1 Creazione del test e validation set

Viene utilizzato il rapporto 80% (grandezza del Train set) a 20% (grandezza del Test Set) rispetto alla grandezza totale del dataset.

Il Train set è quello che il modello usa per imparare a fare previsioni, mentre il Test set è una parte di dati che il modello non ha mai visto e che usiamo per capire se le sue previsioni sono affidabili. Questo è fondamentale per garantire che il modello sia in grado di generalizzare e fare previsioni corrette anche su nuovi dati, e non solo su quelli che ha già studiato.

5.1 Random Forest

La Random Forest è un algoritmo di machine learning supervisionato che combina molti alberi decisionali, ciascuno addestrato su campioni diversi del dataset e su sottoinsiemi casuali delle feature. In questo modo, la Random Forest riesce a migliorare la precisione rispetto a un singolo albero e a ridurre il rischio di overfitting. Nel nostro caso, stiamo usando la Random Forest per prevedere il risultato del test ("Test_Result_enc") a partire da molteplici caratteristiche (feature) dei dati. Grazie alla sua robustezza e flessibilità, la Random Forest è particolarmente adatta a problemi come il nostro, dove ci sono molte feature, alcune delle quali categoriche e codificate, e diverse classi di output.

	precision	recall	f1-score	support
0	0.33	0.33	0.33	3314
1	0.33	0.32	0.33	3344
2	0.33	0.34	0.33	3366
accuracy			0.33	10024
macro avg	0.33	0.33	0.33	10024
weighted avg	0.33	0.33	0.33	10024

Figure 22: Report Random Forest

Il modello non riesce a distinguere tra le classi target meglio di una scelta casuale su 3 valori. Tutte le metriche sono molto basse e simili tra le classi, il che indica:

- nessuna combinazione delle feature disponibili permette di prevedere la variabile target in modo efficace.
- il modello non sta imparando pattern utili dai dati.

Questo dataset, essendo completamente bilanciato e privo di qualsiasi correlazione tra le feature e la variabile target, rappresenta un ottimo esempio didattico. Mostra come, in assenza di pattern o relazioni reali nei dati, anche i migliori algoritmi di machine learning non sono in grado di apprendere o prevedere efficacemente la variabile di interesse.

In altre parole, la qualità e la struttura dei dati sono fondamentali: senza informazioni davvero utili nei dati, nessun modello potrà mai raggiungere buone performance predittive.

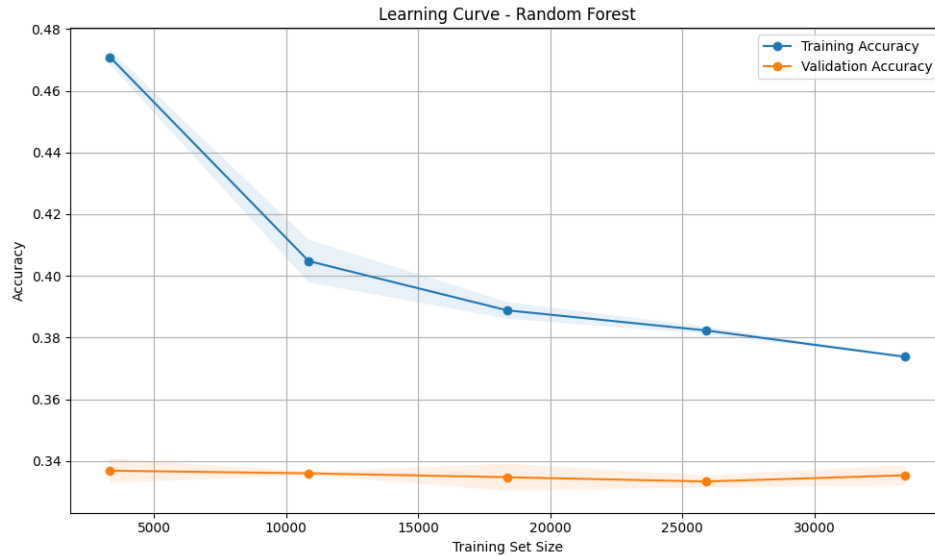


Figure 23: Curva di apprendimento del Random Forest

La curva mostra l'andamento dell'accuratezza su training set (blu) e validation set (arancione) al variare della dimensione del training set.

Il grafico dimostra visivamente che, anche aumentando la quantità di dati, il modello non migliora se non ci sono informazioni utili nelle feature.

La learning curve mostra chiaramente che la validazione rimane piatta vicino al valore casuale, confermando che nessuna delle feature disponibili contiene informazioni utili per prevedere la variabile target. Questo esempio sottolinea l'importanza della qualità e della struttura dei dati nel machine learning: senza pattern reali nei dati, nessun algoritmo può apprendere o generalizzare.

5.2 Regressione Logistica

La regressione logistica è un modello di classificazione supervisionato utilizzato per prevedere l'appartenenza di un'osservazione a una o più classi. Nel caso multiclass, utilizza la funzione softmax per distinguere tra più di due categorie. Nel nostro caso, abbiamo utilizzato la regressione logistica per predire la variabile `Test_Result_enc`, che rappresenta tre possibili esiti: Inconclusive (0), Normal (1) e Abnormal (2), sulla base delle altre feature del dataset.

	precision	recall	f1-score	support
0	0.33	0.20	0.25	3314
1	0.33	0.36	0.34	3344
2	0.33	0.44	0.38	3366
accuracy			0.33	10024
macro avg	0.33	0.33	0.32	10024
weighted avg	0.33	0.33	0.32	10024

Figure 24: Report Regressione Logistica

Dai risultati si osserva che la regressione logistica non sta riuscendo a distinguere tra le tre classi:

- la precision, recall e f1-score per tutte le classi sono circa 0.33, ovvero il valore che ci si aspetterebbe da una classificazione casuale tra tre classi bilanciate.
- l'accuracy complessiva è 0.33, molto bassa e in linea con una scelta casuale.

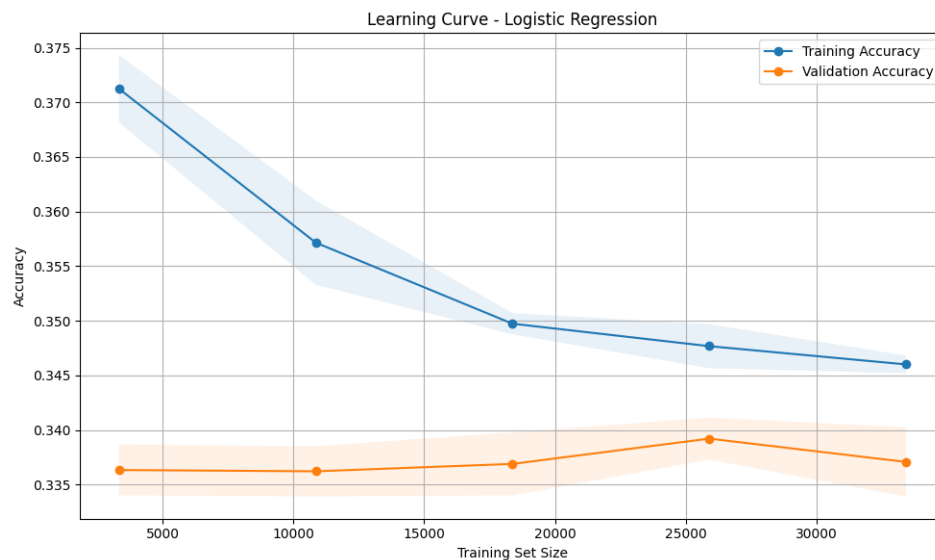


Figure 25: Curva di apprendimento della Regressione Logistica

L'accuratezza sul training set parte più alta ma diminuisce all'aumentare dei dati, stabilizzandosi attorno a 0.345.

L'accuratezza sul validation set resta stabile e bassa indipendentemente dalla dimensione del training set.

Il gap tra training e validation accuracy è molto ridotto, indice che il modello non sta overfittando.

La regressione logistica non riesce a catturare relazioni significative nei dati: sia l'accuratezza in training che in validation sono basse e vicine tra loro, il che è segno di underfitting.

5.3 Support Vector Machine (SVM)

L'algoritmo SVM è un modello di classificazione supervisionato che cerca di trovare le “frontiere” (iperpiani) che separano al meglio le diverse classi nei dati. Nel nostro caso, utilizziamo SVM per cercare di distinguere tra le tre possibili classi del test (“Test_Result_enc”), sfruttando tutte le feature disponibili. SVM è particolarmente efficace quando le classi sono separabili da confini netti, ma può diventare meno efficiente se i dati non presentano pattern chiari.

	precision	recall	f1-score	support
0	0.34	0.20	0.25	3314
1	0.33	0.36	0.34	3344
2	0.33	0.44	0.38	3366
accuracy			0.33	10024
macro avg	0.33	0.33	0.32	10024
weighted avg	0.33	0.33	0.32	10024

Figure 26: Report Support Vector Machine

Anche in questo caso, il classificatore SVM mostra prestazioni molto basse:

- accuracy: 0.33, ovvero circa il valore che ci si aspetta da una classificazione casuale su tre classi.
- precision, recall e f1-score sono molto simili per tutte le classi e vicini al valore casuale (~0.33).
- le metriche macro e weighted average confermano che il modello non sta imparando alcuna relazione significativa tra le feature e la variabile da predire.

Come per la regressione logistica, anche con SVM il modello non riesce a distinguere tra le tre classi. Questo suggerisce che le feature disponibili non contengono informazioni utili per la classificazione.

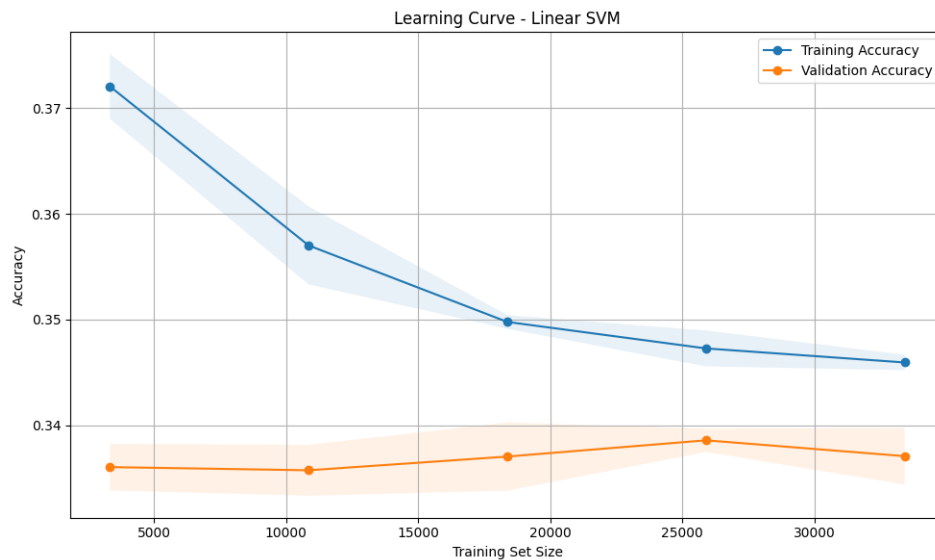


Figure 27: Curva di apprendimento dell'SVM

La curva mostra che la SVM non riesce ad apprendere pattern utili: l'accuratezza su training e validation resta bassa e vicina al caso (circa 0.33), con un gap minimo tra le due curve. È un chiaro esempio di underfitting dovuto all'assenza di informazioni predittive nei dati.

5.4 Naive Bayes

Il Naive Bayes è un classificatore probabilistico che si basa sul teorema di Bayes, assumendo che le feature siano indipendenti tra loro (ipotesi “naive”). Nel nostro scenario, applichiamo Naive Bayes per stimare la probabilità che ciascun esempio appartenga a una delle tre classi (“Test_Result_enc”), usando le informazioni contenute nelle feature. È un modello semplice e veloce, particolarmente adatto quando le feature sono realmente indipendenti e il dataset è ampio.

	precision	recall	f1-score	support
0	0.33	0.30	0.32	3314
1	0.33	0.34	0.33	3344
2	0.33	0.35	0.34	3366
accuracy			0.33	10024
macro avg	0.33	0.33	0.33	10024
weighted avg	0.33	0.33	0.33	10024

Figure 28: Report Naive Bayes

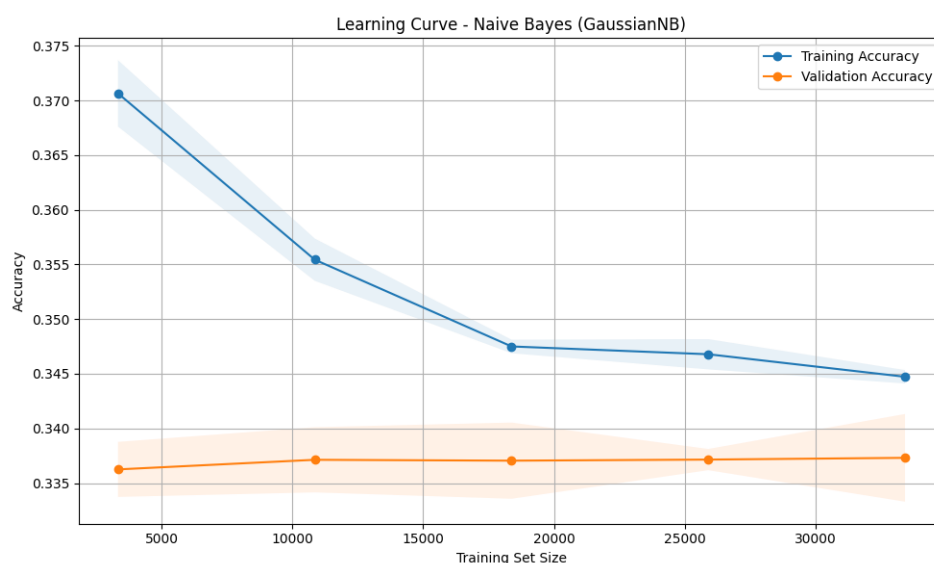


Figure 29: Curva di apprendimento di Naive Bayes

La curva conferma che anche il Naive Bayes non apprende pattern utili: la validation accuracy resta costante e vicino al caso (~ 0.33), segno di underfitting e dati non informativi.

I risultati ottenuti — con accuracy e metriche f1-score sempre intorno al 33% per tutti i modelli — dimostrano che, in assenza di vere relazioni tra feature e target, nessun algoritmo riesce a performare meglio del caso.

Questo scenario sottolinea che anche i modelli più avanzati sono impotenti di fronte a dati privi di pattern: il processo di apprendimento fallisce perché non ci sono regole o segnali statistici da cogliere. Ne deriva che, nella pratica, la fase di analisi e preparazione dei dati (feature engineering, selezione di variabili significative, raccolta di dati realmente informativi) è spesso decisiva per il successo di qualsiasi progetto di machine learning, molto più della scelta del modello stesso.

5.5 Grid Search

La **Grid Search** è una tecnica di ottimizzazione degli iperparametri usata nel machine learning.

In breve:

- Si definisce una **griglia di possibili valori** per gli iperparametri di un modello (es. numero di alberi, profondità massima, valore di C in SVM).
- L'algoritmo prova **tutte le combinazioni possibili** di quei valori.
- Per ciascuna combinazione, il modello viene **addestrato e valutato** (tipicamente con cross-validation).
- Alla fine, viene scelta la configurazione di iperparametri che produce le **migliori prestazioni**.

È molto accurata ma può essere costosa computazionalmente, soprattutto se la griglia è grande.

Nel nostro caso verranno messi a confronto i risultati prodotti dai modelli di ML precedentemente utilizzati con quelli prodotti dalla Grid Search per vedere se questa tecnica produrrà dei miglioramenti.

Tenuto conto del costo computazionale che una ricerca esaustiva può avere se applicata alla Random Forest, verrà usata la Randomized Search, che esplora solo un sottoinsieme casuale delle combinazioni di iperparametri riducendo il costo computazionale.

Per gli altri modelli (Regressione Logistica, SVM e Naive Bayes) è stata invece applicata la Grid Search, che testa in modo esaustivo tutte le combinazioni definite nella griglia, garantendo una ricerca completa ma più dispendiosa in termini di tempo di calcolo.

4. CONFRONTO RISULTATI: BASELINE vs OTTIMIZZATI				
	Baseline	Ottimizzato	Miglioramento	Miglioramento %
Random Forest	0.3305	0.3333	0.0028	0.8452
Logistic Regression	0.3313	0.3313	0.0000	0.0000
LinearSVC	0.3315	0.3315	0.0000	0.0000
Naive Bayes	0.3295	0.3295	0.0000	0.0000

Figure 30: Report Grid Search

Il report mostra un miglioramento inferiore all'1% per quanto concerne la Random Forest, mentre dello 0% per tutti gli altri modelli. In sostanza, l'utilizzo della Grid Search non ha apportato miglioramenti significativi ai modelli.

5.6 Risposta alla RQ2

Q RQ₂. *Come si comportano i principali modelli di Machine Learning di fronte al dataset scelto?*

L'applicazione dei principali modelli di Machine Learning (Random Forest, Regressione Logistica, Support Vector Machine e Naive Bayes) ha mostrato risultati equivalenti a una classificazione casuale. Tutti i modelli hanno raggiunto un'accuracy intorno al 33%, con precision, recall e f1-score distribuiti uniformemente tra le tre classi, senza alcun segnale di apprendimento significativo.

Questi risultati indicano che:

- Non esistono relazioni tra le feature disponibili e la variabile target.
- Anche algoritmi complessi falliscono quando i dati sono bilanciati ma privi di pattern informativi.
- Il limite non risiede nel modello, ma nel dataset: senza segnali statistici rilevanti, nessun algoritmo riesce a generalizzare o migliorare rispetto al caso.

In conclusione, questo scenario evidenzia come il successo di un progetto di ML dipenda più dalla qualità e significatività dei dati (feature engineering, selezione e raccolta di variabili utili) che dalla scelta del modello stesso.

6 Natural Language Processing

In questa fase applichiamo tecniche di anonimizzazione e pseudoanonimizzazione alle note testuali: rileviamo entità sensibili (nomi di persone, date o luoghi) mediante NER (Named Entity Recognition) e pattern Regex (Regular Expressions), quindi le sostituiamo con tag standard (es. [PERSON]). Questo produce un testo riutilizzabile in modo sicuro e facilita analisi successive; effettuiamo anche controlli campionari per verificare la qualità del riconoscimento.

6.1 Creazione delle note cliniche

Per arricchire il dataset è stata generata per ogni riga una **nota clinica sintetica** in linguaggio naturale. A tal fine sono stati definiti tre **template testuali** con struttura e lessico diversi, all'interno dei quali sono stati inseriti i valori specifici delle feature (es. *Name, Age, Gender, Hospital, Medical Condition*, ecc.).

Questa tecnica consente di creare descrizioni cliniche realistiche e variate, simulando la documentazione che normalmente si trova nelle cartelle sanitarie elettroniche. L'approccio a template garantisce sia **coerenza strutturale** tra le note, sia una sufficiente **variabilità linguistica**, utile per testare successivi metodi di elaborazione del linguaggio naturale. In questo modo, ciascun record numerico e tabellare del dataset viene accompagnato da una rappresentazione testuale in forma discorsiva, che potrà essere utilizzata per l'anonimizzazione tramite regex o NER.

Di seguito ci sono le prime tre note cliniche generate come detto precedentemente:

— **RIGA 1** — On 2024-01-31, Bobby Jackson (Male, 30 years old, blood type B-) was admitted to Sons and Miller for treatment of Cancer (Urgent admission). Covered by Blue Cross, the patient was treated by Dr. Matthew Smith who prescribed Paracetamol. Tests revealed Normal. Discharge occurred on 2024-02-02 with a billing charge of \$18856.28.

— **RIGA 2** — Leslie Terry, (Male, 62 years old, blood type A+) was admitted in Emergency mode. The patient was hospitalized on 2019-08-20 to Kim Inc for Obesity. Patient insurance is Medicare. The Dr. Samantha Davies prescribed the Ibuprofen. Tests have shown Inconclusive results. The patient was discharged on 2019-08-26 with a billing amount in this range: \$33643.33.

— **RIGA 3** — On 2022-09-22, Danny Smith (Female, 76 years old, blood type A-) was admitted to Cook PLC for treatment of Obesity (Emergency admission). Covered by Aetna, the patient was treated by Dr. Tiffany Mitchell who prescribed Aspirin. Tests revealed Normal. Discharge occurred on 2022-10-07 with a billing charge of \$27955.10.

6.1.1 Generalizzazione di Age e Billing Amount

Prima di procedere con l'anonimizzazione, generalizzeremo i dati.

Convertiremo l'età in fasce di età e i valori monetari ("Billing Amount") in range specifici.

Questo ci aiuterà a ridurre la granularità dei dati, a renderli meno identificabili e a semplificare l'analisi successiva.

Dopo la generalizzazione avremo in ordine come fasce d'età: young, adult, middle age, elderly. E come range per il billing amount avremo: 0-10.000\$, 10.001-20.000\$, 20.001-30.000\$, 30.001-40.000\$, 40.001-50.000\$, >50.000\$

6.2 Anonimizzazione dei dati sensibili

Secondo il **Regolamento Generale sulla Protezione dei Dati** (GDPR), sono considerati dati personali tutte le informazioni che identificano o rendono identificabile una persona fisica, direttamente o indirettamente. Inoltre, i dati relativi alla salute rientrano nelle categorie particolari di dati personali e richiedono protezioni rafforzate.

Il **GDPR** non fornisce un elenco rigido di campi da anonimizzare, ma richiede che l'anonimizzazione sia tale da rendere impossibile (o estremamente improbabile) la re-identificazione dell'interessato.

Nel dataset analizzato (circa 50.000 record, con distribuzione bilanciata delle varie feature), abbiamo deciso di applicare il seguente criterio:

- Anonimizzare gli identificatori diretti, che collegano immediatamente un record a una persona fisica.
- Generalizzare gli identificatori indiretti ad alto rischio, che potrebbero permettere re-identificazione se combinati ad altri dati.
- Mantenere i dati sanitari e clinici, poiché, una volta scollegati dagli identificatori, non consentono la re-identificazione nel contesto del dataset.

6.2.1 Scelte applicate

In questo contesto, alcune feature vengono anonimizzate mentre altre vengono generalizzate.

Feature anonimizzate:

- Name → identificatore diretto (nome e cognome del paziente).
- Doctor → identificatore diretto (dato personale di un professionista sanitario).
- Hospital → potenziale identificatore (soprattutto in contesti locali con poche strutture).
- Insurance Provider → identificatore indiretto, legato univocamente all'assicurato.
- Date of Admission e Discharge Date → per ridurre il rischio di identificazione tramite cronologia.

Feature generalizzate:

- Age → in fasce d'età young, adult, middle age, elderly.
- Billing Amount → in range monetari: 0-10.000\$, 10.001-20.000\$, 20.001-30.000\$, 30.001-40.000\$, 40.001-50.000\$, >50.000\$.

Campi non modificati:

I restanti campi (Gender, Blood Type, Medical Condition, Admission Type, Medication, Test Results) sono mantenuti perché necessari per analisi clinico-statistiche e, dopo la rimozione degli identificatori diretti e la riduzione di granularità di importi e età, presentano un rischio residuale ritenuto accettabile nel contesto (dataset ampio e bilanciato).

6.3 Gold Dataset

Il **gold dataset** (o *gold standard*) rappresenta un insieme di dati di riferimento di alta qualità, in cui le entità sensibili sono state **annotate manualmente**. Questo dataset non viene utilizzato per l'addestramento dei modelli, ma funge da **benchmark oggettivo** per valutarne le prestazioni.

Nel nostro caso, abbiamo costruito un gold dataset annotando le note cliniche generate con le principali entità da riconoscere, come **nomi di pazienti e medici (PERSON)**, **date (DATE)**, **ospedali e assicurazioni (ORG)**. Ogni testo è accompagnato da una lista di etichette che identificano con precisione i segmenti sensibili.

Confrontando i risultati prodotti dalla RegEx e dai modelli NER con le annotazioni del gold dataset, possiamo calcolare metriche come **accuracy**, **precision**, **recall** e **F1-score**, valutando così quanto i modelli siano efficaci nell'individuare e anonimizzare le entità rilevanti. Questo approccio garantisce una valutazione trasparente e riproducibile delle tecniche di anonimizzazione.

6.4 Metriche utilizzate

Per valutare le prestazioni dei modelli sul nostro **gold dataset**, utilizziamo quattro metriche fondamentali della classificazione: **accuracy**, **precision**, **recall** e **F1-score**.

- **Accuracy**: misura la percentuale di entità correttamente identificate rispetto al totale delle entità considerate (vere e predette). Nel nostro contesto indica, in generale, quanto il modello “indovina” correttamente le entità PERSON, ORG e DATE.
- **Precision**: rappresenta la proporzione di entità che il modello ha riconosciuto e che erano effettivamente corrette. In altre parole, ci dice quanto il modello è “selettivo”, evitando falsi positivi (es. etichettare un ospedale come nome di persona).
- **Recall**: misura la percentuale di entità presenti nel gold dataset che il modello è riuscito effettivamente a trovare. In pratica, indica quanto il modello è “sensibile” nel riconoscere tutte le entità rilevanti, evitando falsi negativi (es. non riconoscere una data importante nel testo).

- **F1-score:** è la media armonica tra precision e recall, e fornisce una misura bilanciata tra le due. Nel nostro caso, è particolarmente utile perché evita che un modello con alta precision ma bassa recall, o viceversa, venga considerato buono solo su una delle due dimensioni.

Nel nostro progetto, queste metriche vengono calcolate confrontando le entità annotate nel **gold dataset** con quelle riconosciute dai modelli. Questo ci permette di avere una valutazione **quantitativa e trasparente** delle prestazioni della RegEx e dei diversi modelli (spaCy: *sm*, *md*, *lg*, Flair e Bert) e di capire quale sia il più efficace per l'anonimizzazione dei dati sensibili nelle note cliniche.

6.5 Anonimizzazione tramite RegEx

Le espressioni regolari (RegEx) sono sequenze di caratteri che definiscono un modello di ricerca all'interno di stringhe di testo. Vengono comunemente utilizzate per individuare, sostituire o manipolare porzioni di testo secondo regole ben precise.

Nel contesto di questo progetto, le regex sono state impiegate per riconoscere e anonimizzare automaticamente le colonne identificate come contenenti dati sensibili (es. Name, Doctor, Insurance Provider, ecc.), sostituendo i valori originali con etichette o codici anonimi.

Non è stata invece applicata la regex alla colonna Hospital: trattandosi di un attributo con un elevato numero di valori univoci e una variabilità simile a quella dei nomi propri, l'utilizzo di pattern regex avrebbe potuto generare ambiguità e falsi positivi (es. confondere un nome di ospedale con un nome di persona).

Di seguito c'è un esempio di RegEx per l'identificazione dell'attributo **Doctor**:

- 'Dr'+[A-Z][a-z]++[A-Z][a-z]+' , 'Doctor [DOCTOR]'
- 'The Dr'+[A-Z][a-z]++[A-Z][a-z]+' , 'The Dr. [DOCTOR]'

```
REPORT ANONIMIZZAZIONE TRAMITE REGEX
Accuracy: 0.8000
Precision: 1.0000
Recall: 0.8000
F1-score: 0.8889
```

Figure 31: Report Regex

I valori ottenuti mostrano ottima precisione (1.0), cioè ogni volta che la regex interviene lo fa correttamente senza introdurre falsi positivi. La recall (0.8) e di conseguenza l'F1 (0.89) risultano invece più basse, perché una parte delle entità non viene anonimizzata.

L'accuracy riflette il fatto che il 20% delle entità gold non è stato anonimizzato. Questo valore è coerente con la recall (0.80): il sistema riesce ad anonimizzare correttamente 4 entità su 5, mentre la restante parte sfugge al pattern regex.

Questo è dovuto principalmente alle organizzazioni ospedaliere, su cui la regex non è stata applicata: il dataset contiene infatti circa 39.000 valori univoci su 50.000, quindi sarebbe impraticabile gestirli con semplici pattern regolari.

L'approccio regex risulta quindi molto affidabile, ma la copertura resta limitata in particolare sugli ospedali, dove servirebbe un modello più flessibile (NER o dizionario dinamico).

6.5.1 Risposta alla RQ3

Q RQ₃. *Quanto sono efficaci le regular expression nell'anonimizzare automaticamente le note cliniche generate?*

La principale criticità riguarda le organizzazioni ospedaliere: il dataset contiene infatti circa 39.000 valori univoci su 50.000, rendendo impraticabile coprirli con semplici pattern regolari. Le regex funzionano bene per categorie standardizzabili (date, età, importi, alcuni nomi propri), ma non riescono a scalare su domini estremamente variabili come gli ospedali o le cliniche.

In conclusione, l'approccio basato su regex è molto affidabile in termini di correttezza, ma mostra una copertura limitata. Per ottenere un'anonimizzazione completa sarebbe necessario integrare metodi più flessibili, come modelli NER, in grado di gestire la varietà lessicale del dominio medico.

6.6 Anonimizzazione Tramite NER

La **Named Entity Recognition (NER)** è una tecnica di *Natural Language Processing (NLP)* che permette di identificare e classificare automaticamente in un testo determinate entità, come nomi di persona, luoghi, organizzazioni, date o valori numerici. A differenza delle espressioni regolari, che si basano su regole rigide e pattern statici, la NER sfrutta modelli di **apprendimento automatico** in grado di riconoscere le entità anche in contesti diversi, con maggiore flessibilità e capacità di generalizzazione.

Nel nostro caso la NER è stata utilizzata con lo stesso obiettivo delle regex, ovvero **anonimizzare i dati sensibili**, ma con un approccio più potente.

Rispetto all'esperimento precedente, abbiamo incluso anche la colonna *Hospital* nell'anonimizzazione: la NER infatti dovrebbe essere in grado di distinguere correttamente i nomi delle strutture sanitarie da quelli delle persone, riducendo il rischio di confusione. Per implementare questa tecnica ci siamo basati su **modelli pre-addestrati** disponibili nelle principali librerie NLP, in particolare:

- **spaCy**, per l'estrazione rapida di entità con modelli leggeri e facilmente integrabili;
- **Flair**, che permette di combinare diversi embeddings e ottenere prestazioni elevate nella classificazione delle entità;
- **BERT**, che grazie al contesto bidirezionale offre una maggiore accuratezza nel riconoscimento delle entità complesse.

6.6.1 Utilizzo di Spacy

Per implementare la Named Entity Recognition abbiamo utilizzato **spaCy**, una delle librerie più diffuse e performanti per il NLP. SpaCy mette a disposizione modelli linguistici già **pre-addestrati** su grandi corpora di testo, che permettono di riconoscere

automaticamente entità come nomi di persona, organizzazioni, luoghi, date e valori numerici.

In particolare, nel nostro progetto abbiamo testato tre modelli della lingua inglese forniti da spaCy:

- **en_core_web_sm**: modello “small”, leggero e veloce, adatto a scenari in cui le risorse computazionali sono limitate. Ha prestazioni più basse ma tempi di esecuzione molto rapidi.
- **en_core_web_md**: modello “medium”, che include vettori semantici a 300 dimensioni. Offre un buon compromesso tra accuratezza e velocità, risultando più preciso rispetto alla versione *sm*.
- **en_core_web_lg**: modello “large”, con vettori semantici ad alta dimensionalità e una copertura linguistica più ampia. È il più pesante in termini di memoria, ma garantisce risultati migliori nell’identificazione delle entità.

L’uso combinato di questi modelli ci consente di confrontare **prestazioni e compromessi**: da un lato la rapidità di esecuzione (*sm*), dall’altro la maggiore accuratezza nell’anonimizzazione dei dati sensibili (*lg*). In questo modo possiamo scegliere consapevolmente il modello più adatto in base agli obiettivi di precisione e alle risorse disponibili.

I risultati ottenuti sui modelli Spacy sono i seguenti:

```
=====
📊 STATISTICHE COMPLETE:
♦ Modello: en_core_web_sm
📄 Campioni processati: 50
📊 Accuracy standard: 44.0%

Precision: 0.3827
Recall:    0.4404
F1-score:  0.4064
```

Figure 32: Report Spacy Small

```
=====
📊 STATISTICHE COMPLETE:
♦ Modello: en_core_web_md
📄 Campioni processati: 50
📊 Accuracy standard: 39.2%

Precision: 0.4580
Recall:    0.3915
F1-score:  0.4107
```

Figure 33: Report Spacy Medium

```
=====
STATISTICHE COMPLETE:
• Modello: en_core_web_lg
• Campioni processati: 50
• Accuracy standard: 51.9%

Precision: 0.4691
Recall:    0.5186
F1-score: 0.4903
```

Figure 34: Report Spacy Large

Dai test effettuati sui 50 campioni del gold dataset, possiamo osservare alcune differenze nelle prestazioni dei tre modelli di spaCy:

- **en_core_web_sm**: ha mostrato un'accuracy del 44% con valori di precision e recall relativamente bassi (0.38-0.44). Questo conferma che il modello “small”, pur essendo veloce, non riesce a catturare bene le entità in testi complessi come quelli sanitari. In diversi casi, ha confuso organizzazioni con persone (“Miller” → PERSON) o ha introdotto entità spurie (“Urgent” → ORG).
- **en_core_web_md**: sorprendentemente ha ottenuto risultati **peggiori dello small** in termini di accuracy (39,2%), pur con una precision leggermente più alta (0.46). Il modello medio sembra introdurre più falsi positivi (es. “Kim Inc” → PERSON invece che ORG, “Inconclusive” → PERSON), riducendo la qualità complessiva delle previsioni. Questo dimostra che avere vettori semantici più ricchi non garantisce automaticamente un miglioramento, soprattutto se il dominio del testo è molto diverso dal training set originale.
- **en_core_web_lg**: è risultato il modello migliore, con un'accuracy del 51,9% e un F1-score di circa 0.49. Pur non essendo perfetto, ha identificato correttamente più entità rispetto agli altri modelli e ha ridotto il numero di errori grossolani. Anche qui si notano alcune confusioni, come entità spurie (“Ibuprofen” → ORG, “Female” → ORG), ma nel complesso riesce a catturare meglio le informazioni chiave (PERSON, ORG, DATE).

In sostanza:

- Lo **small** è troppo limitato per un contesto delicato come quello sanitario.
- Il **medium** non porta vantaggi tangibili e introduce più rumore.
- Il **large** è il più affidabile, pur con margini di miglioramento, e rappresenta la scelta migliore tra i tre per valutare le tecniche di anonimizzazione.

6.6.2 Utilizzo di Bert

Oltre a spaCy, abbiamo valutato anche l'uso di modelli transformer-based, in particolare BERT.

E' stato usato il modello pre-addestrato dbmdz/bert-large-cased-finetuned-conll03-english, ottimizzato sul dataset CoNLL-2003, che include entità come PERSON, ORG, LOC, MISC. Rispetto a spaCy, BERT adotta un approccio più potente basato su deep learning contestuale: ogni parola viene rappresentata in funzione del contesto circostante, consentendo al modello di disambiguare meglio entità ambigue o distribuite su più token. Le categorie di BERT sono state mappate sulle stesse usate con spaCy, in modo da avere un confronto uniforme (PER → PERSON, ORG/MISC/LOC → ORG).

Poiché il modello non prevede direttamente entità come DATE, queste non sono coperte, e ci siamo concentrati principalmente su PERSON e ORG.

```
=====
STATISTICHE COMPLETE:
+ Modello: bert-large-cased-finetuned-conll03-english
+ Campioni processati: 50
+ Accuracy standard: 24.5%

Precision: 0.1691
Recall:    0.2454
F1-score:  0.1956
```

Figure 35: Report Bert

I risultati ottenuti con **BERT (bert-large-cased-finetuned-conll03-english)** mostrano un'accuracy del **24,5%** e valori relativamente bassi di **precision (0,17)**, **recall (0,24)** e **F1-score (0,19)**.

Questi numeri sono indice di una **copertura limitata del modello nel contesto medico-sanitario**. Infatti:

- Il modello è stato addestrato sul dataset **CoNLL-2003**, che contiene testi di tipo *newswire* (notizie giornalistiche).
- Questo dominio è molto distante dai testi clinici, che includono **terminologia medica, riferimenti a date di ricovero/dimissione, nomi di farmaci, ospedali e assicurazioni**, categorie non presenti nel corpus di training.
- Di conseguenza, il modello tende a **confondere entità non rilevanti** (es. "Obesity", "O", "Male") con organizzazioni o altre entità, e manca nel riconoscimento di entità chiave come **DATE** o **PHI** specifici.

In sintesi: **BERT fine-tuned su CoNLL-2003 non è adatto in modo ottimale al dominio clinico**.

6.6.3 Utilizzo di Flair

Oltre Spacy e Bert abbiamo utilizzato Flair, e per la valutazione abbiamo utilizzato il modello flair/ner-english-large, uno dei modelli più potenti messi a disposizione dalla libreria Flair. Flair è noto per ottenere ottimi risultati su task di NER in lingua inglese, soprattutto in domini generici.

Flair è una libreria potente per l'elaborazione del linguaggio naturale. Permette di applicare modelli avanzati su testi per: riconoscimento di entità (NER), analisi del sentiment, analisi grammaticale, supporto per testi biomedici, disambiguazione dei significati e classificazione, e funziona con molte lingue.

Include anche strumenti per le embeddings di parole e testi, permettendo di combinare diversi tipi di rappresentazioni testuali, comprese le Flair embeddings e vari modelli transformer.

Nel nostro setup, Flair è stato configurato per estrarre solo le entità più rilevanti per il nostro caso d'uso (PERSON e ORG), mappando le etichette native (PER, ORG, LOC) sulle categorie standard utilizzate anche da spaCy e BERT.

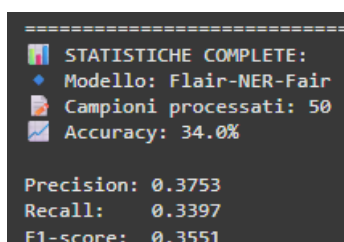


Figure 36: Report Flair

I risultati ottenuti con Flair-NER-Fair mostrano un'accuracy del **34,0%** e valori moderatamente bassi di **precision (0,38)**, **recall (0,34)** e **F1-score (0,36)**.

Questi numeri indicano una capacità limitata del modello nel contesto medico-sanitario. Infatti:

- Flair-NER-Fair è un modello generale, non specificamente addestrato su testi clinici.
- Il dominio clinico presenta nomi di pazienti, medici, ospedali, assicurazioni e date di ricovero/dimissione, categorie spesso assenti nel training originale del modello.
- Nonostante la sua architettura avanzata, la mancanza di adattamento al dominio sanitario riduce le prestazioni quando il focus è sull'anonimizzazione di cartelle cliniche sintetiche.

Quindi: Flair-NER-Fair funziona meglio per entità generiche, ma non è ottimale per dati clinici sensibili o altamente strutturati.

6.7 Confronto tra i vari modelli

Per visualizzare chiaramente le differenze tra i modelli, abbiamo utilizzato un **bar chart comparativo** che mostra **accuracy, precision, recall e F1-score** per ciascun modello NER (spaCy-small, spaCy-medium, spaCy-large, BERT e Flair).

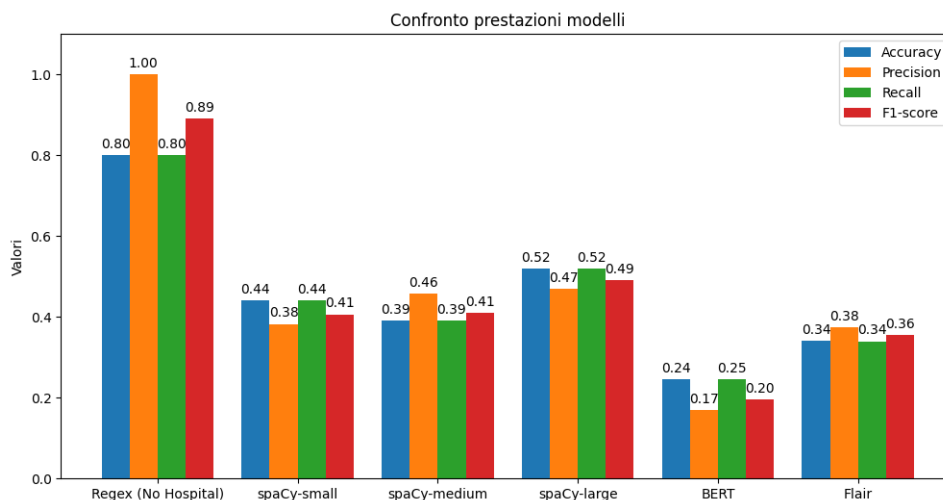


Figure 37: Confronto Modelli (RegEx nclusa)

Sebbene la regex ottenga i risultati migliori su tutte le metriche, questi non tengono conto del fatto che con questa tecnica non viene anonimizzata la feature Hospital, che contiene troppi valori univoci per poter strutturare un'espressione regolare adeguata per l'anonimizzazione.

Tenendo conto di questo aspetto, dal grafico emerge subito che **spaCy-large supera tutti gli altri modelli** su tutte le metriche, confermandone l'efficacia nel contesto clinico. Flair e i modelli spaCy più piccoli hanno performance intermedie, mentre BERT fine-tuned su CoNLL-2003 ottiene i valori più bassi, coerente con il suo training su testi non clinici. Il bar chart permette quindi di confrontare rapidamente le prestazioni e giustifica la scelta di spaCy-large per l'estrazione delle entità sensibili nel nostro workflow di anonimizzazione.

Quindi, per concludere, nel nostro workflow di anonimizzazione abbiamo deciso di utilizzare **spaCy en_core_web_lg**, ritenuto il modello migliore tra quelli testati, per l'estrazione automatica delle entità sensibili, concentrandoci su **PERSON, ORG e DATE**.

Successivamente, applichiamo alcune **regex mirate in post-processing** per gestire casi specifici:

- Sostituzione dei nomi dei medici con [DOCTOR] e dei pazienti con [PATIENT] quando il contesto lo indica (es. presenza di "Dr." o "Doctor").
- Anonimizzazione di compagnie assicurative note come [INSURANCE].

Queste regex **non modificano le predizioni del modello**, ma agiscono solo sul testo anonimizzato risultante.

In aggiunta, abbiamo applicato una **generalizzazione dell'età in fasce** e del **billing amount** in categorie predefinite, così da ridurre ulteriormente la possibilità di identificazione.

Infine, abbiamo eseguito la NER su un sottoinsieme del dataset per generare le **note cliniche anonimizzate**, ottenendo un equilibrio tra accuratezza del riconoscimento delle entità e protezione della privacy dei dati sensibili.

6.7.1 Risposta alla RQ4

Q RQ₄. *Quanto è efficace il Named Entity Recognition (NER) nell'anonimizzare automaticamente le note cliniche generate?*

La NER può anonimizzare alcune entità chiave come date, nomi di pazienti e medici, e assicurazioni, ma le prestazioni dipendono molto dal modello utilizzato. I modelli generici faticano a riconoscere correttamente entità cliniche complesse e introducono errori o confondono categorie simili. I modelli più grandi e sofisticati catturano meglio le informazioni rilevanti, ma non sono perfetti e continuano a mancare alcune entità. In generale, la NER è più flessibile delle regex, ma per un'anonimizzazione completa e affidabile servirebbero modelli addestrati specificamente su testi clinici, eventualmente supportati da regole o dizionari dinamici.

6.8 Sviluppi futuri

Per migliorare ulteriormente il nostro workflow di anonimizzazione, alcuni possibili sviluppi includono:

- **Addestramento di modelli NER specifici per il dominio clinico:** utilizzare corpora medici e sanitari per aumentare la copertura di entità come farmaci, procedure, ospedali o date di ricovero.
- **Espansione delle regole regex e delle categorie di entità:** includere categorie aggiuntive sensibili (es. codici diagnostici, referti di laboratorio, numeri di telefono) e regole più sofisticate basate sul contesto.
- **Integrazione di approcci di anonymization avanzati:** tecniche come k-anonymity, differential privacy o embedding-based masking per proteggere meglio dati strutturati e non strutturati.
- **Utilizzo di combinazione di regex e NER per migliorare le prestazioni:** sfruttare la complementarità tra regole deterministiche e modelli statistici per ridurre falsi positivi e falsi negativi.
- **Utilizzo di modelli generativi:** applicare LLM per riscrivere automaticamente testi clinici in forma anonima preservando la coerenza semantica.