model\_type: llm

base\_model: {model\_path}

input\_features:

  - name: DOMANDE

    type: text

    preprocessing:

      truncation: true

      max\_sequence\_length: 210

      lowercase: true

output\_features:

  - name: RISPOSTE

    type: text

    preprocessing:

      max\_sequence\_length: 1000

      max\_length: 1000

      truncation: true

    metrics:

      - type: bert\_score

        model: dbmdz/bert-base-italian-uncased

      - type: loss

      ### - type: rouge, - type: perplexity

prompt:

  template: >-

    Sei un esperto cardiologo. Rispondi alla seguente domanda in modo empatico e con frasi complete e concise, terminando ogni frase con un punto.

    ### Esempio 1:

    Domanda: Quali sono i sintomi dell'ipertensione?

    Risposta: I sintomi dell'ipertensione includono mal di testa, vertigini e difficoltà respiratorie.

    ### Esempio 2:

    Domanda: Come posso controllare la mia pressione arteriosa?

    Risposta: Puoi controllare la tua pressione arteriosa con una dieta sana, esercizio fisico regolare e farmaci prescritti.

    ### Domanda: {DOMANDE}

    ### Risposta (Assicurati che la tua risposta termini con un punto e abbia senso compiuto):

generation:

  temperature: 0.1    ### per la generazione del testo deve essere molto basso

  max\_new\_tokens: 80

  top\_p: 0.9          ### aiuta a generare testo più coerente e fluido. , il modello avrà un equilibrio tra casualità e coerenza, garantendo che i token generati siano tra quelli con le probabilità più alte.

adapter:

  type: lora

quantization:

  bits: 4

preprocessing:

  split:

    type: random

    probabilities: [0.8, 0.1, 0.1]  ### a gestione di più sottoinsiemi può richiedere risorse aggiuntive.

  global\_max\_sequence\_length: 1210

trainer:

  type: finetune

  epochs: 3

  batch\_size: 2

  eval\_batch\_size: 2

  gradient\_accumulation\_steps: 8  # effective batch size = batch size \* gradient\_accumulation\_steps

  learning\_rate: 1.0e-4  # Prova con un learning rate leggermente più basso

  enable\_gradient\_checkpointing: true

  learning\_rate\_scheduler:

    decay: cosine

    warmup\_fraction: 0.1  # Aumenta il warmup fraction

    reduce\_on\_plateau: 0

### un checkpoint del modello verrà salvato ogni 500 step di addestramento, permettendo il ripristino del modello in caso di interruzioni.

early\_stopping:                             ### monitora la perdita di validazione (val\_loss) e interrompe l'addestramento se la perdita non migliora per 5 epoche consecutive, per prevenire l'overfitting (min: il monitoraggio è configurato per minimizzare la perdita).

  monitor: val\_loss

  patience: 5

  mode: min

logging:                                    ### i log dell'allenamento verranno inviati a TensorBoard.

  log\_to\_tensorboard: true

callbacks:

  - type: progress\_bar

  - type: model\_checkpoint

    save\_best\_only: true

    monitor: val\_loss

    mode: min

  - type: early\_stopping

    monitor: val\_loss

    patience: 5

    mode: min

Immagine che contiene testo, schermata, Diagramma, linea

Descrizione generata automaticamente

Il grafico rappresenta l'andamento della perdita (loss) del modello di addestramento (Training Loss) e di validazione (Validation Loss) su tre epoche.

Analisi del Grafico:

**Andamento della Perdita:**

Training Loss: La linea blu rappresenta la perdita durante l'addestramento del modello. Si osserva una decrescita significativa della perdita dalla prima alla terza epoca. Questo indica che il modello sta apprendendo bene dai dati di addestramento, migliorando le sue previsioni nel tempo.

Validation Loss: La linea arancione rappresenta la perdita durante la fase di validazione del modello. Anche qui si osserva una decrescita della perdita, sebbene con una pendenza leggermente diversa rispetto alla Training Loss. La Validation Loss tende a diminuire più lentamente rispetto alla Training Loss.

Convergenza:

Stabilità: La perdita sia nel training che nella validazione diminuisce in modo consistente. Questo è un segnale positivo, indicando che il modello non solo sta migliorando sulle stesse istanze viste durante l'addestramento, ma sta anche generalizzando meglio sui dati di validazione.

Differenza tra Training e Validation Loss: La differenza tra le due curve è relativamente ridotta, suggerendo che il modello non sta sovra-adattandosi ai dati di addestramento. Se la Validation Loss fosse significativamente più alta della Training Loss, potrebbe indicare un problema di overfitting.

Pendenza:

Diminuzione della Pendenza: Entrambe le curve mostrano una diminuzione della pendenza verso la fine delle epoche. Questo è normale e indica che il modello sta raggiungendo un punto di saturazione, dove ulteriori miglioramenti richiedono più tempo e dati.

Epoche Future:

Possibile Saturazione: Sebbene la perdita stia ancora diminuendo, la riduzione rallenta verso la terza epoca. Questo suggerisce che il modello potrebbe beneficiare di ulteriori epoche di addestramento, ma con un'aspettativa di miglioramenti incrementali minori.

Potenziale di Overfitting: Se si continuasse ad addestrare il modello per molte più epoche senza un'adeguata strategia di regolarizzazione, potrebbe iniziare a sovra-adattarsi ai dati di addestramento.

Conclusioni e Raccomandazioni:

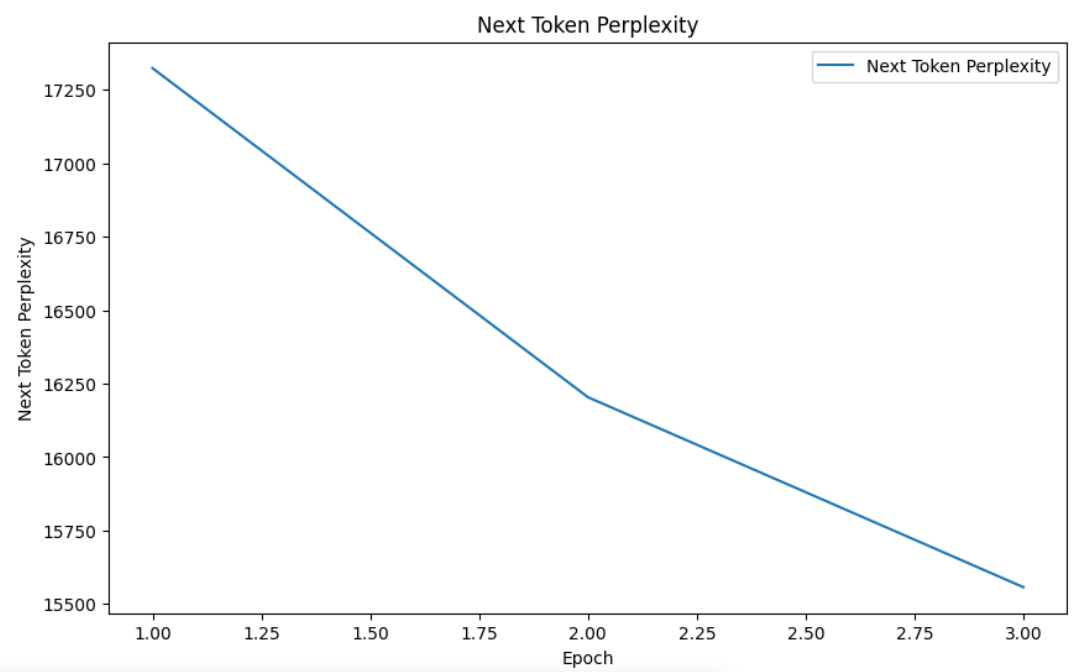
Performance Generale: Il modello sta performando bene, con un miglioramento costante della perdita sia durante l'addestramento che la validazione.

Ulteriori Epoche: Potrebbe essere utile continuare l'addestramento per alcune epoche aggiuntive per vedere se la perdita continua a diminuire in modo significativo.

Monitoraggio dell'Overfitting: Continuare a monitorare la differenza tra Training Loss e Validation Loss per garantire che il modello non inizi a sovra-adattarsi ai dati di addestramento.

Fine-tuning dei Parametri: Considerare l'ottimizzazione di altri iperparametri, come il tasso di apprendimento o il dropout, per vedere se possono contribuire a miglioramenti aggiuntivi nella performance del modello.

In sintesi, il grafico mostra che il modello sta apprendendo in modo efficace, con una buona tendenza di diminuzione della perdita. Continuare l'addestramento e il monitoraggio può aiutare a ulteriormente migliorare le sue capacità di generalizzazione.



Il grafico mostra l'andamento della perplessità del token successivo (Next Token Perplexity) nel corso di tre epoche di addestramento.

**1. Andamento della Perplessità:**

La perplessità è una misura di quanto bene un modello prevede una distribuzione di probabilità. In generale, una perplessità più bassa indica una migliore capacità predittiva del modello.

Nel grafico, si osserva una chiara diminuzione della perplessità del token successivo con l'aumentare delle epoche. Questo suggerisce che il modello sta migliorando nel prevedere il token successivo man mano che l'addestramento procede.

**2. Diminuzione Costante:**

La perplessità scende costantemente da oltre 17.250 a poco più di 15.500. La riduzione è particolarmente significativa tra la prima e la seconda epoca, con una pendenza più marcata. Questo può indicare che il modello ha appreso molto nelle prime fasi dell'addestramento.

Dopo la seconda epoca, la perplessità continua a diminuire, ma ad un ritmo leggermente più lento. Questo è normale e suggerisce che il modello stia raggiungendo un punto di convergenza.

**3. Significato della Perplessità:**

Una riduzione della perplessità significa che il modello sta diventando più efficace nel prevedere il prossimo token nella sequenza. Questo è cruciale per i modelli di linguaggio poiché indica una maggiore comprensione del contesto e una migliore generazione di testo. La perplessità è particolarmente utile per valutare modelli di linguaggio perché tiene conto delle distribuzioni di probabilità. Un valore più basso è sempre desiderabile, indicando un modello più performante.

**4. Implicazioni Pratiche:**

Modello Migliorato: La riduzione della perplessità indica che il modello è diventato più affidabile e accurato nella previsione del token successivo. Questo migliora la qualità del testo generato, rendendolo più coerente e contestualmente rilevante.

Potenziale Ulteriore Addestramento: Sebbene la perplessità stia diminuendo, il grafico suggerisce che potrebbero essere necessari ulteriori epoche di addestramento per continuare a migliorare le performance del modello, soprattutto se si osserva che la perplessità non si è ancora stabilizzata.

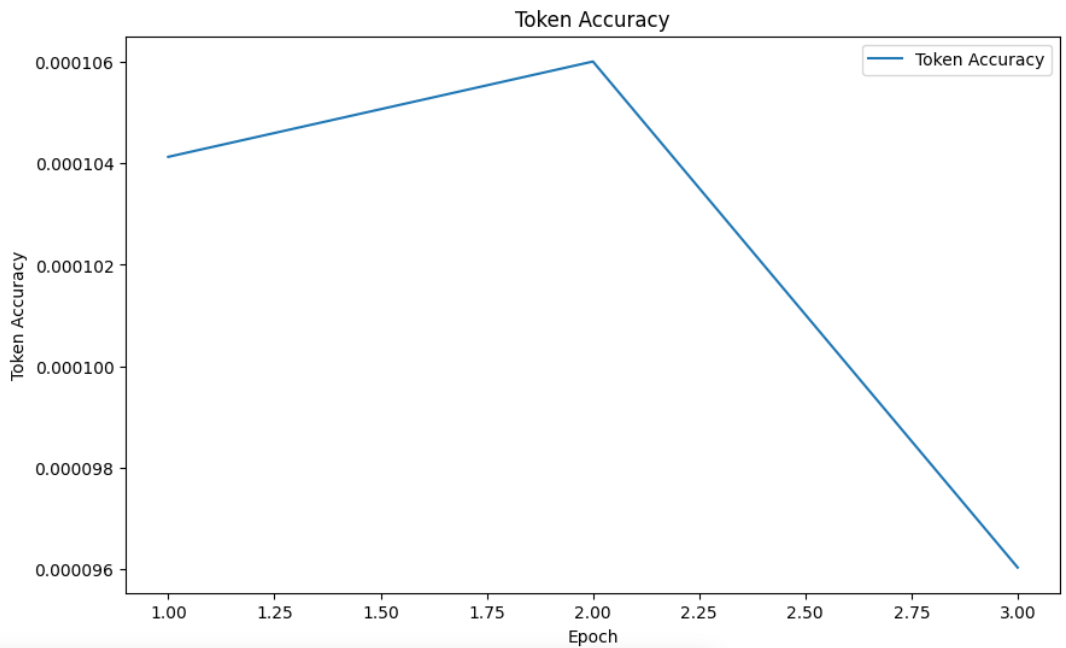
**Conclusioni e Raccomandazioni:**

Continua l'Addestramento: Poiché la perplessità sta ancora diminuendo, si consiglia di proseguire con ulteriori epoche di addestramento per vedere se i miglioramenti continuano.

Valutazione dei Risultati: Monitorare altre metriche di performance insieme alla perplessità per avere una visione completa delle capacità del modello.

Tuning degli Iperparametri: Potrebbe essere utile sperimentare con diversi iperparametri (come il tasso di apprendimento) per ottimizzare ulteriormente il modello.

In sintesi, il grafico di "Next Token Perplexity" mostra un miglioramento costante nelle capacità predittive del modello, suggerendo un buon processo di addestramento con potenziale per ulteriori miglioramenti.



Analisi del Grafico "Token Accuracy"

Il grafico mostra l'andamento dell'accuratezza del token (Token Accuracy) nel corso di tre epoche di addestramento.

**1. Andamento dell'Accuratezza:**

L'accuratezza del token è una misura di quanto correttamente il modello prevede i singoli token rispetto alle risposte attese.

Nel grafico, si osserva un andamento non lineare dell'accuratezza del token. L'accuratezza inizialmente aumenta dalla prima alla seconda epoca, ma poi diminuisce significativamente dalla seconda alla terza epoca.

**2. Aumento e Diminuzione dell'Accuratezza:**

Prima e Seconda Epoca: L'accuratezza del token aumenta leggermente, passando da circa 0.000104 a 0.000106. Questo suggerisce che, inizialmente, il modello stava migliorando nel prevedere correttamente i token.

Seconda e Terza Epoca: L'accuratezza del token diminuisce drasticamente, passando da 0.000106 a circa 0.000096. Questa riduzione significativa potrebbe indicare che il modello ha iniziato a sovra-adattarsi ai dati di addestramento o che ci sono state fluttuazioni nel processo di addestramento.

3. Interpretazione dell'Accuratezza:

Aumento Iniziale: L'aumento iniziale dell'accuratezza suggerisce che il modello stava apprendendo efficacemente e migliorando la sua capacità di prevedere correttamente i token.

Diminuzione Successiva: La drastica diminuzione potrebbe essere causata da diversi fattori:

Sovra-adattamento: Il modello potrebbe aver iniziato a sovra-adattarsi ai dati di addestramento, portando a una riduzione dell'accuratezza.

Rumore nei Dati: Potrebbe esserci del rumore nei dati di addestramento che ha influenzato negativamente l'accuratezza.

Iperparametri: Gli iperparametri utilizzati (ad esempio, il tasso di apprendimento) potrebbero non essere ottimali, causando fluttuazioni nell'accuratezza.

**4. Implicazioni Pratiche:**

Monitoraggio dell'Addestramento: È importante monitorare attentamente l'accuratezza durante l'addestramento per identificare eventuali problemi come il sovra-adattamento.

Tuning degli Iperparametri: Potrebbe essere utile sperimentare con diversi iperparametri per vedere se è possibile stabilizzare e migliorare ulteriormente l'accuratezza del token.

Valutazione Completa: Utilizzare altre metriche di performance, come la perplessità e la perdita, insieme all'accuratezza del token, per avere una visione completa delle capacità del modello.

Conclusioni e Raccomandazioni:

Continua l'Addestramento con Cautela: Potrebbe essere necessario continuare l'addestramento con aggiustamenti agli iperparametri per vedere se l'accuratezza del token può essere stabilizzata e migliorata.

Valutazione dei Dati: Analizzare i dati di addestramento per identificare eventuali problemi di rumore che potrebbero influenzare negativamente l'accuratezza.

Esperimenti con Iperparametri: Sperimentare con diversi tassi di apprendimento e altre configurazioni per trovare l'impostazione ottimale che stabilizza l'accuratezza del token.

In sintesi, il grafico di "Token Accuracy" mostra un andamento non lineare dell'accuratezza del token, suggerendo la necessità di ulteriori analisi e possibili aggiustamenti nel processo di addestramento per migliorare le performance del modello.  
  
Immagine che contiene testo, linea, diagramma, Diagramma

Descrizione generata automaticamente

Analisi del Grafico a Dispersione per Epoch

Il grafico presenta una visualizzazione a dispersione delle metriche di addestramento, validazione e test su tre epoche per tre metriche chiave: Loss, Perplexity e Token Accuracy.

**1. Loss per Epoch**

Training Loss: La perdita di addestramento mostra un andamento decrescente dall'epoca 1 all'epoca 3. Questo è un segnale positivo, indicando che il modello sta apprendendo e adattandosi meglio ai dati di addestramento.

Validation Loss: Anche la perdita di validazione decresce, ma con una riduzione più stabile e meno pronunciata rispetto alla perdita di addestramento.

Test Loss: La perdita di test è la più bassa tra le tre, suggerendo che il modello generalizza abbastanza bene sui dati di test. Tuttavia, è importante notare che la perdita di test aumenta leggermente all'epoca 2 prima di diminuire nuovamente all'epoca 3.

**2. Perplexity per Epoch**

Training Perplexity: La perplessità di addestramento mostra una leggera fluttuazione, con una tendenza complessiva a diminuire, indicando un miglioramento della capacità del modello di prevedere correttamente i token successivi.

Validation Perplexity: La perplessità di validazione è relativamente stabile, con una leggera diminuzione. Questo suggerisce che il modello sta migliorando nella sua capacità di generalizzare le previsioni sui dati di validazione.

Test Perplexity: La perplessità di test è leggermente più alta rispetto alla validazione, ma mostra una tendenza simile, con una leggera diminuzione complessiva.

**3. Token Accuracy per Epoch**

Training Token Accuracy: L'accuratezza dei token di addestramento mostra una fluttuazione significativa. Dopo un aumento iniziale, diminuisce alla terza epoca. Questo potrebbe indicare che il modello sta affrontando difficoltà nel migliorare ulteriormente l'accuratezza dei token.

Validation Token Accuracy: L'accuratezza dei token di validazione è stabile ma bassa, suggerendo che il modello non sta migliorando significativamente la sua accuratezza sui dati di validazione.

Test Token Accuracy: L'accuratezza dei token di test mostra una fluttuazione simile a quella dell'addestramento, ma con valori leggermente inferiori, indicando che il modello ha difficoltà nel mantenere una buona accuratezza sui dati di test.

**Implicazioni e Raccomandazioni**

Riduzione della Perdita: La riduzione della perdita di addestramento e di validazione è un segnale positivo, ma l'aumento temporaneo della perdita di test suggerisce che potrebbero esserci dei problemi di generalizzazione. Potrebbe essere utile introdurre tecniche di regolarizzazione per mitigare questo problema.

Perplessità: La perplessità in diminuzione è un buon segnale, ma le fluttuazioni suggeriscono che il modello potrebbe beneficiare di ulteriori iterazioni di addestramento o di un tuning più fine degli iperparametri.

Accuratezza dei Token: Le fluttuazioni nell'accuratezza dei token indicano che il modello potrebbe non essere stabile. È importante monitorare questo andamento e considerare l'uso di tecniche di stabilizzazione, come la riduzione del tasso di apprendimento o l'uso di gradient clipping.

Iperparametri: Potrebbe essere necessario sperimentare con diversi set di iperparametri per trovare una configurazione che migliori la stabilità e le performance del modello.

**Conclusioni**

Il grafico mostra un miglioramento generale nelle metriche di perdita e perplessità, ma evidenzia fluttuazioni nell'accuratezza dei token. Per ottenere un modello più stabile e con migliori performance, è consigliabile continuare a monitorare le metriche, effettuare ulteriori iterazioni di addestramento e sperimentare con diverse configurazioni di iperparametri.