BERT for log classifying, clustering and interpretation

Andrei Pârjol

May 2024

1 Introducere

Acest articol prezintă implementarea, antrenarea și folosirea modelului BERT pentru a clasifica , clusteriza și interpreta fișiere cu log-uri. Am folosit timp îndelungat acest model antrenat de mine la locul de muncă (data analyst) pentru a automatiza analiza fișierelor log (clasificare și grupare) și pentru interpretarea și fixarea bug-urilor din cod.

Modelul permite o analiză rapidă a unui volum mare de date cu acuratețe ridicată.

Codul sursă pentru antrenarea modelului și testare poate să fie găsit pe Github : https://github.com/andreiparjolfac/bert-for-log-classifying-clustering-and-interpretation

2 Modelul NLP folosit

Modelul **BERT** (prescurtarea de la **B**idirectional **E**ncoder **R**epresenations from **T**ransformers) este un model deep learning pentru procesarea limbajului natural bazat pe arhitectura "transformer-ilor", dezvoltat și menținut de către departamentul de cercetare Google. În prezent sunt disponibile două versiuni ale modelului : $BERT_{BASE}$ compus din 12 encodere cu 12 scannere bidirecționale având aprox. 110 milioane de parametrii de intrare sau $BERT_{LARGE}$ compus din 24 encodere cu 16 scannere bidirectionale având aprox. 340 milioane de parametrii de intrare. Ambele modele vin pre-antrenate folosind corpusul Toronto BookCorpus(800 milioane cuvinte) și Enciclopedia Wikipedia (2,5 miliarde de cuvinte).

În acest model NLP, textul este transformat in reprezentări numerice numite tokens iar apoi fiecare token este transformat intr-un vector folosind o tabelă *"word embedding"*. Prin această metodă se reușește să se codifice și contextul cuvintelor. [Dev+19]

3 Clasificare

Primul scop al modelului este de a putea să clasifice și să eticheteze fișierele log după eroarea pe care descriu : NO_ERROR, GAME_ISSUE, SERVER_ISSUE, CONTENT_ERROR, etc.

Modelul va fi antrenat folosind învățarea suprevizată în care vom folosi fișiere log și etichetele asociate acestora. Pentru implementare vom folosi modelul preantrenat $BERT_{BASE}$ deoarece vocabularul din fișierele log este relativ mic și dorim un model cât mai rapid pentru a clasifica cât mai multe fișiere. Înainte de a fi trimise ca input pentru model (fie pentru evaluare sau învățare) fișierele log necesită o preprocesare care constă în împărțirea fiecarui fișier în mai multe chunk-uri (fiecare chunk conținând maxim 512 cuvinte/tokens) , cel mai important chunk fiind ultimul deoarece sfârșitul unui fișier log descrie eroarea produsă. Fiecare chunk este delimitat printr-un caracter special numit $[SEP]$ și este folosit pentru a clasifica fișierul din care provine. De asemenea putem să definim tokeni speciali în caz că vrem să ignorăm username-uri, ora/data, numele server-ului, etc.

În implementare modelul a avut o accuratețe de 97% pe mulțimea de antrenament compusă din aproximativ 300 de mii de fișiere log etichetate și o accuratețe de 95% pentru 2000 fișiere log noi.

4 Clustering

Pentru partea de file/text Clustering vom folosi libraria Python *Faiss* dezvoltată de Facebook AI ce ne permite să definim o "distanță" pentru fișierele text și să aplicăm un model k-NN pentru a le grupa. Acest cluster poate fi folosit mai departe pentru a descoperi anumite pattern-uri care apar de exemplu probleme recurente sau un număr anormal de mare de issue-uri de un anumit tip. [Dou+24]. Libraia Faiss are nevoie doar de partea care este responsabilă cu transformarea fișierului text într-un vector embedded , deci poate să fie considerată o versiune mai ușoară de clasificare din moment ce putem să clasificăm un fișier și apoi să căutăm toate fișierele asemanătoare.

5 Interpretare

Modelul BERT poate fi folosit și pentru intepretarea fișierelor log , precum sumarizarea evenimentelor sau descrierea pașilor pentru a reproduce un anumit bug/issue. Acest lucru este posibil deoarece reprezentarea embedded păstrează și contextul cuvintelor.

6 Concluzie

Modelul NLP BERT are foarte multe aplicații în domeniul precesării limbajului natural. Pentru analiza fișierelor log text acesta este capabil să clasifice și să grupeze un volum mare de fișiere text cu o accuratețe ridicată. Reprezentarea modelului și librăriile disponibile(precum Keras, Pytorch sau Faiss) fac ușoară învățarea unui model pre-antrenat pentru a fi folosit în scopuri diferite.

References

[Dev+19] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

[Dou+24] Matthijs Douze et al. "The Faiss library". In: (2024). arXiv: 2401.08281 [cs.LG].