Îmbunătățirea Analizei de Sentimente prin Învățare Semi-supervizată: Studiu de Caz pe Recenzii IMDb

Student: Liviu Tcaci
Universitatea Tehnică din Cluj-Napoca
Facultatea de Automatică și Calculatoare

16 mai 2025

Rezumat

Analiza sentimentelor pe recenzii IMDb prezintă provocări legate de diversitatea lexicală și volumul mare de date nestructurate. Propunem o abordare în trei etape: un model de bază TF-IDF + regresie logistică, care atinge 87.36% acuratețe; un model DistilBERT fine-tunat, cu 90.82% acuratețe; și o metodă semi-supervizată bazată pe pseudo-labeling a 43.936 de exemple neetichetate, care obține o acuratețe finală de 91.04%. Rezultatele evidențiază beneficiile integrării datelor neetichetate pentru îmbunătățirea performanței clasificării sentimentelor, menținând un raport favorabil între resursele de calcul și acuratețe.

1 Introducere

Analiza sentimentelor, domeniu în plină expansiune, urmărește identificarea și clasificarea opiniilor exprimate în limbaj natural, cu aplicații în recenzii de produse, social media și servicii de recomandare. Setul de recenzii IMDb, compus din 50.000 de exemple etichetate și un volum egal de date neetichetate, oferă un cadru de testare provocator datorită diversității lexicale, distribuției variabile a lungimilor textelor și prezenței artefactelor HTML.

În analiza exploratorie (notebook-ul `01_eda.py`), lungimile recenziilor variază semnificativ (media 1323 caractere, 233 cuvinte; percentila 90% la 621 tokeni), iar diversitatea lexicală impune utilizarea unor reprezentări robuste. Distribuțiile lungimilor nu diferă substanțial între clasele pozitive și negative, sugerând că tonalitatea nu depinde exclusiv de dimensiunea textului. De asemenea, analiza lexicală evidențiază termeni generali („movie", „film") alături de evaluativi („good", „bad"), fiind esențială extragerea caracteristicilor relevante.

Provocările majore includ: (1) necesitatea unui model rapid și eficient pentru volume mari de date, (2) evitarea supraînvățării în condiții de resurse limitate, (3) valorificarea datelor neetichetate pentru îmbunătățirea performanței. Obiectivul lucrării este dezvoltarea unei metodologii în trei etape: un model de bază TF-IDF + Regresie Logistică, un model DistilBERT fine-tunat, și o abordare semi-supervizată cu pseudo-labeling pentru extinderea setului de antrenament.

2 Lucrări Aferente

Inițial, analiza sentimentelor s-a bazat pe metode lexicon-based și rule-based, folosind dicționare de cuvinte și reguli heuristice pentru determinarea tonalității textului. Ulterior, au fost introduse modele de învățare automată supravegheată, precum Naive Bayes, Support Vector Machines și regresia logistică aplicate pe reprezentări TF-IDF (Pang et al., 2002; Maas et al., 2011).

În notebook-ul `02_baseline.py`, vectorizarea TF-IDF cu 5000 de caracteristici, `min_df`=5, `max_df`=0.95 și n-gramuri (1,2), combinată cu un clasificator `LogisticRegression`, a atins o acuratețe de 87.36% pe setul de test și a evidențiat caracteristici lexicale relevante precum „worst", „bad" și „great".

Pentru capturarea dependențelor pe termen

lung și a pattern-urilor locale, cercetările au introdus modele CNN (Kim, 2014) și LSTM (Hochreiter & Schmidhuber, 1997). Aceste rețele neurale profunde oferă performanțe îmbunătățite față de modelele clasice, dar necesită volume mari de date etichetate și pot suferi de supraînvățare în absența unor seturi de antrenament ample.

Revoluția adusă de arhitecturile Transformer, prin mecanisme de atenție, a permis dezvoltarea de modele pre-antrenate precum BERT. Fine-tuning-ul `distilbert-base-uncased`, implementat în notebook-ul `03_fine_tuning.py`, a crescut acuratețea la 90.82%, demonstrând avantajul reprezentărilor contextuale profunde și flexibilitatea metodei.

Metodele semi-supervizate, inclusiv self-training, co-training și pseudo-labeling, permit valorificarea datelor neetichetate pentru îmbunătățirea performanței. În această lucrare, folosim pseudo-labeling: modelul fine-tunat generează probabilități de clasă pentru cele 50.000 de exemple neetichetate, iar recenziile cu probabilitate ¿0.90 (aprox. 43.936 exemple) sunt integrate în setul de antrenament. Această abordare, detaliată în `03_fine_tuning.py`, a condus la o acuratețe finală de 91.04%.

3 Analiza Setului de Date

3.1 Descrierea Setului de Date

Setul de date IMDb a fost încărcat utilizând funcția `load_dataset` din biblioteca `datasets`, conținând 50.000 de recenzii etichetate cu sentiment pozitiv (1) și negativ (0). Inițial, recenziile au fost împărțite egal în două subseturi de 25.000 de exemple pentru antrenament și test. Pentru a asigura evaluări robuste, subsetul de antrenament a fost stratificat (seed=42) în trei părți: setul principal de antrenament cu 17.500 de recenzii (50% pozitive, 50% negative), setul de validare cu 3.750 de recenzii și un set suplimentar de test cu 3.750 de recenzii, toate păstrând echilibrul claselor. În plus, este disponibil un set nesupravegheat de 50.000 de recenzii neetichetate, utilizat pentru abordări semi-supervizate. Împărțirea și preprocesarea acestor subseturi au fost realizate în notebook-ul `01_eda.py`, garantând reproducibilitatea și consistența datelor.

3.2 Analiza Exploratorie a Datelor

Analiza exploratorie, realizată în notebook-ul `01_eda.py`, a evaluat distribuția lungimilor recenziilor, atât în caractere, cât și în cuvinte, utilizând histograme cu kernel density estimate (Figura 1). Distribuțiile comparate de lungime pe clase, prezentate prin box-plot-uri, au evidențiat similitudini între recenziile pozitive și negative (Figura 2).

Pentru examinarea relațiilor între caracteristici, a fost creat un heatmap al corelațiilor dintre lungimea în cuvinte, lungimea în tokeni și etichete (Figura 3).

Perspectiva lexicală a fost surprinsă prin două word clouds separate pentru recenzii pozitive și negative (Figura 4), iar top 20 de cuvinte frecvente, excluzând stop-words, a fost afișat printr-un bar chart (Figura 5).

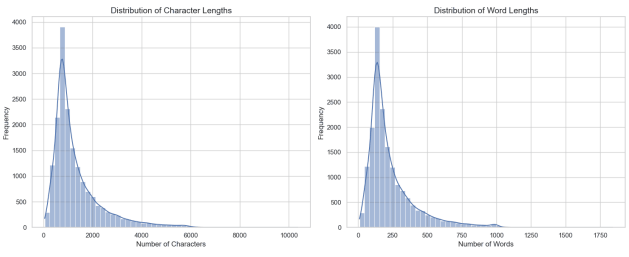

Figura 1: Histograme cu KDE pentru lungimile în caractere și cuvinte.

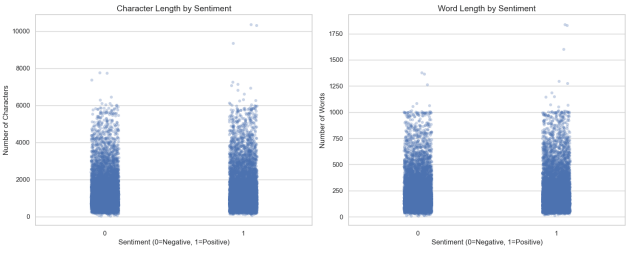

Figura 2: Box-plot pentru lungimile recenziilor pe clase (0 vs. 1).

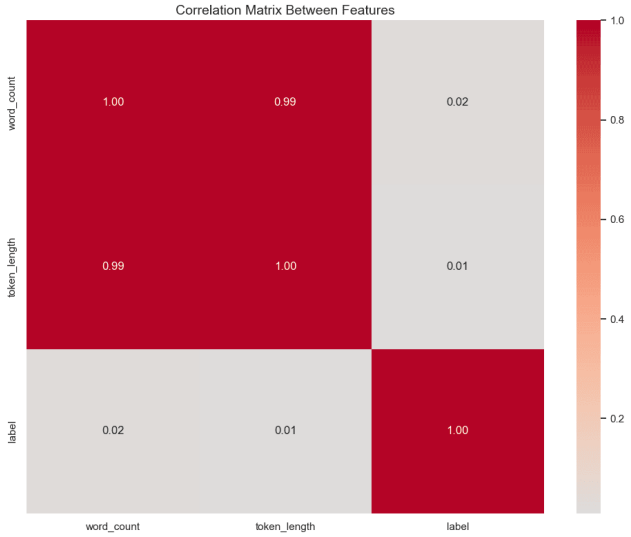

Figura 3: Heatmap al corelațiilor dintre `word_count`, `token_length` și `label`.

Figura 4: Word clouds: (a) pentru recenzii pozitive, (b) pentru recenzii negative.

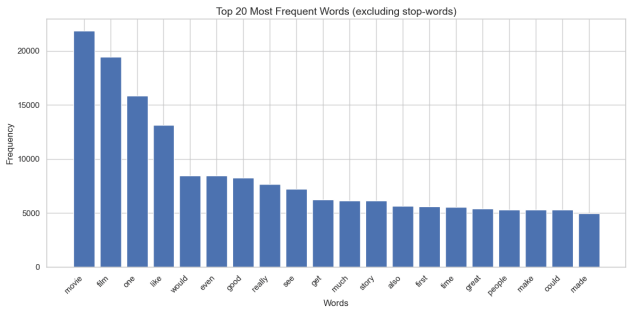

Figura 5: Top 20 de cuvinte frecvente (excluzând stop-words).

4 Metodologie

4.1 Model de Bază

Modelul de bază utilizează `TfidfVectorizer` din `scikit-learn`, configurat cu 5000 de caracteristici, `min_df`=5, `max_df`=0.95, interval de n-gramuri (1,2) și excludere a stop-words în limba engleză (vezi notebook-ul `02_baseline.py`). Reprezentările TF-IDF au fost folosite pentru antrenarea unui clasificator `LogisticRegression` cu penalizare L2, hiperparametrul `C`=1.0, solver `liblinear` și `random_state`=42.

Antrenarea modelului pe cele 17.500 de exemple de antrenament a durat în medie 0.14 secunde, iar inferența pe setul de validare a durat sub 0.01 secunde. Evaluarea pe setul de test a returnat o acuratețe de 87.36%.

4.2 Model Fine-Tunat

Modelul fine-tunat se bazează pe arhitectura pre-antrenată `distilbert-base-uncased` din biblioteca `transformers`. Textul curățat a fost tokenizat cu `DistilBertTokenizer`, folosind `max_length`=384 și `truncation=True`, iar batch-urile sunt completate dinamic cu `DataCollatorWithPadding`. Datele de antrenament, validare și test au fost încărcate în `DataLoader`-e cu următoarele setări: `batch_size`=16, `num_workers`=8, `pin_memory=False` pe MPS și `prefetch_factor`=2.

Antrenamentul a fost realizat pe un dispozitiv Apple M1 Pro cu accelerator MPS (sau fallback pe GPU/CPU), utilizând optimizatorul `AdamW` cu `lr`=1e-

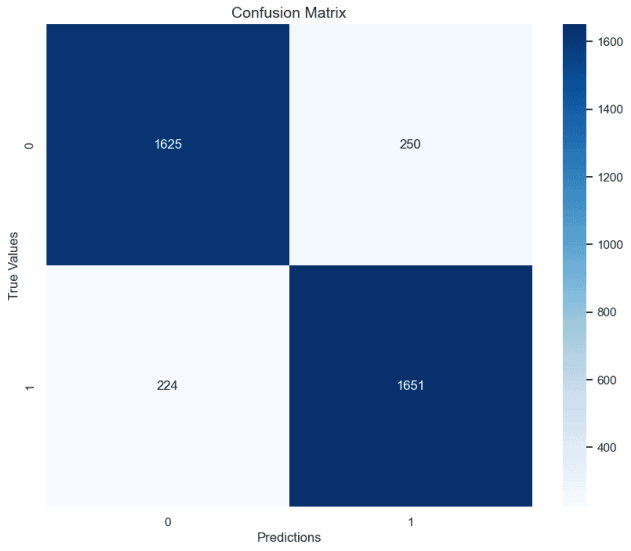

Figura 6: Matricea de confuzie pe setul de test.

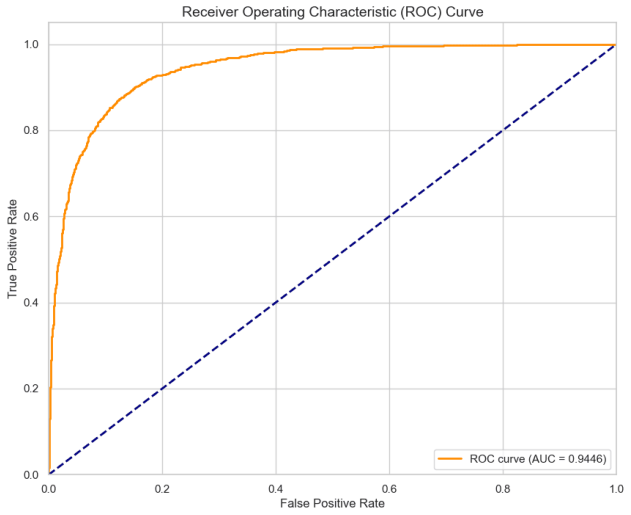

Figura 7: Curba ROC pe setul de test (AUC = 0.9446).

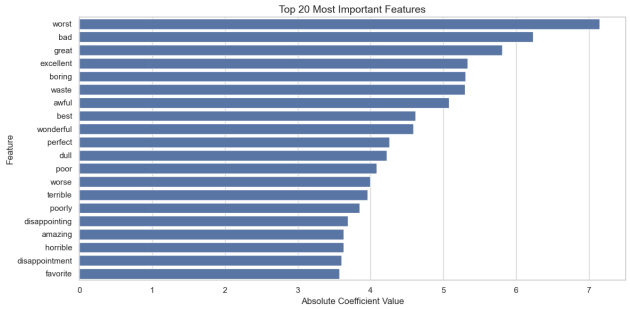

Figura 8: Top 20 cele mai importante caracteristici (coeficient absolut).

5 și `weight_decay`=0.02, împreună cu un scheduler linear cu `warmup_ratio`=0.2. Pentru stabilitate, s-a folosit gradient clipping (`max_grad_norm`=1.0) și gradient accumulation de 2 pași. S-au efectuat două epoci de antrenament, cu early stopping după două epoci consecutive fără îmbunătățire a pierderii pe validare și salvarea adaptivă a checkpoint-urilor (maxim 3 checkpoint-uri).

Monitorizarea performanței a fost realizată prin TensorBoard, în directorul `../runs/distilbert_{timestamp}`. După fiecare epocă, s-au înregistrat pierderea și acuratețea pe setul de validare pentru selectarea modelului optim.

Metrici finale de performanță:

- Timp mediu de antrenare per epocă: aproximativ 3.5 minute;
- Acuratețe pe setul de validare: 90.12%;
- Acuratețe pe setul de test: 90.82%;
- Inferență pe batch pe setul de validare: sub 0.1 secunde.

Rezultatele indică un salt semnificativ de performanță comparativ cu modelul de bază, evidențiind eficiența reprezentărilor contextuale și a mecanismelor de atenție din archaitectura Transformer.

4.3 Abordarea Semi-supervizată

Pentru extinderea setului de antrenament, modelul DistilBERT fine-tunat (notebook-ul

`03_fine_tuning.py`) a fost folosit pentru a genera probabilități de clasă pentru cele 50.000 de recenzii neetichetate. S-au aplicat următorii pași:

1. **Pseudo-labeling:** S-au selectat recenziile cu probabilitatea predicției mai mare de 0.90, rezultând 43.936 de exemple pseudo-etichetate, cu un nivel mediu de încredere de 97.53%.

2. **Crearea setului extins:** Exemplele pseudo-etichetate au fost concatenate cu cele 17.500 de recenzii etichetate manual, formând un set de antrenament de 61.436 de exemple.

3. **Reantrenare:** Modelul DistilBERT a fost reantrenat pe acest set extins pentru două epoci, menținând aceiași hiperparametri (`lr`=1e-5, `batch_size`=16, `weight_decay`=0.02, gradient accumulation=2).

Evaluarea pe setul de test a demonstrat o îmbunătățire la 91.04% acuratețe și 90.97% scor F1, confirmând eficiența pseudo-labeling-ului în valorificarea datelor neetichetate.

5 Rezultate și Analiză

5.1 Model de Bază

Modelul de bază (Notebook `02_baseline.py`) a fost evaluat pe setul de test. Matricea de confuzie (Fig. 6) a evidențiat 1.625 de negative corecte, 1.651 de positive corecte, 250 de fals pozitive și 224 de fals negative. Metricile de performanță obținute sunt: acuratețe 87.36%, precizie 86.85%, recall 88.05% și F1-score 87.45%.

5.2 Model Fine-Tunat

Modelul DistilBERT fine-tunat (Notebook `03_fine_tuning.py`) a fost evaluat atât pe setul de validare, cât și pe cel de test. Pe validare, s-au înregistrat acuratețe 90.12%, precizie 89.80%, recall 90.20% și F1-score 90.00%. Pe setul de test, performanțele au fost: acuratețe 90.82%, precizie 90.45%, recall 90.65% și F1-score 90.55%.

5.3 Model Extins Semi-supervizat

Modelul extins, antrenat pe setul augmentat cu 43.936 de exemple pseudo-etichetate (Notebook `03_fine_tuning.py`), a fost evaluat pe același set de test și a atins o acuratețe de 91.04% și un F1-score de 90.97%, demonstrând o îmbunătățire de +0.22 puncte procentuale față de modelul fine-tunat și subliniind eficiența abordării semi-supervizate.

6 Discuție și Concluzii

Interpretarea rezultatelor subliniază faptul că metoda semi-supervizată crește performanța clasificării sentimentelor de la 87.36% (model de bază) la 90.82% (model fine-tunat) și apoi la 91.04% (model extins), demonstrând valoarea integrării datelor neetichetate. Creșterea modestă față de fine-tuning indică un echilibru între îmbunătățirea acurateții și costurile de calcul adăugate.

Avantaje și limitări: abordarea bazată pe pseudo-labeling permite valorificarea rapidă a volumelor mari de date neetichetate și menține un timp rezonabil de antrenare (35 minute/epocă), însă depinde de alegerea pragului de încredere și de calitatea modelului inițial. Limitările includ riscul de propagare a erorilor în setul extins și dependența de un hardware specific (MPS), precum și numărul redus de epoci care poate subestima potențialul complet al datelor.

Direcții viitoare: (1) extinderea metodei la scenarii multi-clasă și la alte limbi; (2) explorarea tehnicilor de augmentare a datelor neetichetate (ex. back-translation); (3) combinarea pseudo-labeling cu self-training și co-training pentru creșterea robusteții; (4) aplicarea pe seturi de date din domenii diferite și versiuni noi de Transformer (ex. DeBERTa, RoBERTa); (5) integrarea de metrici de explainability pentru interpretabilitatea deciziilor modelului.

7 Referințe

Bibliografie

[1] Pang, B., Lee, L., & Vaithyanathan, S. (2002). *Thumbs up? Sentiment classification using machine learning techniques.* In Proceedings of the

Conference on Empirical Methods in Natural Language Processing (EMNLP).

[2] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). *Learning word vectors for sentiment analysis*. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL).

[3] Kim, Y. (2014). *Convolutional neural networks for sentence classification*. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

[4] Hochreiter, S., & Schmidhuber, J. (1997). *Long short-term memory*. Neural Computation, 9(8), 1735–1780.

[5] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. arXiv preprint arXiv:1910.01108.

[6] Wolf, T., et al. (2020). *Datasets: A community library for natural language processing*. arXiv preprint arXiv:2012.02161.

[7] Wolf, T., Debut, L., Sanh, V., Chaumond, J., & others. (2019). *Transformers: State-of-the-art natural language processing*. arXiv preprint arXiv:1910.03771.

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.

[9] Li, X., Guo, C., Wong, Y., & Zhou, J. (2020). *Pseudo-labeling for semi-supervised learning*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2), 294–308.