Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Adam Mikolašek, Veronika Žatková

Určovanie sentimentu filmových recenzií

Projekt z predmetu NSiete

Študijný program: Inteligentné softvérové systémy
Ak. rok: 2019/2020, zimný semester
Cvičiaci: Ing. Matúš Pikuliak

Klasifikácia sentimentu vo filmových recenziách

1. Motivácia

Veľké množstvo ľudí sa pri výbere filmov riadi práve názormi a recenziami iných. Určenie sentimentu sa neobmedzuje iba na recenzie, ale je využiteľné pri drvivej väčšine rečových prejavov.

Schopnosť určiť sentiment recenzií nemusí byť len finálnym produktom, ale aj základom pre ďalšie analýzy. Pokiaľ vieme určiť sentiment recenzií filmu, agregáciou sentimentov identifikovaných pri filme vieme odhadnúť celkový názor verejnosti na daný film a využívať ho ako alternatívny zdroj hodnotenia filmov. Určenie negatívneho sentimentu je zároveň jedným z prvých krokov pre pomoc pri identifikovaní úmyselne nenávistných komentárov.

2. Súvisiace práce

Dataset, ktorý sme pre danú problematiku našli, bol už využitý v inej práci[1]. Cieľom tejto práce bolo určovanie sentimentu a sémantiky textu, a využitie týchto informácií na klasifikáciu komentárov do dvoch skupín a to podľa toho či komentár patril k pozitívnemu alebo negatívnemu hodnoteniu. Výsledky navrhnutého modelu boli testované na dvoch datasetoch, “Pang and Lee Movie Review Dataset” a “Large Movie Review Dataset”, pričom druhý z týchto datasetov je bližšie opísaný v kapitole 3. Výsledky, ktoré navrhnutý model dosiahol boli porovnané s rôznymi inými technikami na spracovanie prirodzeného jazyka, ako sú Bag-of-words, Latent semantic analysis (LSA) a Latent Drichlet allocation (LDA). Navrhnutý model dosiahol najlepšie výsledky práve v spojení s technikou Bag-of-Words a to 88.89, teda v 88.89% prípadoch kategorizoval komentár správne.

[1] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. (2011). Learning Word Vectors for Sentiment Analysis. *The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).*

3. Dataset

V rámci nášho projektu sme sa rozhodli využiť dataset filmových recenzií “Large Movie Review Dataset v1.0”[2], publikovaný univerzitou Stanford. Dataset pozostáva z 50 000 výrazne polárnych filmových recenzií, mapovateľných na URL adresu filmu, ku ktorému patria (URL sú súčasťou datasetu). Recenzie sú oanotované - každá recenzia má priradenú hodnotu “pozitívna” alebo “negatívna” na základe jej sentimentu. Celková početnosť týchto dvoch tried je vyvážená - 25 000 rezencií je pozitývnych a 25 000 negatívnych. Výrazná polarita recenzií znamená, že negatívne recenzie patria k hodnoteniam <= 4 z 10 a pozitívne recenzie k hodnoteniam >= 7 z 10.

Maximálny počet recenzií pre jeden film je 30 (pre predídenie predpovedania rovnakých hodnôt sentimentu pre jeden film). Dataset je taktiež rozdelený na trénovaciu a testovaciu množinu v rovnakom pomere (25 000 : 25 000), pričom platí, že žiaden z filmov v trénovacej množine sa nenachádza v testovacej množine. K recenziám je taktiež dostupné ich pôvodné hodnotenie vo formáte počtu bodov (maximum je 10).

Súčasťou datasetu je aj ďalších 50 000 recenzií filmov, ktoré sú však neanotované (napríklad pre využitie v rámci metód učenia bez učiteľa). Jedná sa najmä o neutrálne recenzie.

4. Návrh riešenia

Naše riešenie pozostáva z viacerých krokov:

1. Exploratívna analýza dát.
2. Predspracovanie dát - v našom prípade výrazne využívajúce prístupy spracovania prirodzeného jazyka
3. Vytvorenie reprezentácie dát vhodné pre vstup do neurónovej siete (embedding) vrátane samotného výberu vhodného embeddingu.
4. Návrh architektúry a implementácia neurónovej siete, trénovanie a validácia.
5. Vyhodnotenie výsledkov.

[2] http://ai.stanford.edu/~amaas/data/sentiment/

5. Implementácia

Zadanie sme implementovali pomocou jazyka Python, v prostredí Jupyter notebook-ov. Pre prácu s dátami využívame najmä knižnice Pandas a Numpy.
Pre spracovanie jazyka sme využili kniznicu NLTK. Pre implementovanie neurónových sietí používame Keras.

5.1 Predspracovanie dát

Prvým krokom pre predspracovanie dát je načítanie datasetu. Načítavané dáta boli opísané v kapitole 3. Každý načítaný záznam obsahuje tieto informácie:

- Comment - Text hodnotenia
- Sentiment - Hodnota 0 alebo 1 podľa toho či ide o pozitívny alebo negatívny komentár

Druhým krokom je spracovanie textu komentárov, teda NLP (spracovanie prirodzeného jazyka). Spracovanie jazyka začína upravením textu (odstránením skrátených foriem slov, viacerých medzier zasebou a podobne). Následne zadefinujeme stopwords - slová, ktoré same o sebe nemajú žiadny význam a preto je vhodné ich z komentárov odstrániť. Poslednou časťou spracovania komentárov je prevedenie slov do základnej formy. Na toto sme zatiaľ použili stemmer, pričom neskôr možno použijeme aj lemmatizer (ten pre svoje správnejšie fungovanie využíva aj POS tagging).

5.2 Embedding

Embedding začína vytvorením slovníka (dictionary). Slovník je vytvorený zo slov nachádzajúcich sa v trénovacej časti datasetu a je zoradený podľa počtu výskytov slov. Toto je z dôvodu ak by sme nechceli používať celý slovník ale iba istú časť najčastejšie sa vyskytujúcich slov.

Slovník má veľkost 79709 slov, pričom na indexe 15000 je slovo vyskytujúce sa 9x, a na indexe 35000 už len slovo vyskytujúce sa 2x. Zatiaľ sme obmedzili slovník na 15000 najčastejších slov.

Keďže slovník je vytvorený z trénovacej množiny, v testovacej množine sa môžu nachádzať slová, ktoré v slovníku nie sú - tieto slová sú odstránené.

Dáta vstupujúce do neurónovej siete musia mať rovnaké dĺžky. Túto dĺžku sme nastavili na 100 slov (toto číslo sa ešte môže zmeniť), a dlhšie komentáre sme orezali. Kratšie komentáre sme doplnili nulami - zero padding.

Pri embeddingu máme možnosť natrénovať vlastný (čo má výhodu silného zastúpenia doménových slov) alebo použiť už vytvorený. V projekte plánujeme použiť, resp. natrénovať vlastné embeddingy Fastext resp. Word2vec (a porovnať ich). Toto trénovanie zatiaľ v projekte nie je keďže ide o prvú iteráciu. Je vytvorený iba defaultný embedding implementovaný v rámci balíku keras.

5.3 Neurónová sieť

Architektúra siete je ‘many_to_one’, keďže na vstupe máme väčší počet slov a výstupom je 1 label. Jednotlivé vrstvy sú sekvenčne zoradené za sebou:

Obr. 1 - architektúra základného modelu NN.

Použité vrstvy v neurónovej sieti:

- Embedding vrstva
  - Input_dim o veľkosti vocab-u - v našom prípade 150001
  - Ouput_dim o veľkosti embedding-u - 100
  - Maska nastavená na nuly (kvôli zero paddingu)
- Bidirectional vrstva
  - Aj vstupný aj výstupný layer je lstm s veľkosťou 64
- Dense vrstva
  - Output vrstva
  - Aktivačná funkcia sigmoid
  - Vystupuje z nej 1 hodnota
- Dropout vrstva
  - Pridaná kvôli pretrénovaniu modelu (overfitting)

Tieto parametre a vrstvy môžu byť ešte v ďalšej časti projektu upravené alebo zmenené.

6. Vyhodnotenie

Pri vyhodnocovaní správnosti klasifikácie sentimentu, ktorá je problémom **binárnej klasifikácie**, sme si ako metriku zvolili **presnosť** (accuracy). Loss funkciou je **binary cross-entropy** (logaritmická chyba).

Každý model sme trénovali na datasete 25000 filmových recenzií, z čoho polovica bola pozitívna a polovica negatívna. Testovací dataset bol rovnakej veľkosti a rovnakého zastúpenia tried, takže naše trénovacie a testovacie dáta boli v pomere 50:50.

Náš prvý baseline model dosiahol po 10 epochách trénovania nasledovné skóre:

Train accuracy: 0.9972 Train loss: 0.0082
Valid accuracy: 0.8073 Valid loss: 1.1078

Aj keď to nie je najhoršie skóre (minimálne sme porovnateľne lepší ako náhodný model), vzhľadom na vývoj skóre počas epoch sme presvedčení, že naša sieť sa pretrénováva. Počas jednotlivých epoch trénovacie skóre stúpa, kým validačné klesá.

Druhý baseline model, so snahou aspoň mierne zabrániť pretrénovaniu, dosiahol napokon podobné skóre:

Train accuracy: 1.0000 Train loss: 0.0081
Valid accuracy: 0.8062 Valid loss: 0.8747

Do budúcna chceme určite zvýšiť počet epoch trénovania a bojovať s pretrénovaním siete.

7. Ďalšie kroky

Možnosti, ktoré by sme chceli v ďalšej fáze projektu vyskúšať, sa týkajú viacerých častí projektu.

V rámci prípravy dát a NLP časti by sme chceli jednak obmedziť zoznam anglických stopslov (napríklad tak, aby vo vete zostalo slovo ‘not’, ktoré môže napomôcť pochopeniu významu vety) a zároveň zostaviť aspoň minimálny zoznam doménových stopslov (slovo ako movie alebo film).

Namiesto stemovania vieme taktiež skúsiť lemmatizer, ktorý nám vráti menej drasticky skrátenú verziu slov (v základnom tvare).

V rámci embeddingu slov sme zatiaľ využili jednoduchý prístup zostavenia slovnej zásoby a vytvorenia defaultného embeddingu, ktorý je implementovaný v Keras balíku. Tu je priestor pre použitie už natrénovaných embeddingov, ako napr. Fastext, ktoré môžu obohatiť slovnú zásobu a pomôcť chápať text, alebo priestor pre natrénovanie vlastného pokročilejšieho embeddingu, akým je už spomínaný Fastext alebo Glove. Výsledky sa následne môžu porovnať.

Rovnako je tu priestor pre obohatenie / zmenu aktuálnej architektúry siete a skúšanie rôznych (hyper)parametrov (veľkosť, aktivačné funkcie, optimizer, veľkosť embeddingu, …).