Cucu Ștefan-Cătălin
Grupa 363

Translation Source Dialect Identification
Documentație

1. Introducere

Problema propusă constă în prezicerea dialectului nativ (englez, scoțian, irlandez) al unui text bazat pe traducerea sa în diferite limbaje (daneză, germană, spaniolă, italiană, daneză). Această documentație prezintă rezultatele și statisticile obținute în urma clasificării dialectelor, cât și posibilele relații dintre date.

2. Citirea și procesarea datelor

Datele primite constau în discursuri din Parlamentul European ținute de către europarlamentari din Anglia, Scoția și Irlanda, sub forma unor fisier de tip *.csv*. Am folosit librăria *pandas* pentru citirea datelor sub forma unor *dataframes*.

Statistici referitoare la setul de date se regăsesc în tabelul următor, unde putem remarca că vorbitorii nativi ai Angliei sunt mult mai reprezentați în setul de date comparativ cu cei din Scoția și Irlanda, sugerând că setul nostru de date ar fi profund nebalansat.

	Nr. intrări	% din total	Lungime medie	Deviația standard
Anglia	22700	54.60%	1842.46	1265.00
Scoția	10535	25.34%	1314.30	1165.27
Irlanda	8335	20.05%	1813.12	1143.36

De asemenea, pentru a analiza conținutul intrărilor specifice fiecărui dialect, am extras primele 35 cele mai întâlnite cuvinte în funcție de dialectul nativ al vorbitorilor.

	Primele 35 cele mai întâlnite cuvinte din vocabular
Anglia	Would, European, commission, president, like, parliament, report, union, states, member, EU, people, us, council, one, countries, support, many, need, committee, time, important, work, commissioner, want, Europe, however, new, rights, group, make, could, first, take, way
Scoția	Would, European, report, commission, president, parliament, like, EU, member, union, states, support, countries, council, important, Europe, one, us, people, commissioner, new, need, many, rights, committee, time, however, policy, group, take, hope, welcome, Scotland, therefore, fisheries
Irlanda	European, would, like, union, president, commission, member, states, parliament, people, eu, Ireland, support, report, important, countries, Europe, irish, many, one, council, new, us, need, policy, time, rights, however, work, ensure, commissioner, issue, take, human, political

Observăm că există o suprapunere semnificativă între vocabularul eurodeputaților – domină termeni specifici contextului (ex. „EU”, „European”, „commission”, „Mr. President”, „parliament” etc.) și de interes comun (ex. „human rights”, „policy” etc.). Cu toate acestea, vocabularele eurodeputaților din Scoția și Irlanda se diferențiază prin utilizarea frecventă fie a unor termeni specifici zonelor („Scotland”, „fisheries”, „Ireland”, „irish” etc.). Putem constata astfel că am putea reprezenta cu success datele folosind un model tip *Bag-of-words*.

Datele au fost curățate folosind următorul *regex* pentru a înlătura semne de punctuație și alte semne rămase:

```
sentences = [re.sub("[-.,;:!?\"\'\/()_*=`]", "", sentence).replace('\n', ' ').strip().lower() for sentence in train_df['text'].values]
```

Ulterior acestea au fost tokenizate utilizând librăria *nltk*:

```
def tokenize_text(text, language):
    tokens = []
    for sent in nltk.sent_tokenize(text, language=language):
        for word in nltk.word_tokenize(sent, language=language):
            if len(word) < 2:
                continue
            tokens.append(word.lower())
    return tokens
```

3. Clasificarea datelor

3.1. Strategiile abordate

Datorită naturii multilingvistice a setului de date, se remarcă două strategii posibile pentru construirea modelului: fie construirea unui singur clasificator pentru întreg setul de date, fie construirea unui clasificator pentru fiecare limbă în care au fost traduse textele.

În urma experimentelor, am remarcat că nu există o strategie predominant mai bună – unele modele au o acuratețe mai bună pe întreg setul de date, altele au o acuratețe medie mai bună când împărțim setul de date în funcție de limba în care a fost tradus.

În tabelul următor voi ilustra câteva modele selectate de clasificare și scorurile obținute în funcție de cele 2 strategii.

Model	Strategie	Scor
KNN	Intreg setul	0.44137
LinearSVC	Set de date divizat	0.67712
LogisticRegression	Intreg setul de date	0.69372
LogisticRegression	Set de date divizat	0.66955
MultinomialNB	Intreg setul de date	0.65933

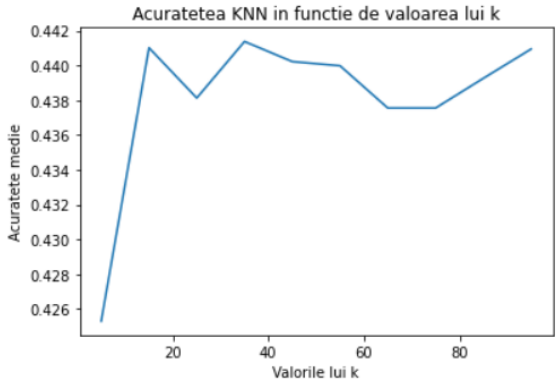
În continuare voi prezenta câte un model pentru fiecare dintre cele 2 strategii:

3.2. Întreg setul de date: KNN

Modelul „celor mai apropiați *k* vecini” (KNN – *K Nearest Neighbors*) este un algoritm neparametrizat de învățare supervizata. Acesta prezice eticheta unui exemplu test ca fiind eticheta predominantă ale celor mai apropiate k exemple de antrenare. Are un singur parametru – *k,* numărul de vecini luați în considerare.

Modelul a fost aplicat asupra reprezentărilor vectoriale ale tuturor textelor, obținute prin *CountVectorizer()*.

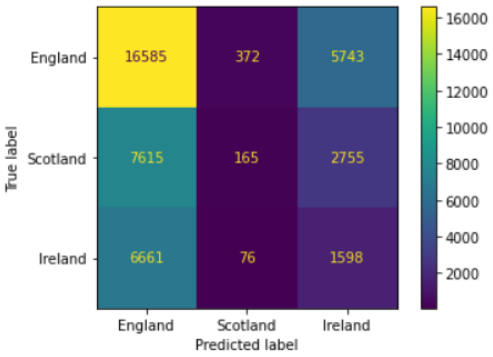
Am ales valoarea pentru hiperparametrul *k* experimental, în figura următoare regăsindu-se acuratețea medie a modelului folosind *StratifiedKFold(n_splits=5)* pentru generarea setului de validare pentru multiple valori ale lui *k*.



Cele mai bune metrici au fost dobândite pentru *k = 35*, acestea fiind observate în următorul tabel:

Fold	Acuratete
0	0.41881164301178736
1	0.45212893913880203
2	0.48099591051238877
3	0.4089487611258119
4	0.44599470772191485
Medie	0.4413759923021409

Matricea de confuzie a modelului este reprezentată în figura următoare:



Se observă motivul pentru care avem o acuratețe foarte joasă – modelul tinde să prezică predominant eticheta *England*, deoarece aceasta predomină în setul de date original.

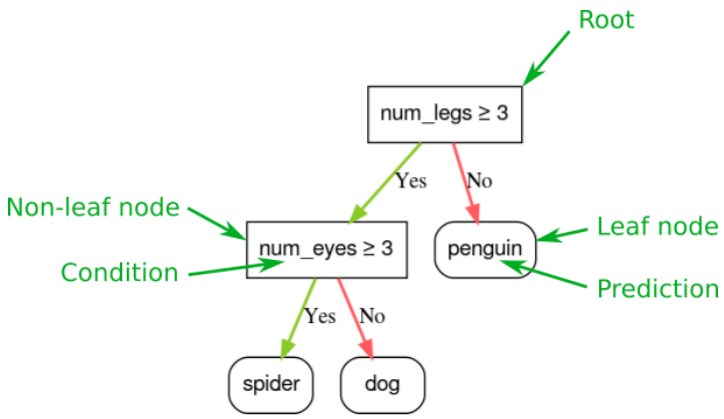
Statisticile referitoare la durata antrenării acestui model pe întreg setul de date sunt:

```
CPU times: user 16.6 ms, sys: 18.9 ms, total: 35.5 ms
Wall time: 34.5 ms
KNeighborsClassifier(n_neighbors=35)
```

3.3. Set de date divizat: Gradient Boosted Decision Trees

Pentru setul de date divizat am folosit algoritmul de *GBDT* din framework-ul open-source *LightGBM (Light Gradient-Boosting Machine)*, dezvoltat de către Microsoft. Acesta se bazează, în primul rând, pe algoritmul de arbore de decizie.

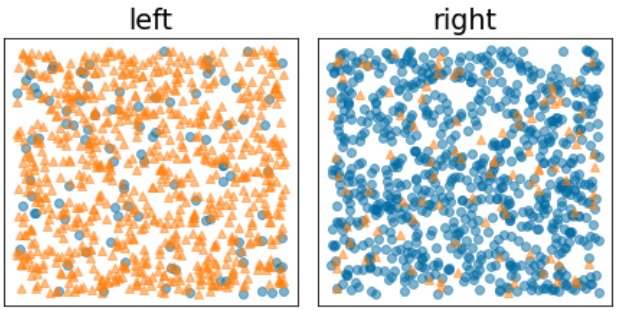
Un arbore de decizie este un model compus dintr-o colecție de condiții organizate ierarhic sub forma unui arbore. Fiecare nod fără frunze conține o condiție, iar fiecare nod cu frunze conține o predicție. Inferența unui model de arbore de decizie este calculată prin direcționarea unui exemplu de la rădăcină (în partea de sus) către unul dintre nodurile frunză (în partea de jos) în funcție de condiții.



Majoritatea algoritmilor utilizați pentru a antrena arbori de decizie funcționează cu o strategie de „divide et impera”. Algoritmul începe prin a crea un singur nod (rădăcina) și crește în mod recursiv arborele de decizie.

La fiecare nod, toate condițiile posibile sunt evaluate și notate. Algoritmul selectează "cea mai bună" condiție, iar apoi se repetă în mod recursiv și independent pentru ambele noduri copii. În cazul în care nu se găsește nicio condiție satisfăcătoare, nodul devine frunză. Predicția frunzei este determinată ca fiind cea mai reprezentativă valoare de etichetă din exemple.

Cel mai utilizat algoritm de divizare pentru task-urile de clasificare se bazează pe entropia *Shannon* și creează condiții de forma $X_i \geq t$. Urmărim astfel să divizăm cât mai bine datele curente în *n* submulțimi.



Gradient boosting este o metodologie aplicată asupra unui alt algoritm de învățare automată. Practic, avem 2 tipuri de modele: unul „slab”, în cazul nostru un arbore de decizie, și unul „puternic”, alcătuit din mai multe modele slabe. La fiecare pas, antrenăm un nou model slab pentru a prezice eroarea actualului model puternic. Modelul slab (adică "eroarea") este apoi adăugat la modelul puternic cu semn negativ pentru a reduce eroarea modelului puternic.

Folosind acest algoritm, am creat câte un model pentru fiecare limbă în care au fost traduse textele, aplicându-l asupra reprezentărilor vectoriale ale textelor din limbile respective, obținute prin *CountVectorizer()*.

Implementarea din *LightGBM* pentru algoritmul de *GBDT* are următorii hiperparametri:

- *num_leaves*: numărul maxim de frunze pentru arbori (default=31)
- *max_depth*: adâncimea maximă a arborelui; <= 0 înseamnă nicio limită (default=-1)
- *learning_rate:* rata de învățare a algoritmului de *boosting* (default=0.1)
- *n_estimators:* nr. de arbori pe care îi antrenăm (default=100)
- *min_split_gain:* reducerea minimă a funcției *loss* necesară pentru a face o partiție suplimentară pe un nod de frunză al arborelui (default=0.)
- *min_child_weight: s*uma minimă a ponderilor instanței necesară la o frunză (default=1e-3)
- *min_child_samples: n*umărul minim de date necesare pentru o frunză (default=20)
- *reg_alpha:* termen de regularizare L1 pe ponderi (default=0.)
- *reg_lambda:* termen de regularizare L2 pe ponderi (default=0.)

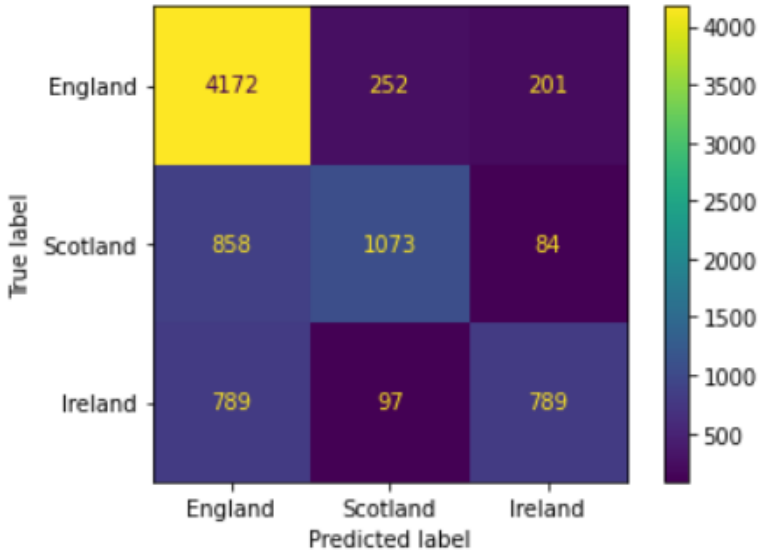
Am ales valoarea pentru hiperparametrilor experimental, în tabel regăsindu-se valoarea acurateții modelului de GBDT pe un set de validare (20% din setul de antrenare) în funcție de hiperparametrii alesi. Menționez că hiperparametrii care nu sunt menționați în tabel au fost lăsați cu valorile lor default.

N_estimators	Min_child_samples	Reg_alpha	Reg_lambda	Acuratete medie
100	20	0	0	0.7208659049909802
200	20	0	0	0.7214672279013831
200	20	0.5	0	0.7226698737221888
200	20	0.5	0.5	0.72291040288635
200	10	0.5	0.5	0.7256764882742032

Astfel, am ales setul de hiperparametrii cu acuratețea medie cea mai mare. Pentru acest model, am obținut următoarele acurateți pentru fiecare limbă in parte:

Limba	Acuratete
español	0.7209861695730607
Deutsch	0.7300060132291041
dansk	0.7330126277811184
italiano	0.724594107035478
Nederlands	0.719783523752255

Matricea de confuzie pe setul de validare este reprezentată în figura următoare:



Se observă că modelul poate prezice foarte bine dialectele din Scoția și Irlanda, dar are dificultăți pentru cel englez.

Statisticile referitoare la durata antrenării acestui model pe întreg setul de date sunt:

```
CPU times: user 2min 58s, sys: 13 s, total: 3min 11s
Wall time: 1min 50s
```

Acest model a obținut o performanță de 0.71338 pe setul public de date (40% din setul total de date) în cadrul competiției de pe Kaggle, echivalent cu locul 7 în cadrul clasamentului final.

4. Concluzie

Identificarea dialectului sursă din traduceri reprezintă o problemă inedită, pentru care încă nu avem o soluție perfectă, dar modelele ilustrate mai sus au o acuratețe suficientă pentru a fi de folos în cazuri reale.

5. Bibliografie

1. KNeighborsClassifier. scikit-learn. [Citat: 21 11 2022.] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.

2. Decision Trees. Google Machine Learning Education. [Citat: 21 11 2022.] https://developers.google.com/machine-learning/decision-forests/decision-trees.

3. Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 3149-3157.