Izvješće o eksperimentima

Eksperimenti su dokumentirani tako da je naziv foldera u koji se spremaju log-ovi (ujedno i naziv modela) sljedeći:

→ D1/D2 - označava na kojem je datasetu rađen (D1-NCBI-disease, D2-BC5CDR)

→ B/E/ftB - označava koji je word embedding korišten (B-bioBERT, E-bioEMLo, ftB-*fine tuning* bioBERT-a)

→ C/_ - označava koristi li se CNN char embedding (C-da, _-ne)

→ L/G - označava vrstu ćelije RNN-a (L-BiLSTM, G-BiGRU)

→ <broj> - broj koji označava konfiguraciju preostalih hiperparametara (svaki - eksperiment u config.log datoteci sadrži sve parametre s kojima je pokrenut)

→ A – A označava da se koristi attention

Primjeri:

D1_B_C_L_1 - označava eksperiment gdje je model treniran nad NCBI-disease datasetu, uz bioBERT embedding, koristeći char CNN i uz LSTM ćeliju, riječ je o prvoj kombinaciji hiperparametara i ne koristi se attention.

D2_E___G_1 - označava eksperiment gdje je model treniran nad BC5CDR datasetu, uz bioELMo embedding, ne koristeći char CNN i uz GRU ćeliju, riječ je o prvoj kombinaciji hiperparametara i NE koristi se attention.

D2_E___G_1_A - označava eksperiment gdje je model treniran nad BC5CDR datasetu, uz bioELMo embedding, ne koristeći char CNN i uz GRU ćeliju, riječ je o prvoj kombinaciji hiperparametara i koristi se attention.

Implementacijski detalji*:*

- Early stopping je postavljen tako da se gleda f1-mjera (strict) na validacijskom skupu podataka.
- Dodan je lr_scheduler – ReduceLROnPlateau s parametrima (`mode='max', factor=0.5, patience=2`) koji isto gleda f1 (strict) na validacijskom skupu podataka i ako se dvije epohe za redom vrijednost ne poveća, smanjuje lr za faktor 0.5. (*Napomena: probleme s ekplodirajućim loss-om koje ovo rješava sam uočila tek nakon implementacije attentiona pa je lr_sceduler prisutan samo u *_A modelima*)
- Za sada je implementiran multi-head attention layer koji na ulazu ima skriveni sloj RNN-a (uzeta je Pytorch implementacija sloja) – 2 rada koja su koristila ovu vrstu attention-a i poslužila kao inspiracija: https://ieeexplore.ieee.org/document/8798611 i https://arxiv.org/pdf/2002.00735. Još je stavljena konkatenacija izlaza

multi-head attention sloja i skrivenog sloja RNN-a (isprobano je sa i bez i malo bolji rezultat je kad se napravi konkatenacija pa je ovako trenutno implementirano) – to su *_A modeli.

- Dodana je *fine tuning* opcija za bioBERT embedding. Dakle, u sklopu modela (BiRNN-CRF) ne uzimaju se samo statične reprezentacije riječi, nego se i one uče (propagira se gradijent i na bioBERT model). Korišteni lr za *fine tuning* BERT-a obično je 2e-5 pa je i tu tako stavljeno – to su *_ftB_* modeli.

Postavke za konfiguraciju 1:
hidden_size: 512
num_layers: 1
dropout: 0.3
lr: 1e-3
batch_size: 32
optimizer: "adam"
epochs: 300
max_length: 256
max_grad_norm: 5.0
early_stopping: 10
cnn_vocab: "abcdefghijklmnopqrstuvwxyz0123456789-,;.!?:'\"/\\|_@#$
%^&*~`+-=<>()[]{}"
cnn_max_word_len: 20
cnn_embedding_dim: 256
feature_size: 256
att_num_of_heads: 16
ft_lr: 2e-5 #ovo je learning rate za *fine tuning* bioBERT-a

U tablici koja slijedi, prikazane su dobivene f1-mjere po skupovima podataka (Napomena: ovi rezultati dobiveni su za jedno pokretanje s istim postavljenim seedom za svaki eksperiment). Za bc5cdr prikazana je odvojena f1-mjera po entitetima (chem-kemikalija, dis-bolest) te mikro usrednjena f1-mjera na cijelom skupu podataka. Također, za oba skupa podataka, prikazan je i defaul i strict izračun. (Napomena: strict način rada osigurava da se predikcije entiteta računaju kao točne isključivo ako potpuno odgovaraju stvarnim granicama entiteta i njegovom tipu).

MODEL	BC5CDR-CHEM	BC5CDR-DIS	BC5CDR	NCBI-DIS	BC5CDR-CHEM (strict)	BC5CDR-DIS (strict)	BC5CDR (strict)	NCBI-DIS (strict)
B_C_G_1	0.84	0.78	0.81	0.76	0.38	0.74	0.64	0.73
B___G_1	0.84	0.76	0.81	0.77	0.44	0.74	0.67	0.73
B___L_1	0.85	0.77	0.81	0.78	0.41	0.74	0.66	0.75
B_C_L_1	0.85	0.78	0.82	0.79	0.46	0.75	0.68	0.75
B___L_1_A	0.9	0.8	0.85	0.82	0.58	0.76	0.73	0.78
B_C_L_1_A	0.89	0.81	0.85	0.82	0.6	0.76	0.73	0.78
B___G_1_A	0.9	0.8	0.86	0.77	0.56	0.76	0.72	0.75
B_C_G_1_A	0.9	0.82	0.86	0.82	0.61	0.78	0.74	0.79
E___G_1	0.93	0.82	0.88	0.78	0.72	0.77	0.76	0.75
E_C_G_1	0.92	0.82	0.88	0.79	0.71	0.77	0.76	0.76
E___L_1	0.93	0.82	0.88	0.8	0.79	0.76	0.77	0.76
E_C_L_1	0.93	0.82	0.88	0.82	0.79	0.77	0.77	0.79
E_C_L_1_A	0.93	0.84	0.89	0.84	0.79	0.78	0.79	0.79
E___G_1_A	0.94	0.84	0.89	0.84	0.8	0.78	0.79	0.79
E___L_1_A	0.94	0.84	0.89	0.84	0.81	0.78	0.79	0.8
E_C_G_1_A	0.94	0.84	0.89	0.85	0.81	0.79	0.79	0.8
ftB___L_1	0.92	0.85	0.89	0.86	0.72	0.8	0.78	0.85
ftB_C_L_1	0.93	0.85	0.89	0.86	0.74	0.8	0.79	0.85
ftB___G_1	0.93	0.85	0.89	0.87	0.72	0.8	0.79	0.86
ftB_C_G_1	0.93	0.85	0.89	0.87	0.72	0.8	0.78	0.86
ftB___L_1_A	0.92	0.85	0.89	0.88	0.71	0.8	0.78	0.86
ftB_C_L_1_A	0.93	0.85	0.89	0.87	0.72	0.79	0.78	0.86
ftB___G_1_A	0.93	0.85	0.89	0.87	0.74	0.8	0.79	0.86
ftB_C_G_1_A	0.93	0.85	0.89	0.88	0.71	0.81	0.79	0.87

Analiza i zaključci:

-> Modeli koji koriste bioELMo imaju bolje performanse od bioBERT-a koji je korišten kao fiksni ekstraktor značajki (možda jer bioBERT nije large model, tu bi se to moglo još isprobati), a najbolji rezultati su dobiveni za modele kod kojih je rađen *fine tuning* bioBERT-a

-> Dodavanje ovako definiranog char CNN embeddinga ne pomaže nešto puno

-> Dodavanje ovako definiranog attention sloja isto ne pomaže puno

-> Strict f1-mjere (očekivano) niže

-> *Fine tuning* puno pomaže u odnosu na „fiksni" BERT na oba dataset-a, dok je u odnosu na ELMo poboljšanje vidljivo na NCBI-disease datasetu i za *strict* mjeru na BC5CDR datasetu kad se uz ELMo ne koristi attention, ostali slučajevi daju slične metrike. Zanimljivo je to da se bioBERT „muči" s chem entitetima u striktnom načinu rada te gledajući samo tu kolonu ELMo daje najbolje rezultate, a kod defaultnog načina rada, vrijednosti f1 za chem kolonu su veće nego za disease kof svih modela.

-> BiLSTM i BiGRU daju uglavnom podjednake rezultate, BiGRU je možda malo bolji u nekim modelima pogotovo ako se uzima u obzir i činjenica da ima manje parametara

-> Trenutno najbolji modeli - ftB_C_G_1_A

Plan za dalje:

-> isprobati još neke druge konfiguracije (mijenjat hidden_size, lr, itd.)

-> (*Pitanje: Jel bi imalo smisla napraviti* fine tuning *bioBERT-a na recimo BC5CDR datasetu (po originalnom paperu ili po ovako nekom tutorijalu koji ima jednostavniji klasifikacijski sloj https://medium.com/@whyamit101/fine-tuning-bert-for-named-entity-recognition-ner-b42bcf55b51d) i onda te težine*

koristiti za „fiksni” bioBERT od kojeg uzimamo samo embedding za BiRNN-CRF model treniranog na drugom datasetu (NCBI-disease)?)
-> Proučiti i implementirati dice loss u sklopu multitask-a (po uzoru na https://aclanthology.org/2021.findings-acl.424.pdf)

Hyperparameter tuning

1. Korak

-> napravljen tuning parametara (*hidden_size, lr, ft_lr, optimizer, dropout*) nad obje baze na modelu ftB_C_G_1_A. Dakle, osim parametara navedenih u zagradi za ostale su korištene vrijednosti iz konfiguracije 1.
Pokrenuto nad n_trials = 70, a vrijednosti raspona su sljedeće:

```
model_args['hidden_size'] = trial.suggest_categorical('hidden_size', [256, 512, 768])
model_args['lr'] = trial.suggest_float('lr', 5e-4, 1.5e-3, log=True) #trenutni lr na 1e-3
model_args['ft_lr'] = trial.suggest_float('ft_lr', 1e-5, 3e-5, log=True) #trenutni ft_lr na 2e-5
model_args['optimizer'] = trial.suggest_categorical("optimizer", ["adam", "adamw"])
model_args['dropout'] = trial.suggest_uniform("dropout", 0.15, 0.45)
```

Dobiveni rezultati:
D1_hyperparam_tuning - *Best Hyperparameters: {'hidden_size': 256, 'lr': 0.0013367211092689295, 'ft_lr': 2.7170546791615555e-05, 'optimizer': 'adamw', 'dropout': 0.31302679090368124}*
D2_hyperparam_tuning - *Best Hyperparameters: {'hidden_size': 512, 'lr': 0.0010195052729147587, 'ft_lr': 2.318100656620731e-05, 'optimizer': 'adamw', 'dropout': 0.2997721353782536} (Napomena: ovdje je na Supeku triggerano zaustavljanje na 68./70. pokušaja jer je dostignut walltime job-a)*

2. Korak

-> hiperparametri dobiveni u koraku 1 postavljeni fiksno i nakon toga pokrenut tuning ostalih (uz n_tirals 100, za sada napravljeno samo na D1):

```
model_args['attention'] = trial.suggest_categorical("attention", [False, True])
if model_args['attention']:
        model_args['att_num_of_heads'] = trial.suggest_categorical("att_num_of_heads", [4,
                                                    8, 16])
model_args['char_cnn_embedding'] = trial.suggest_categorical("char_cnn_embedding",
                                            [False, True])
if model_args['char_cnn_embedding']:
        model_args['char_embedding_dim'] =
        trial.suggest_categorical("char_embedding_dim", [128, 256])
        feature_size = trial.suggest_categorical("feature_size", [128, 256])
model_args['cell'] = trial.suggest_categorical('cell', ['lstm', 'gru'])
model_args['device'] = 'cuda' if torch.cuda.is_available() else 'cpu'
```

Dobiveni rezultati:
D1_hyper_param_tuning - *Best Hyperparameters: {'attention': False, 'char_cnn_embedding': False, 'cell': 'lstm'}*

3. Korak

S dobivenim hiperparametrima u prethodnim koracima napravljeno treniranje i testiranje na 5 različitih seedova te su u tablici prikazane srednje vrijednosti i standardne devijacije (1. redak za hiperparametre nakon koraka 1, 2. redak za hiperparametre iz koraka 1 uz promjenu ćelije iz BiGRU i BiLSTM i 3. redak za hiperparametre nakon koraka 2)

MODEL	BC5CDR	NCBI-DIS	BC5CDR (strict)	NCBI-DIS (strict)
ftB_C_G_3_A_mean	88.87 +/- 0.34	87.19 +/- 0.47	78.51 +/- 0.5	85.33 +/- 0.41
ftB_C_L_3_A_mean		86.86 +/- 0.63		85.16 +/- 1.06
ftB___L_3___mean		87.44 +/- 0.17		85.77 +/- 0.34

<u>Analiza i pitanja:</u>

-> Po trenutno dobivenim rezultatima, char CNN i attention uopće ne doprinose poboljšanju modela tako da je ftB___L_3___mean najbolji trenutni model jer ima i najvišu srednju vijednosti i najmanju std.dev.

-> Ne znam je li ovakav pristup podešavanja hiperparametara u dva koraka dobar, sigurno nije baš korektan jer mijenjanje arhitekture utječe i na ostale hiperparametre (lr, itd.), ali mi se činilo da sve odjednom podešavam da bi to bilo previše hiperparametara za sampliranje pa bi vjerojatno i n_trials trebao biti puno veći da bi rezultati bili donekle mjerodavni... (Da probam pokrenuti s tuning svih odjednom pa stavim n_trials na 120 ili tako nešto?)