AHLT

Name Entity Recognition and Drug-Drug Interaction from Biomedical Texts

Pere Ayats
Olga Valls

Aim of the project

Task 9.1: Recognition and classification of pharmacological substances.

→ Extract Name Entities ⇒ DRUGS

Task 9.2: Detection of Drug-Drug interactions.

→ Four different types of interactions: Advise, Effect, Mechanism and Int

Task 9.1

Recognition and classification of pharmacological substances.

HOW? → Train a model using the feature vectors from the training data

1. Extract features from the sentences of the training set
2. Train a model → Conditional Random Field (CRF)
3. Apply this model to the test data to predict the entities
4. Evaluate the results using the official scorer.

1. Extract features

Features that provide **GOOD results**:

form: the token itself
formlower: the token itself in lowercase
suf3: last 3 characters of the token
suf4: last 4 characters of the token
isUpper: is the token uppercase?
isTitle: is the first character of the token in uppercase?
isDigit: is the last character a digit?
hasSymbol: has the token any of the following characters? \+|-|,|\(|\)|[0-9])
inDron: is the token inside the drug list from the Drug Ontology?

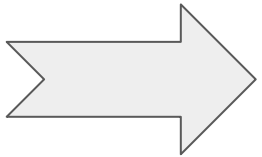
External list of Drugs provided from the Drug Ontology from Bio Portal.

Features that provide **BAD results**:

suf5: last 5 characters of the token
iniCons: does the token starts with a consonant?
iniVowel: does the token starts with a vowel?
has2cons: has the token 2 consonants together?
has3cons: has the token 3 consonants together?
has2vowels: has the token 2 vowels together?
has3vowels: has the token 3 vowels together?
has3Suffix: are the last 3 characters in the suffixes list?
has4Suffix: are the last 4 characters in the suffixes list?
has5Suffix: are the last 5 characters in the suffixes list?
lastUpper: is the last character of the token in uppercase?
lastDigit: is the last character of the token a digit?
postag: postag of the token
lemma: lemma of the token
combinations of **bigrams** and **trigrams**
wordfreq: frequency of the token (also for **bigrams** and **trigrams**)

2. Train a CRF

Features → CRF → Classify



Drug

Brand

Group

Drug_n

```
DDI-DrugBank.d731.s0|96-102|RITUXAN|brand
DDI-DrugBank.d731.s1|68-76|cisplatin|drug
DDI-DrugBank.d592.s0|17-41|Sodium Tetradecyl Sulfate|drug
DDI-DrugBank.d592.s0|111-130|antiovulatory agents|group
DDI-DrugBank.d592.s2|0-6|Heparin|drug
```

3. Predict the entities of the test set

CRF takes context into account whenever the prediction is done

4. Evaluation of the results

form
formlower
suf3
suf4
isUpper
isTitle
isDigit
hasSymbol
inDron

Average Measures

Precision: **0.95**

Recall: **0.59**

F1 score: **0.65**

Task 9.2
Extraction of Drug-Drug interactions

2 approaches used:

- Convolutional Neural Network (CNN)

- Naive Bayes (NB)

Convolutional Neural Networks (CNN)

Process

Sentences XML

For each pair of drugs:

Get content of the sentence ⇨ Name of drugs is not important (DrugX, DrugY)

Create vocabulary set from the training data

Convert the training and the test data into indices from that vocabulary

Create the CNN with the embedding layer

Predict the Drug-Drug interaction from the test set

Architecture of the CNN

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 181, 300)          2046300
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 177, 128)          192128
_________________________________________________________________
global_max_pooling1d_1 (Glob (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               16512
_________________________________________________________________
activation_1 (Activation)    (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 5)                 645
_________________________________________________________________
activation_2 (Activation)    (None, 5)                 0
=================================================================
Total params: 2,255,585
Trainable params: 2,255,585
Non-trainable params: 0
```

Evaluation of the results

Hyper-parameters used:

```
embedding_dims = 300
filters = 128
kernel_size = 5
hidden_dims = 128
batch_size = 64
epochs = 10
optimizer = adam
lr = 0.001
```

Average Measures

Precision: **0.6497**

Recall: **0.4466**

F1 score: **0.5293**

```
DDI-DrugBank.d776.s38|DDI-DrugBank.d776.s38.e3|DDI-DrugBank.d776.s38.e4|1|effect
DDI-DrugBank.d776.s38|DDI-DrugBank.d776.s38.e3|DDI-DrugBank.d776.s38.e5|0|null
DDI-DrugBank.d776.s38|DDI-DrugBank.d776.s38.e4|DDI-DrugBank.d776.s38.e5|0|null
DDI-DrugBank.d776.s39|DDI-DrugBank.d776.s39.e0|DDI-DrugBank.d776.s39.e1|1|mechanism
```

Failed approaches

- Yoon Kim’s Model

- Self-training word embeddings

Naive Bayes (NB)

Process

Sentences XML

For each pair of drugs:

Get content of the sentence Get action related to the interaction

Create word embeddings for the sentences and the verbs (action)

Feed embeddings to the model

Predict the Drug-Drug interaction from the test set

Evaluation of the results

Hyper-parameters used:

`Gaussian Naive Bayes`

Average Measures

Precision: **0.1532**

Recall: **0.5169**

F1 score: **0.2363**

BAD RESULTS!! → Not enough features? Model does not suit the problem?

Conclusions

- Different approaches could be taken for both tasks
- Word Embeddings played a big part in the development
- The main issue was to find a good combination of model and features