MUD Laboratory Projects

During the subject's laboratory sessions, we will explore different approaches to process natural language focusing on two real tasks:

- **Language Detection:** Given a text document or sentence, detect in which language it is written.
- **Drug – Drug Interaction:** Given a sentence including two or more commercial drugs, identify their interaction in a patient. For this task, we will split the work in two sub-tasks: 1. Identify drug names in a sentence and 2. Identify their interaction.

Lab Deliverable 1: Language Detection

The objective of this lab session is becoming familiar with NLP data and the additional challenges you may find compared to tasks based on structured data. To do so, we will focus on the task of Language Detection based on the following tutorial, which provides 21859 sentences written in 22 languages, including several language families and writing scripts:

https://www.kaggle.com/martinkk5575/language-detection/notebook11

In order to guide your work we will propose two exercises using this data:

1) First baseline.

2) Preprocessing and Document Structure.

1. First Baseline

Our first objective is to become familiar with unstructured data and natural language. We start by analyzing an already working system, trying to understand how our decisions affect performance. Over the same naive Bayes classifier and frequency count features system, we will analyze the following two scenarios:

- **Character** level granularity, with a vocabulary of 1000 tokens.
  - python langdetect.py -i dataset.csv -a char -v 1000
- **Word** level granularity, with a vocabulary of 1000 tokens.
  - python langdetect.py -i dataset.csv -a word -v 1000

Using the output metrics and plots, compare both systems, explaining the differences in performance and how the parameters used condition their performances and the kind of errors observed.ta

2. Preprocessing and Document Structure

Using what you learned from the previous exercise, modify the system using word level granularity. All modifications are allowed, some examples to modify are:

- Vocabulary size. Number of tokens (words or characters) that our features represent.
- *preprocess.py:* In this file, you should add all preprocessing steps before computing features from the data. Some suggestions of steps that you can apply are sentence splitting, tokenization, remove punctuation, lemmatization, or any ad hoc step of your choice.
- *classifiers.py: In this file, you should add new classifier models using the frequency features as input.*

Two important details to consider:

- Some of the processes may be language-specific. As this information is our label, we **can not** use this information in our preprocessing steps. All steps must be applied equally to all sentences, or based on features you can extract from the sentences only.
- Some steps may affect the number of sentences, In those cases you would have to implement the required measures to modify the labels accordingly.

Improving performance is **not** the objective of the task. Compare the different models and try to understand how the different steps affects the system's errors and performance.

Lab Deliverables 2-4: Drug-Drug Interaction

The lab project consists in building a system for *SemEval-2013 Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts*.

We will proceed in guided step at each lab session, though you have some freedom to try your own approach at some points.

The DDIExtraction 2013 task concerned the recognition of drugs and extraction of drug-drug interactions that appear in biomedical literature.
Two subtasks were proposed for the challenge:

1) The recognition and classification of drug names.
2) The extraction and classification of their interactions.

Both tasks are independent and evaluated separately (that is, the second task is evaluated on the gold standard drug mentions, not on the output of the first task).

The participants were free to address either one the tasks, or both. However, for the lab project you are required to address **both** of them.

1. Task description and participant systems

The official site of the challenge is https://www.cs.york.ac.uk/semeval-2013/task9.html.

- A description of the results of the challenge can be found in [Segura-Bedmar at al, 2013]
- Also, papers describing each participant system are also available.

All papers can be found at https://aclanthology.coli.uni-saarland.de/events/semeval-2013 or in the `papers` folder provided with the lab project material.

2. Challenge data

The corpus used for the task is the DDI corpus [Herrero-Zazo et al, 2013].
A short description of the DDI corpus provided by *SemEval-2013 Task 9* organizers can be found at https://www.cs.york.ac.uk/semeval-2013/task9/data/uploads/the-corpus-ddi.pdf. and in the `papers` folder.

A copy of DDI corpus is included in the attached material. Please follow the license constraints described in the above paper regarding distribution and use.

3. Project overview

The project will consist of solving both tasks in the challenge. For each task, the following developments will be required

1. **[Lab sessions 2]** NERC Task (Recognition and Classification of drug names)

   Develop a CRF model to solve the task:

   1. Build a feature extractor to encode the data
   2. Train, tune, and test models with the obtained feature vectors

2. **[Lab sessions 3 and 4]** DDI Task (Drug-drug interaction detection and classification)

   Develop a ML-based model to solve the task

   1. Build a feature extractor to encode the data
   2. Train, tune, and test models with the obtained feature vectors

3. **[Lab Sessions 5 to 6]** Neural Network approaches

   Replace the feature-based classifier for each task with a DNN solution, and evaluate the results.

Further details and guidance on each step will be given in the lab sessions.

Check the slides for each lab session to find out what are you expected to deliver at each step of the project.

4. Contents of the Lab package

The package for the Lab project contains the following folders and files

`LabProjectMUD.pdf`	- This file.
`LangDetect/`	- Code and data for Language Detection experiments
`data/`	- Data in csv format.
`source/`	- Code to run the experiments.
`DDI/`	- Code and data for DDI experiments
`data/`	- Folder containing the training and test corpora.
`train/`	- Train corpus
`devel/`	- Development corpus
`test/`	- Test corpus
`resources/`	- Folder containing knowledge extracted from external databases
`papers/`	- Folder containing papers about the task
`Corpus/`	- Folder containing papers about the corpus
`SharedTask/`	- Folder containing papers about the shared task and participant systems approaches.
`Evaluation/`	- Folder containing papers about evaluation metrics, and formats for evaluation scripts.
`util/`	- Folder containing evaluation scripts and other utilities.

References

[Segura-Bedmar et al, 2013] I. Segura-Bedmar, P. Martínez, M. Herrero Zazo. **SemEval-2013 Task 9 : Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013)**. *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pg 341--350, Atlanta, Georgia, USA, 2013.

[Herrero-Zazo et al, 2013] M. Herrero-Zazo, I. Segura-Bedmar, P. Martínez, T. Declerck: **The DDI corpus: An annotated corpus with pharmacological substances and drug-drug interactions**. *Journal of Biomedical Informatics* 46(5): 914-920 (2013)

[Raj et al, 2017] D. Raj, S. K. Sahu, A. Anand. **Learning local and global contexts using a convolutional recurrent network model for relation classification in biomedical text**. *Proceedings of 21th CoNLL 2017*, pages 311-321

[Lim et al, 2018] S. Lim, K. Lee, J. Kang. **Drug drug interaction extraction from the literature using a recursive neural network**. *PLOS ONE* 13(1): e0190926. 2018

[Asada et al, 2017] M. Asada, M. Miwa, Y. Sasaki. **Extracting Drug-Drug Interactions with Attention CNNs**. *Proceedings of BioNLP 2017*, pg 9-18. Vancouver, Canada, 2017

Appendix: Resources

Apart from resources (software and data) suggested in each lab session, you are encouraged to experiment with alternative/additional tools (e.g. to obtain better features, or better performing algorithms)

Some recommended tools and resources:

- **Language processing**
  - *NLTK*: http://www.nltk.org/
  - *FreeLing*: http://nlp.cs.upc.edu/freeling
  - *TextServer: http://textserver.cs.upc.edu/textserver*

- **Machine Learning**
  - *SciPy*: https://www.scipy.org/
  - *scikit-learn*: http://scikit-learn.org/
  - *Keras*: https://keras.io
  - *PyTorch:* https://pytorch.org/
  - *crfsuite:* http://www.chokkan.org/software/crfsuite/
    https://github.com/scrapinghub/python-crfsuite

- **External Knowledge**
  - *DrugBank*: https://www.drugbank.ca/
  - *HSDB*: https://www.nlm.nih.gov/enviro/hsdbcasrn.html

- **Utility tools**
  - *XML.dom*: https://docs.python.org/3.7/library/xml.dom.html

- **Word Embeddings**
  - *Word2vec:* https://code.google.com/archive/p/word2vec/
  - *FastText:* https://fasttext.cc/
  - *Glove:* https://nlp.stanford.edu/projects/glove/