Data Mining - SPEIT/SJTU
Citation Prediction Challenge

École Polytechnique

March 2022

1 Description of the Challenge

The goal of this project is to study and apply machine learning/artificial intelligence techniques to a link prediction problem. Link prediction is the problem of predicting the existence of a link between two entities in a network. The problem has recently attracted a lot of attention in many domains. For instance, in social networks, one may be interested in predicting friendship links among users, while in biology, predicting interactions between genes and proteins is of paramount importance. In this challenge, you will deal with the problem of predicting whether a research paper cites another research paper. More specifically, you are given a citation network consisting of several thousands of research papers, along with their abstracts and their lists of authors. The pipeline that is typically followed to deal with the problem is similar to the one applied in any classification problem; the goal is to use edge information to learn the parameters of a classifier and then to use the classifier to predict whether two nodes are linked by an edge or not.

The challenge is hosted on Kaggle, a platform for predictive modelling on which companies, organizations and researchers post their data, and statisticians and data miners from all over the world compete to produce the best models. The challenge is available at the following link: `https://www.kaggle.com/c/dm-speit-2022`. To participate in the challenge, use the following link: `https://www.kaggle.com/t/b2926120ee09442c96dee0b15e1059d6`.

2 Dataset Description

As mentioned above, you will evaluate your methods on a citation network. The dataset contains papers that have been published at machine learning, artificial intelligence, data mining and natural language processing conferences and journals. You are given the following files (which are available at: `https://www.dropbox.com/sh/fhfjjtk0sr7pmse/AAD4ZEtHv9OI5HfVO22tdMX0a?dl=0`).

1. **edgelist.txt**: a citation network created from papers published at machine learning, artificial intelligence, data mining and natural language processing venues. Nodes correspond to papers, while edges represent citation relationships. The graph is undirected. Therefore, if there is an edge between nodes v and u, then either paper v cites paper u or paper u cites paper v. The graph consists of $138,499$ vertices and $1,091,955$ edges in total. Note that the graph actually contains a larger number of edges, but some of them have been removed and you are asked to identify them in the context of the challenge.

2. **abstracts.txt**: it contains the abstracts of the $138,499$ papers. Each row of the file contains the ID of a paper and its abstract. The following string is used to separate the ID from the abstract: $|--|$.

3. **authors.txt**: this file contains the authors of the $138,499$ papers. Each row of the file contains the ID of a paper and its authors. The following string is used to separate the ID from the authors: $|--|$. For papers that have more than one author, the comma character (,) is used to separate the different authors.

4. **test.txt**: this file contains $106,692$ ordered node pairs. Each row of the file contains the ID of the source node and the ID of the target node. The final evaluation of your methods will be done on these node pairs and the goal will be to predict if there is an edge between the two elements of each pair or not.

3 Evaluation

The performance of your models will be assessed using the logarithmic loss measure, also known as binary cross entropy loss. This metric is defined as the negative log-likelihood of the true class labels given a probabilistic classifier's predictions. Specifically, the binary log loss is defined as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\big(y_i \log(p_i) + (1-y_i)\log(1-p_i)\big)$$

where N is the number of samples (i.e., node pairs), y_i is 1 if there is an edge between the two nodes and 0 otherwise, and p_i is the predicted probability that there is an edge between the two nodes.

4 Provided Source Code

You are given two scripts written in Python that will help you get started with the challenge. The first script (`graph_baseline.py`) uses solely graph-based features with a logistic regression classifier for making predictions. The second script (`text_baseline.py`) uses features extracted from the abstracts of the papers along with a logistic regression classifier. As part of this challenge, you are asked to write your own code and build your own models to predict whether two nodes are linked by an edge. You are advised to use both graph-theoretical and textual information.

5 Useful Python Libraries

In this section, we briefly discuss some tools that can be useful in the challenge and you are encouraged to use.

- A very powerful machine learning library in Python is `scikit-learn`[1]. It can be used in the preprocessing step (e.g., for feature selection) and in the classification task (several regression algorithms have been implemented in `scikit-learn`).

- A very popular deep learning library in Python is `PyTorch`[2]. The library provides a simple and user-friendly interface to build and train deep learning models.

[1] `http://scikit-learn.org/`
[2] `https://pytorch.org/`

- Since you will deal with data represented as a graph, the use of a library for managing and analyzing graphs may be proven important. An example of such a library is the `NetworkX`[3] library of Python that will allow you to create, manipulate and study the structure and several other features of a graph.
- Since you will also deal with textual data, the Natural Language Toolkit (`NLTK`)[4] of Python can also be found useful.
- `Gensim`[5] is a Python library for unsupervised topic modeling and natural language processing, using modern statistical machine learning. The library provides all the necessary tools for learning word and document embeddings.

6 Rules and Details about the Submission of the Project

Rules. The following rules apply to this challenge: (i) one account is allowed per participant (ii) there is a limit in the size of each team (at most 2 members), (iii) privately sharing code outside of teams is not permitted, (iv) there is a limit in the number of submissions per day (at most 5 entries per day), (v) use of external data is not allowed (except from word embeddings). For instance, you are not allowed to use external data to determine if a paper cites another paper or not.

Evaluation and Submission. Your final evaluation for the project will be based on (1) the presentation you will give (**50**%), (2) on your position on the private leaderboard and the log loss that will be achieved (**20**%), and (3) on your total approach to the problem and the quality of the report (**30**%). As part of the project, you have to submit the following:

- A 4-5 pages report, in which you should describe the approach and the methods that you used in the project. Since this is a real classification task, we are interested to know how you dealt with each part of the pipeline, e.g., how you created your representation, which features did you use, which classification algorithms did you use and why, the performance of your methods (log loss and training time), approaches that finally didn't work but are interesting, and in general, whatever you think that is interesting to report.
- A directory with the code of your implementation (not the data, just the code).
- Create a `.zip` file containing the code and the report and submit it to the platform.
- **Deadline: 15/04/2022 23:59 GMT+8**

Presentation: As mentioned above, you will be asked to present the approach you followed. Therefore, you will need to prepare some slides (using ppt or any other tool you like).
Date of presentation: TBA

[3] `http://networkx.github.io/`
[4] `http://www.nltk.org/`
[5] `https://radimrehurek.com/gensim/`