EPL451: Data Mining on the Web – Lab 2

University of Cyprus
Department of
Computer Science

Παύλος Αντωνίου

Γραφείο: Β109, ΘΕΕ01

Project – Choose one of:

1. https://www.kaggle.com/c/house-prices-advanced-regression-techniques
   - Predict the final price of each home
     - Ends in Kaggle: 11:59 pm, Wednesday 1 March 2017 UTC
2. https://www.kaggle.com/c/two-sigma-financial-modeling
   - Predict the value of one financial variable
     - Ends in Kaggle: 11:59 pm, Wednesday 1 March 2017 UTC
3. https://www.kaggle.com/c/two-sigma-connect-rental-listing-inquiries
   - Predict how popular an apartment rental listing is
     - Ends in Kaggle: 11:59 pm, Tuesday 25 April 2017 UTC

- **Project Delivery Day: April 18, 2017**

Recommended steps

- Project 1
  - a set of 79 features given
  - extract “useful” features – dimensionality reduction techniques
  - train a model to predict house price
  - Useful links: http://scikit-learn.org/, https://github.com/GaelVaroquaux/sklearn_europython_2014,
- Project 2
  - a set of more than 100 features given
  - extract “useful” features – dimensionality reduction techniques
  - train a model to predict target value y

	id	timestamp	derived_0	derived_1	derived_2	derived_3	derived_4	fundament	...	echnical_43	technical_44	y
0	10	0	0.370326	-0.006316	0.222831	-0.213030	0.729277	-0.335633		2.0	NaN	-0.011753
1	11	0	0.014765	-0.038064	-0.017425	0.320652	-0.034134	0.004413		2.0	NaN	-0.001240
2	12	0	-0.010622	-0.050577	3.379575	-0.157525	-0.068550	-0.155937		2.0	NaN	-0.020940
3	25	0	NaN	NaN	NaN	NaN	NaN	0.178495		2.0	NaN	-0.015959
4	26	0	0.176693	-0.025284	-0.057680	0.015100	0.180894	0.139445		.0	NaN	-0.007338

Project

- A successful submission of your project will consist of three parts:
  - Source code of your implementation along with instructions of how to compile and run your program.
  - The input data you have used for your experiments in the right input format. For each input file you provide you should also submit the corresponding output file of your program.
  - A description of your project (6 pages, two columns), specifying in detail your goals, approach, milestones, evaluation methodology and experimental results. A document that describes how you did your experiments and what are your obtained results.
- Project presentation (15 min / team + 5 min Q/A)

Task1- N-Gram

- N-Gram είναι η συνεχόμενη ακολουθία Ν όρων από μια δεδομένη ακολουθία κειμένου ή ομιλίας
  - Βρίσκει εφαρμογή στη φωνητική αναγνώριση
- Οι όροι μπορεί να είναι συλλαβές, γράμματα λέξεις κτλ ανάλογα με την εφαρμογή
- Παράδειγμα:
  - Όροι: γράμματα, N = 3
    - Ερώτηση: Βρείτε τα 3-grams που προκύπτουν από την πρόταση "good morning"
    - Απάντηση: "goo", "ood", "od ", "d m", " mo", "mor", … κτλ.
  - Όροι: λέξεις, N = 2
    - Ερώτηση: Βρείτε τα 2-grams που προκύπτουν από την πρόταση "good morning my friend"
    - Απάντηση: "good morning", “morning my", “my friend"

Task1: N-Gram

- Αλλάξτε τον κώδικα του WordCount έτσι ώστε να μετρά πόσες φορές εμφανίζονται διαδοχικά πέντε συνεχόμενες λέξεις (5-Grams).
- Μπορείτε να βρείτε τον κώδικα του WordCount στο

  http://www.cs.ucy.ac.cy/courses/EPL451/labs/LAB02/WordCount.java
- Αν δεν έχετε datasets μπορείτε να τα κατεβάσετε από εδώ

  http://www.cs.ucy.ac.cy/courses/EPL451/labs/LAB02/dataset.zip

Task1: N-Gram

- Η συνάρτηση ***map*** θα παίρνει ως είσοδο (input):
  - key = line offset (δεν μας ενδιαφέρει)
  - value = μια ολόκληρη γραμμή από ένα από τα αρχεία
- Η συνάρτηση ***map*** θα δίνει ως έξοδο (output):
  - key = λίστα με πέντε λέξεις
  - value = 1
- Η συνάρτηση ***reduce*** θα παίρνει ως είσοδο (input):
  - key = λίστα με πέντε λέξεις
  - value = [λίστα με αριθμούς 1]
- Η λίστα θα περιέχει τόσα 1 όσες φορές εμφανίζονται οι πέντε λέξεις στα δεδομένα μας.

Task1: N-Gram

- Η συνάρτηση ***reduce*** στο τέλος θα δίνει ως έξοδο (output):
  - key = πέντε λέξεις
  - value = το άθροισμα των 1 (δηλαδή ό,τι κάνει και το WordCount)

Task2: Anagram

- Ένας αναγραμματισμός είναι ο σχηματισμός λέξης με μετάθεση των γραμμάτων μιας άλλης λέξης
- Π.χ
  - Refills→fillers
  - Relayed→layered
  - Rentals→antlers
  - Rebuild→builder
- Πρέπει να βρείτε τον αναγραμματισμούς σε ένα τεράστιο αρχείο εισόδου
- Dataset
  - http://www.puzzlers.org/pub/wordlists/unixdict.txt

Task2: Anagram

- Κάποια αποτελέσματα της διαδικασίας reduce:

```
2  hasn't,shan't
2  cascara,caracas
2  ramada,armada
2  drawback,backward
2  bacterial,calibrate
2  bandpass,passband
2  aboard,abroad
2  wabash,bashaw
3  banal,laban,nabla
```

APPENDIX

- mapper and reducer classes are declared as inner classes to your application class
- Both inner classes have to be declared **static** in order not to depend on the parent class
- Hadoop uses reflection to create an instance of the class for each map or reduce task that runs
  - **reflection** is a *process of examining or modifying the run time behavior of a class at run time*.
  - job.setMapperClass(Map.class);
    - Map.class is an object that represents the class Map on runtime
- If inner mapper or reduce class declared without the static keyword, the java compile actually creates a constructor which expects an instance of the parent class to be passed in at construction