Lab. Assignment I

(two sessions)

AIW / 2018-2019

Multilingual Text Classification and Information Extraction

OBJECTIVES

The objective of this laboratory assignment is the implementation of a trainable information extraction system for two different application domains and two languages (Spanish and English).

CONSIDERATIONS

Two laboratory sessions (4 hours in total) with your teacher(s) will be used to understand how to carry out the implementation and to solve any doubts. Additional extra time will be required to complete the whole assignment. The first (completed) and the second seminar -- on the GATE system -- are also the basis for this Lab.

During the first session you will start implementing **a text classification system** in **Java** using the Weka machine learning library.

The second session will be dedicated to start the implementation of **information extraction systems** using the GATE libraries, batch machine learning plug-in, and other plug-in(s) available in GATE (e.g. Spanish processing resources). Each group will implement **TWO** information extraction systems. One system should be able to extract information from Spanish texts while the second system should be able to extract information from English texts. Your teacher will explain in detail how to implement the systems with one practical example.

INTRODUCTION

The application domains we will work with are the following: aviation accidents, earthquakes, terrorist attacks, and train accidents. Some examples of texts in each domain and language are provided below:

- **Aviation Accident**:

  2009. June 30. Yemenia Flight 626, an Airbus A310-300 flying from Sana'a, Yemen to Moroni, Comoros, crashes into the Indian Ocean with 153 people aboard; 1 12-year-old is found clinging to the wreckage.

  2009. 30 de junio: el vuelo 626 de Yemenia chocó en cercanías a Comoras, en el Océano Índico.

- **Earthquake:**

  The 1906 Ecuador-Colombia earthquake occurred at 15:36 UTC on January 31, off the coast of Ecuador, near Esmeraldas. The earthquake had a magnitude of 8.8 and triggered a destructive tsunami that caused at least 500 casualties on the coast of Colombia.

  31 de enero de 1906 - Ecuador - Un sismo de 8.8 registrado cerca de la costa de Ecuador y Colombia generó un fuerte tsunami que mató hasta 1000 personas. Se sintió a lo largo de la costa pacífica de América Central hasta San Francisco y tan lejos como el oeste de Japón.

- **Train Accident:**

  January 30, 1993. Ngai Ndethya, Kenya: 65 killed in a Mombasa-bound passenger train carrying 600 passengers which plunged into a river a bridge washaway.

  30 enero 1993. Al menos 150 muertos al descarrilar un ferrocarril cerca de Mombasa (Kenia).

- **Terrorist Attack:**

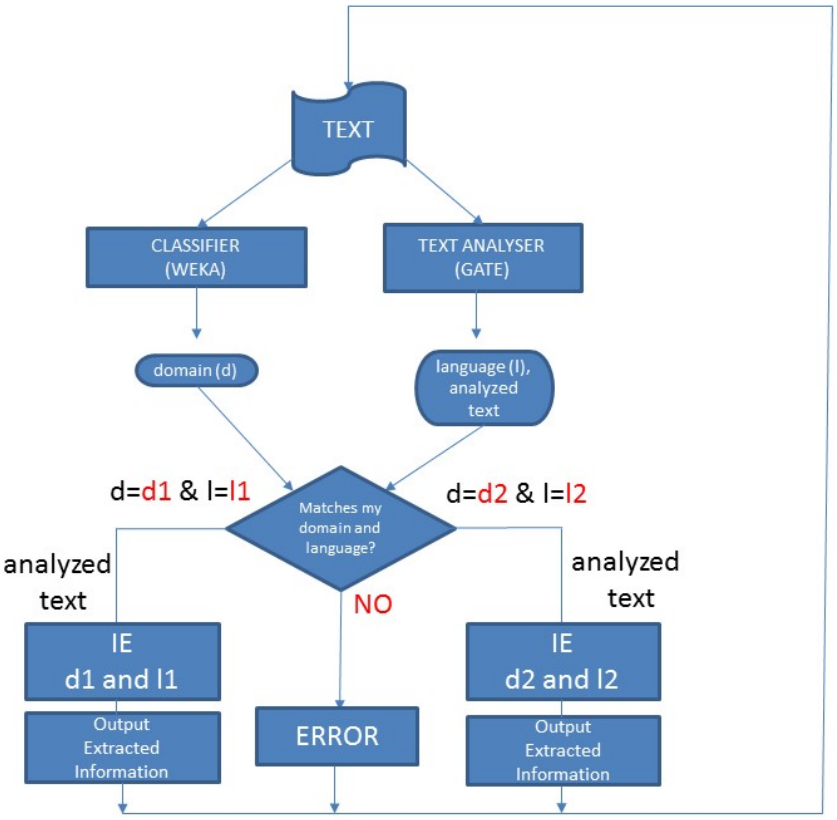
  2008. July 26. India, Ahmedabad. A series of seventeen blasts killing 49 and injuring 160 people.

  26 de julio de 2008: 16 pequeñas bombas explotan en Ahmedabad, en Gujarat, costándole la vida a 45 personas e hiriendo a 161. Un pequeño grupo llamado "Los Mujahideen indios" dicen haber sido los autores.

To reduce the complexity of the assignment, each group will work with **TWO domains** only (each domain in a different language), the information of what two domains you are going to work with will be published in Moodle.

The system to be developed will operate in the following way:

Given a text in one of the two domains you have to deal with, a classification system will decide to which domain the text belongs to. Then, the text will be sent to an appropriate information extraction system in the identified domain and language to extract key information in the domain. A schema of the application to be developed is shown below for the domains and languages *d1 and l1*, *and d2 and l2*:



TRAINING AND APPLYING THE CLASSIFIER

The application to be created will be developed in **JAVA** following **the example code and the indications of your teacher.**

In order to develop a text classification system you will use an algorithm from the **Weka** library (the NaiveBayes classifier) together with texts from the **CONCISUS Corpus** provided by your AIW teachers to train the system (see "PROVIDED MATERIALS" later in this document).

The texts files (*txt) for each of the three domains provided in the CONCISUS Corpus will be used to create an ARFF file suitable for training a text classification system in Weka. **You will have to implement a program** to create the ARFF file which will contain one instance for each of the **txt** documents provided with the CONCISUS Corpus. Each instance of the ARFF file (after the @data keyword) should have the text contained in the txt file and the class the text belongs to (airplane, earthquake, train, attack) as indicated below

```
'2007 July 17 - TAM Airlines Flight 3054, an Airbus A320, crashes at Congonhas-Sao Paulo Airport, Brazil killing all 187 people on board and 12 on the ground.',aiplane

'The 1957 Andreanof  Islands earthquake was a magnitude 8.6 MW (8.3 Ms) megathrust earthquake that took place on March 9, 1957.',earthquake

'June 24, 2002 -  Igandu train disaster, Tanzania : Nearly 300 people die when a passenger train rolls backwards into a goods train.',train

'Sri Lanka, January 31: A LTTE suicide bomber detonates at Central Bank in Colombo, killing 90 and wounding 1,400.', attack

'7 de febrero de 2003. Un atentado atribuido a las Fuerzas Armadas Revolucionarias de Colombia (FARC) contra un club de Bogotá frecuentado por la elite política mata a 36 personas y hiere a otras 150.',attack


'1988 3 de julio: el vuelo 655 de Iran Air fue derribado por la fragata estadounidense ISS Vincennes, matando a sus 290 tripulantes.', airplane
```

It should be clear from the directory structure provided and from the name of the file which is the correct class for each instance.

The header of the ARFF file should have the appropriate format. Make sure the created ARFF file is correct by loading it in WEKA as demonstrated in the first AIW seminar.

Once the classifier has been trained in your JAVA application, you should be able to classify a completely new text in one of the two domains of your application.

In order to test how your algorithm classifies unseen documents (in your java program) please use examples provided in the **testing_classifier.txt** file.

DEVELOPING THE INFORMATION EXTRACTION SYSTEM

The information extraction system will first analyse the document with an specific **application** to produce linguistic annotations (tokens, sentences, POS tags, named entities) and then apply a trained extraction module (**another application**) to the analyzed text to identify in the text key types of information. Finally, the information identified by the system will be used for producing the output. The linguistic analyser will be developed using algorithms from the GATE library (and plugins) and will be explained in full during a seminar.

The information to be identified in the texts are the following:

Domain	**Information to Extract**
Earthquake	Country; Date; Magnitude; Region; Time; Fatalities; Epicentre; Injured
Aviation Accident	Airline; DateOfAccident; FlightNumber; NumberOfVictims; Place; TypeOfAccident; TypeOfAircraft; Year
Train Accident	TypeOfTrain; TypeOfAccident; Place; NumberOfVictims; DateOfAccident; Cause; Survivors
Terrorist Attack	DateOfAttach; Fatalities; Injured; NameOfVictim; Perpetrator; Place; TypeOfAttack

In order to develop each extraction system you will use the CONCISUS annotated corpus which contains texts *already annotated* with the information you need to extract. Each trainable system will be developed with a GATE plug-in which contains a Support Vector Machines learning algorithm. The system will learn from the linguistic information produced by your text analyser and the human annotations in the document. **This will be explained during the second laboratory session.**

FINAL SYSTEM

The final system will integrate the **classifier** and **the information extraction** systems **in one single application** able to perform two tasks: the classification of the text, the analysis of the text (including detection of the language of the text), and correct extraction of the required information from the text.

USER INTERFACE

The user interface we are asking for your application is very simple. You should program **a main class** able to interact with the user as shown in the following example (suppose your application is developed for *train accidents (in English)* and *aviation accidents (in Spanish)*!)

****

Text Classification and Extraction App. This App recognizes texts belonging to the following two domain: earthquakes and train accidents.

Training the Text Classifier.....done!

Loading the Multilingual Text Analyzer….done!

Loading IE System for aviation accidents (Spanish) ..... done!

Loading IE System for train accidents (English) .... done!

READY FOR YOUR TEXT> January 23, 2006 - Bioce train disaster: A passenger train crashes into a ravine near Podgorica, Serbia and Montenegro , killing 46 people and injuring 198 .

YOUR TEXT IS ABOUT TRAIN ACCIDENTS IN ENGLISH

CALLING THE EXTRACTION SYSTEM....

I FOUND THE FOLLOWING USEFUL INFORMATION ABOUT THE TRAIN ACCIDENT:

DATE: January 23, 2006

PLACE: Podgorica, Serbia and Montenegro

VICTIMS: 46

TYPE OF TRAIN: passenger train

.....

READY FOR YOUR TEXT> 2009. 30 de junio: el vuelo 626 de Yemenia chocó en cercanías a Comoras, en el Océano Índico.

YOUR TEXT IS ABOUT AVIATION ACCIDENTS IN SPANISH

CALLING THE EXTRACTION SYSTEM....

I FOUND THE FOLLOWING USEFUL INFORMATION ABOUT THE AVIATION ACCIDENT:

DATE: 30 de junio

YEAR: 2009

FLIGHT NUMBER: vuelo 626

AIRLINE: Yemenia

PLACE: Comoras

.....

READY FOR YOUR TEXT> QUIT

GOOD BYE!

****

Note: In case the text can not be classified in one of the two domains and language you are considering, then you can output "Sorry, I can't process the text".

PROVIDED MATERIALS

All necessary material for the development will be provided in Moodle including JAVA examples for implementing your application.

To obtain the dataset for training the systems please request the Concisus corpus from http://www.taln.upf.edu/pages/concisus/index.html (download area). The material **you'll need is the file AIW_DATA_2018.zip** provided in the main

distribution. For the first session you'll need the files under directory **text_files** for the second session you'll need the files under directory **annotated_files.**

PRESENTING YOUR WORK TO AN AUDIENCE

During the second laboratory session you will have the opportunity to publicly explain to your colleagues the status of your project. This will give your teacher the change to mark your work so far.

MATERIALS TO SUBMIT:

You are required to submit a NetBeans Java project and a package .jar that can be run on different machines from the command line (java ... –jar "...your_jar.jar"). Do not use absolute paths (C:\…) in your code, instead, use relative paths that start from the classpath of your project (i.e. the folder of your project). If you want to use absolute paths you need to use a configuration file and explain how to use it in the report (we will show how to export a project to jar in the first seminar).

You are also required to submit a **well written report** (in English) in pdf format of the work carried out. The maximum length of your report is 8 pages (minimum 4 pages). The report should **clearly state the involvement of each of the members of the group** (what each person has done in the project), should also have an **introduction, description of the dataset used, description of the software, implementation of the different programs, explanation of how to run the system, and a conclusion**. Include all necessary **bibliography and web sites** (in appropriate format) you have consulted to carry out the work. The report should clearly indicate the names and IDs of the members of the team.

EVALUATION:

1. Text Classification implementation and program to create the ARFF training file + correct arff file (20 points)
2. Information Extraction systems implementation + annotated data produced for training the IE systems (40 points)
3. Integration of 1. and 2. in the program (10 points)
4. Output produced (15)
5. Report: (15 points)

DEADLINE TO SUBMIT YOU ASSIGNMENT:

*** 5th February 2019 23:55 *** through the link provided in Moodle

NOTE: A penalty of 10 points per each 24-hours delay applies

REFERENCES

Francesco Barbieri, Francesco Ronzano, Horacio Saggion: Summarization and Information Extraction in your Tablet. Procesamiento del Lenguaje Natural 55: 203-206 (2015).

Horacio Saggion, Sandra Szasz: The CONCISUS Corpus of Event Summaries. LREC 2012: 2031-2037.

H. Witten, Eibe Frank, Mark A. Hall Data Mining: Practical Machine Learning Tools and Techniques, Third Edition (The Morgan Kaufmann Series in Data Management Systems) 3rd Edition. 2011.

Cunningham et al. Developing Language Processing Components with GATE Version 8 (a User Guide)