## Web Mining and Retrieval course

# KeLP tutorial

This folder contains examples showing how to implement a Kernel-based classifier for the Question Classification task, by adopting **KeLP** [Filice et al, 2015], i.e., the Kernel-based Learning Platform developed in the Semantic Analytics Group of the University of Roma Tor Vergata.

This folder contains the following files and folders:

- `README.md` : this readme;
- `WmIRQuestionClassificationExample.java` : the java source code showing how to implement a question classifier; it shows how to load a dataset, instantiate a new kernel and a SVM learning algorithm, how to learn a classifier and to evaluate it on the test dataset;
- `WmIRQuestionClassificationExampleFromJson.java` : the java source code showing how to load the question classifier from a JSON file; it shows how to load a dataset, read the kernel function and the learning algorithm from a JSON file, learn a classifier and evaluate it on the test dataset;
- `lib/` : this folder contains a JAR file with the complete set of KeLP functionalities required to compile and run the java examples (the version 2.0.2 of KeLP is used);
- `json/` : this folder contains the definition in JSON of a SVM learning algorithm enforced with different kernel functions, in particular:
    - `ova_lin.klp` : a linear kernel applied to a sparse vector reflecting a Bag-of-Word model derived from a question;
    - `ova_poly.klp` : a polynomial kernel (with degree set to 2) applied to a sparse vector reflecting a Bag-of-Word model derived from a question;
    - `ova_tk.klp` : a tree kernel function (proposed in [Vishwanathan and Smola, 2003]) applied to a tree derived from the dependency parse tree of a question, as proposed in [Croce et al, 2011];
    - `ova_comb.klp` : the kernel function resulting from the sum of a linear kernel with a tree kernel (the ones described above);
    - `ova_comb-norm.klp` : the same kernel of the previous combination, but former kernels are normalized before the sum;
- `qc_data/` : this folder contains the train and test dataset for question classification provided by [Li and Roth, 2002], converted to a valid format for KeLP.

---

## Requirements

The following examples require the JAVA JDK 6 installed on the machine.

## How to compile the examples

From the main folder, launch the following commands:

```
javac -cp lib/kelp-full2.0.2.jar WmIRQuestionClassificationExample.java
javac -cp lib/kelp-full2.0.2.jar WmIRQuestionClassificationExampleFromJson.java
```

## How to run the examples

This tutorial shows how to build a Kernel based SVM classifier with KeLP. The following JAVA examples show how to :

- read a training and a test dataset;
- instantiate a kernel function (from the Java code or reading it from a JSON file);
- learn a multi-class classifier from annotated examples (according to a One-VS-All schema);
- evaluate the performance of the classifier over the test set by measuring the accuracy, i.e., the percentage of test questions associated to their real question class.

### The `WmIRQuestionClassificationExample.java` example

This example shows how to implement algorithms and kernel functions directly through JAVA code. Once it has been compiled, execute the following command:

```
java -cp .:./lib/kelp-full2.0.2.jar WmIRQuestionClassificationExample training_set
_path test_set_path kernel[lin| poly | tk | comb | comb-norm] c_svm
```

where:

- `training_set_path` : is the path of the file containing the training dataset (e.g. `qc_data/qc_train.klp` )
- `test_set_path` : is the path of the file containing the test dataset (e.g. `qc_data/qc_test.klp` )
- `kernel` : a string indicating one of the implemented kernel functions; in particular:
    - `lin` : a linear kernel applied to a sparse vector reflecting a Bag-of-Word model derived from a question;
    - `poly` : a polynomial kernel (with degree set to 2) applied to a sparse vector reflecting a Bag-of-Word model derived from a question;
    - `tk` : a tree kernel function (proposed in [Vishwanathan and Smola, 2003]) applied to a tree derived from the dependency parse tree derived for the sentence, as proposed in [Croce et al,

    2011];
    - `comb` : the kernel function resulting from the sum of a linear kernel with a tree kernel (the ones described above);
    - `comb-norm` : the same kernel function of the previous combination, but former kernels are normalized before the sum;

- `c_svm` : the regularization parameter used within the SVM learning algorithm.

When executed, the program will save a SVM model with the Support Vectors.

### The `WmIRQuestionClassificationExampleFromJson.java` example

This example shows how to implement a classifier by reading the learning algorithm and kernel functions from a JSON files. Once it has been compiled, execute the following command:

```
java -cp .:./lib/kelp-full2.0.2.jar WmIRQuestionClassificationExampleFromJson trai
ning_set_path test_set_path json_file_path
```

where:

- `training_set_path` : is the path of the file containing the training dataset (e.g. `qc_data/qc_train.klp` )
- `test_set_path` : is the path of the file containing the test dataset (e.g. `qc_data/qc_test.klp` )
- `json_file_path` : the path of the JSON file describing the learning algorithm and kernel. Please, refer to the set of JSON files contained in the `json/` folder.

Please, refer to the presentation **svm*kelp*practice** for more details about the code and the input data.

---

# References

[Li and Roth, 2002] Xin Li, Dan Roth, Learning Question Classifiers. COLING'02, Aug., 2002.

[Vishwanathan and Smola, 2003] S.V.N. Vishwanathan and A.J. Smola. *Fast kernels on strings and trees*. In Proceedings of Neural Information Processing Systems, 2003.

[Croce et al, 2011] Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2011. *Structured lexical similarity via convolution kernels on dependency trees*. In EMNLP, Edinburgh.

[Filice et al, 2015] Simone Filice, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2015. *Kelp: a kernel-based learning platform for natural language processing*. In Proceedings of ACL: System Demonstrations, Beijing, China, July.

---

# How to cite KeLP

If you find KeLP usefull in your researches, please cite the following paper:

```
@InProceedings{filice-EtAl:2015:ACL-IJCNLP-2015-System-Demonstrations,
    author = {Filice, Simone and Castellucci, Giuseppe and Croce, Danilo and Basili,
    title = {KeLP: a Kernel-based Learning Platform for Natural Language Processing}
    booktitle = {Proceedings of ACL-IJCNLP 2015 System Demonstrations},
    month = {July},
    year = {2015},
    address = {Beijing, China},
    publisher = {Association for Computational Linguistics and The Asian Federation
    pages = {19--24},
    url = {http://www.aclweb.org/anthology/P15-4004}
}
```

# Useful Links

KeLP site: http://sag.art.uniroma2.it/demo-software/kelp/

SAG site: http://sag.art.uniroma2.it

Source code hosted at GitHub: https://github.com/SAG-KeLP