UNIVERSITÀ DI PISA

MSc Artificial Intelligence and Data Engineering

Data Mining and Machine Learning

TEXT MINING

Design and implement a framework in python

Alberto CARUSO, Pietro CALABRESE

Sommario

Introduzione

The framework is based on a python notebook to better handle the development process and to make the debug phase quicker. The aim is to develop a service that could be embedded in a web application, that allows to quickly categorize political documents, in particular the ones that belong from Q&A sessions of the various parliamentary istitutions.

In order to get our aim, we used a supervised machine learning model for automatic text classification. This experiment is carried out comparing 12 different models, based on two different types of text representation, Bag Of Word and Word Embeddings.

As the lecterature suggests, we applied classical learning classification algorithms to BOW text representation, whereas the word embeddings ara used in combination with deep learning models.

Global Parameters

Dataset Parameters

In this section the parameters to set dataset information are provided by the user

dataset_path : in this variable, he specifies the path of the document that will be categorized

sheet_name : because of the notebook is implemented to handle .xls files, the variable specifies the name of the sheet that have to be processed

text : specifies the name of the attribute in the file, that is used as input text

review : specifies the name of the attribute in the file, that contains the class for each input text

language : set the language of the input text

Experimental Parameters

n_fold : set the number of fold that the cross-validation performs.

Number_of_feature : this is present only in notebooks that use BOW representation, and It sets the max number of features that the algorithm will use in the vocabulary.

language_w : this is present only in notebooks that use WE representation. It sets the path of the word embedding vector of FastText.

Bag Of Word

Support Vector Machine

A support vector machine constructs a set of separating hyper-planes in the inputs' vector space, which can be used for classification among other tasks

Parameters

C : is used to set the amount of regularization. The C parameter trades off correct classification of training examples against maximization of the decision function's margin. For larger values of C, a smaller margin will be accepted if the decision function is better at classifying all training points correctly. A lower C will encourage a larger margin, therefore a simpler decision function, at the cost of training accuracy.

γ : the gamma parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'. The gamma parameters can be seen as the inverse of the radius of influence of samples selected by the model as support vectors.

kernel : specifies the kernel type to be used in the algorithm. It must be one of 'linear', 'poly', 'rbf', 'sigmoid'.

Complement Naïve Bayes

It is a modified version of the multinomial naïve bayes classifier for texts, addressing word dependencies assumption and unbalanced classes.

Parameters

α : like that found in multinomial naïve bayes, the smoothing priors $\alpha \geq 0$ accounts for features not present in the learning samples and prevents zero probabilities in further computations. Setting $\alpha=1$ is called Laplace smoothing, while $\alpha<1$ is called Lidstone smoothing.

Passive Aggressive Classifier

An online machine learning algorithm building a set of separating hyper-planes in the inputs' vector space based on a single input at a time.

Parameters

C : is used to set the amount of regularization, defines the maximum step size.

tol : is used to define the stopping criterion, the iterations will stop when (loss > previous_loss - tol).

Multi-Layer Percepton

An artificial neural network with a single hidden layer that learns a non-linear function approximator for classification.

Parameters

hidden_layer_size : the i-th element represents the number of neurons in the ith hidden layer.

activation : define the activation function for the hidden layer.

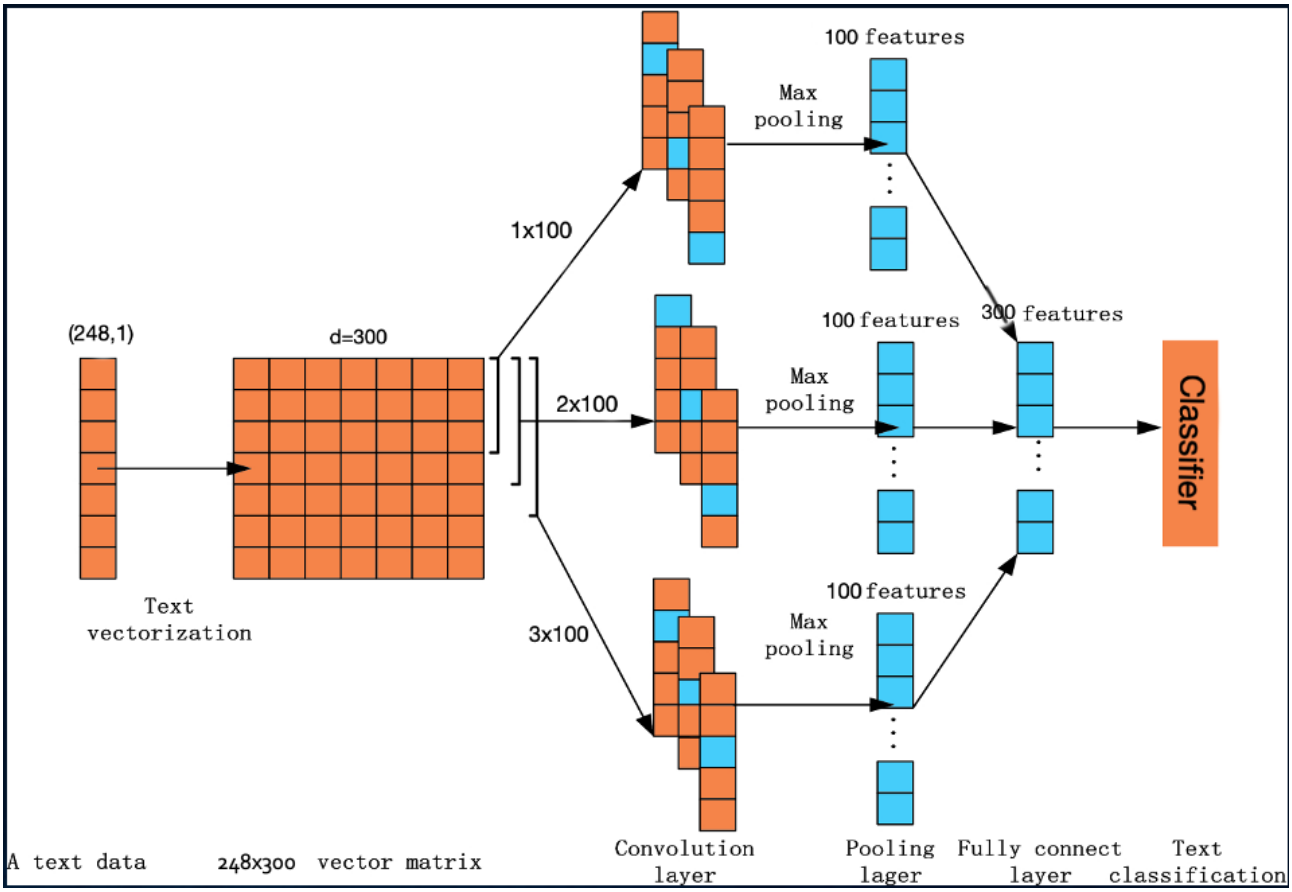
optimizer : is used to set the solver for weigth optimization.

α : is a regularization term, it sets the L2 penalty parameter. L2 regularization reduces overfitting by allowing some training samples to be misclassified.

learning_rate : define schedule for weight updates.

Word Embedding

Convolutional Neural Network



A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of a series of convolutional layers that convolve with a multiplication or other dot product.

The neural network was defined through an architecture implemented with the use of a one-dimensional convolution, as it deals with sentences. Then three filters of dimension n are applied, and then passed to a neural network which carried out the classification.

Parameters

Batch_size : define the number of sentences inside the batch.

Nubmer_of_filters : define the number times in which the filter is applied to the input matrix.

Filter_size : define the size of each region of filter.

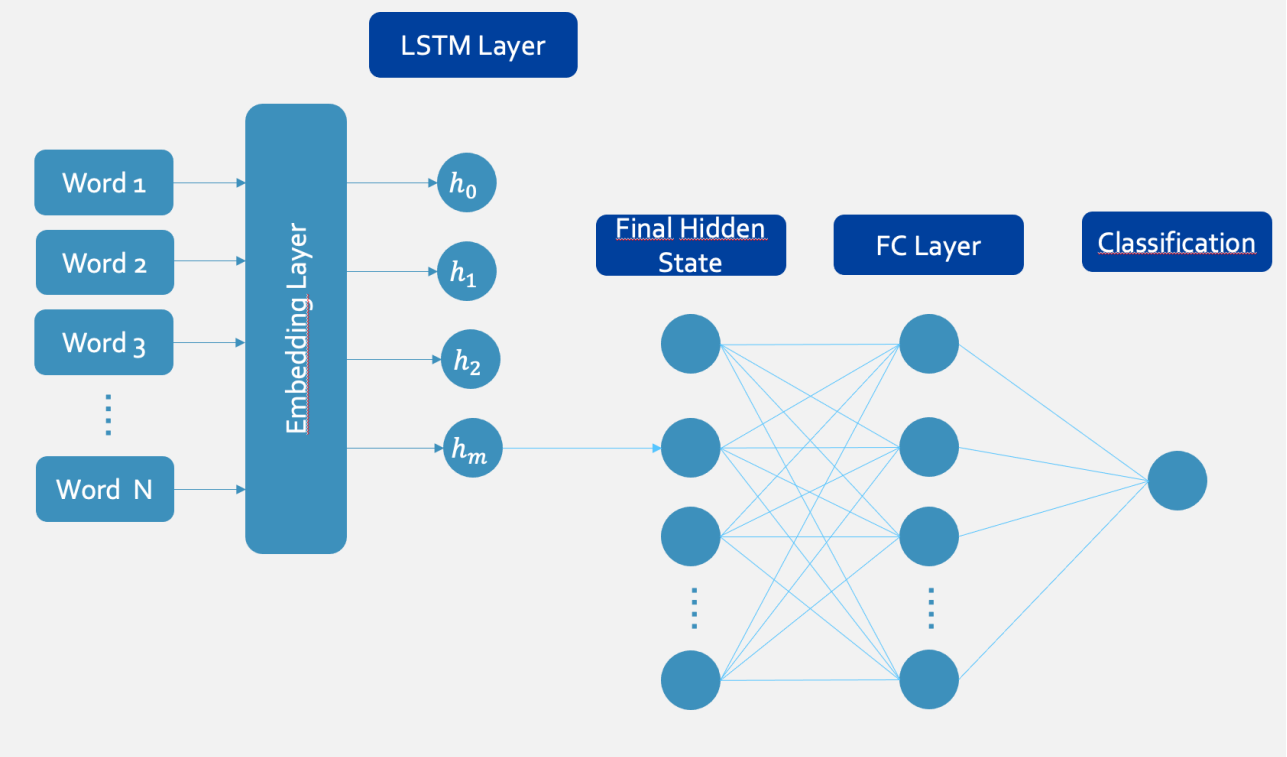
Dropout_pc : defines the rate at which the dropout layer is applied

activation : define the activation function for the hidden layer.

optimizer : is used to set the solver for weigth optimization.

Long Short-Term Memory

It is an artificial recurrent neural network with multiple hidden layers and feedback connections that can process sequences of data, such as speech or text.



The network is designed to have the number of neurons equal to the size of the max length of the senteces. each word will be processed by a hidden state and the output of the last state will be sent to a full connected layer which will give the input to the classifier.

Parameters

num_layers : define the number of layer that compose the LSTM network

hidden_dim : define the number of hidden neuron for each layer

activation : define the activation function for the hidden layer.

optimizer : is used to set the solver for weigth optimization.