supper&supper

NLP – Understanding scientific language

Project goal

There are two different types of gene mutations, one that contributes to the growth of cancer tumors and the other that contributes to neutral gene mutations. Currently, this classification of gene mutations is done manually. This is a very time-consuming task where a clinical pathologist has to manually check and classify each gene mutation based on findings from text-based clinical literature.

This use case describes the development of an algorithm to automatically distinguish these two different types of mutations using Natural Language Processing (NLP) to analyze scientific articles. This approach can save the clinical pathologist a lot of time and energy. By knowing the mutation class, the physician can accurately diagnose the disease or better treat the patient.

Provided data

A training data set of 3321 samples and a testing data set of 368 samples were available. The training data contains complete information including the full-text scientific publication, responding gene name and the mutation class determined by oncologists. The genes are divided into 9 different classes. The test data has the same structure as the training data, but misses the mutation class. The mutation class from test data is predicted by the algorithm and evaluated as the performance of the model.

Challenges

The data is very unbalanced, with classes 3, 8 and 9 representing 5% and classes 7 and 4 representing over 50% of the overall data. The amount of training samples is relative small for a 9-class NLP-classification. The use of pre-trained models might be limited due to the high specificity for scientific literature. The texts themselves have a variable extent, ranging from only 80 to over 70.000 words in length. Bringing the texts to a uniform size might lead to loss of important information.

Applied methods

Before feeding the texts into a model, a variety of pre-processing steps are necessary. The texts are converted into a list of so called “tokens”, meaning single words. Unwanted tokens, such as stop words or words that occur only once, are filtered out. Next, a step called “word embedding” is performed - the conversion from tokens to numbers. Three different methods were used, namely *Bag of Words, TF-IDF* and *Word2Vec.* We compared two Word2Vec models, a pre-trained one by Google and a self- trained model from scratch.

For the following classification, three classical machine-learning algorithms were employed and further compared: logistic regression, random forest classification, as well as support vector classification.

Additionally we used BERT with a fine-tune approach for word embedding and modelling. BERT is a novel framework of Google and broke many previous benchmarks, making it the current state-of-the-art in natural language processing. It can be used for a variety of tasks and delivers consistently good results on all of them.

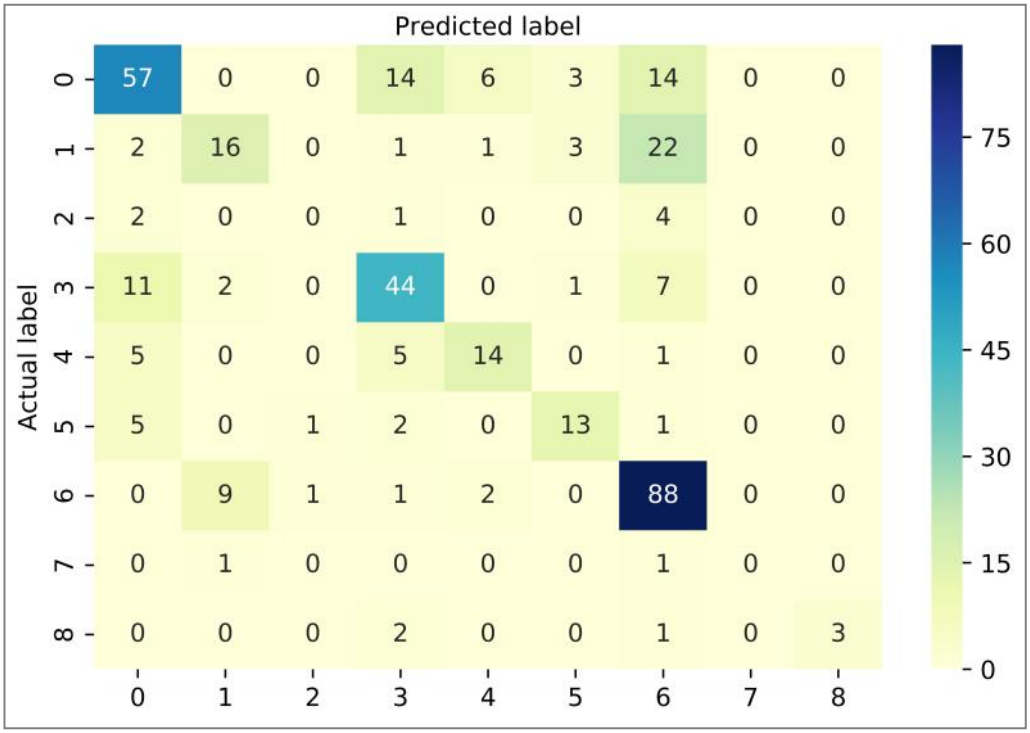
Project outcome

For evaluation we used the Multi-Class Log Loss (MCLL) of the prediction probability. The lower the MCLL, the better performance is. The baseline was created with random probabilities of the class. Out of the three used models, random forest classification was the best among all methods. Out of the used word embedding techniques the self-trained *Word2Vec* continually delivers the best results. The combination of self-trained W2V and random forest model provides the best performance. With this combination a precision of 63,5% was reached (MCLL 0.94).

Surprisingly, pretrained models of BERT couldn't improve the results from the methods above. Possible explanations are the need for more data or a general bad performance of pre-trained models on a very specific subject, such as scientific literature. Potential for further improvement lies in the generation of more training data and hyperparameter optimization. Due to the immense runtime, no support vector classification was performed for *Bag of Words*.

	Bag of Words	TF-IDF	Pre-trained W2V	Self-trained W2V	BERT	Baseline
Logistic Regression	1.08	1.03	1.40	0.99	1.13	2.54
Random Forest Classifier	1.02	1.01	0.98	**0.94**		
Support Vector Classifier	—	1.01	1.22	0.97		

Multi-Class Log Loss of the tested models and embedding techniques (lower = better)



Confusion matrix of the classification with random forest and data from self-trained *Word2Vec*

The outcome of this project are the predictions of genetic mutation classes based on scientific papers. If a gene mutation with associated variation is considered for a patient, the potential mutation class for this mutation can be predicted or at least narrowed down with the help of associated scientific articles. The oncologist or pathologist can furthermore use this information for the diagnose. Like this the cure probability can be increased by giving a timely treatment through an early disease detection.