Final Project

Text Classification using Zero-Shot Learning Technique

MIS-64061

Advanced Machine Learning

Murali Shanker

mshanker@kent.edu

Kent State University, U.S.A

Submitted By

RamaKrishna Mullapudi

rmullapu@kent.edu

811173251

Objective:

To perform Text Classification using Zero-Shot Learning Technique and multi-Class Classification using Zero-Shot Learning Technique.

Abstract:

In general Machine Learning learns patterns and features about the training classes and then when it is given a test sample it labels it with a class it is most similar to the features learnt by our algorithm or model. Most machine learning methods focus on classifying instances whose classes have already been present in the training set. Moreover, many applications require classifying instances whose classes have not seen before. Zero-shot Learning (ZSL) is about leveraging supervised learning with no additional data or any other training data set.

Approach:

The approach that I used to perform the text classification is Zero-Shot Learning (ZSL).

In ZSL the machine learning model (machine) can describe to which class an unlabeled sample belongs to when it does not fall into the category of any of the trained categories, which can be described as zero-shots for the data points.

ZSL is referred to a specific type of task in which a classifier is trained on one set of labels and then evaluated on different set of labels that the classifier has never seen before. In simple words, Zero-Shot model allows us to classify data, which wasn't used to build to a model. ZSL refers to a specific use case of machine Learning where you want the model tom classify data based on very few or even no labelled example.

Working:

Whenever the ZSL model is trained on a task or text it takes in two sequences and determines whether they contradict each other or support each other. ZSL first maps instances to intermediate attributes, which can be seen classes, human-specified or data-dependent attributes. Then the predicted attributes are mapped to a large number of unseen classes.

The text classification comes under the Natural Language Processing which has effective methods of learning from the enormous amounts of unlabeled data available on the internet. The text classification works based on the Natural Language Interference (NLI).

The NLI considers two sentences: a “Premise” and a “Hypothesis”. The task is to determine whether the hypothesis is true or false when given the premise. You can see that in the below example.

Premise	Label	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction	The man is sleeping.
An older and younger man smiling.	neutral	Two men are smiling and laughing at the cats playing on the floor.
A soccer game with multiple males playing.	entailment	Some men are playing a sport.

In the model we feed both the premise and the hypothesis through the model together as distinct segments, then the model generates a score depending on the relationship between the premise and hypothesis. By default, the model assumes that one the hypothesis is true and gives scores which add up to 1.

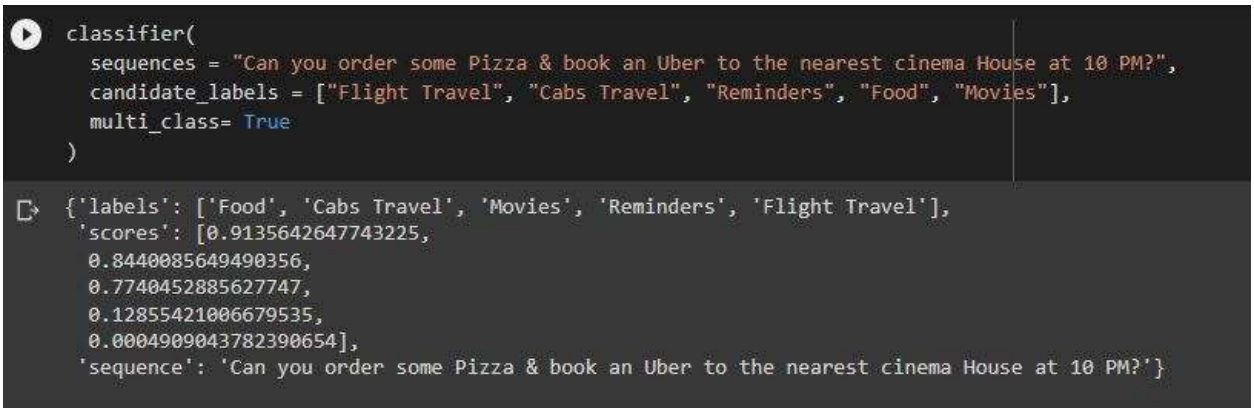
```
premise = "who won the cricket match today?"
hypothesis = ["climate", "sports", "economics"]

classifier(premise, hypothesis)
```

```
{'labels': ['sports', 'climate', 'economics'],
 'scores': [0.9922264218330383, 0.00483263423666358, 0.0029409478884190 32],
 'sequence': 'who won the cricket match today?'}
```

In the above example, we can see the test that we want to classify is given in the premise whereas the labels to which we want to classify the text into is given in the hypothesis.

The classifier compares all the labels present in the hypothesis and scores them that are closely related to the premise.

Multiclass Classification:



We can achieve Multi-Class Classification using ZSL technique, to do a multiclass classification we just have to pass the command "**multi_class = True"** as shown in the above picture. In multi class classification the scores will be independent, but each score will be between **0** and **1**.

Advatages:

In Zero-shot pipelining we can define our own labels and run the classifier to assign a probability to each label. We can run the classifier on the model until we are satisfied with the result.

We can Multi-class classification and sentiment detection using Zero-shot pipelining technique.

The ZSL technique can be applied in both annotated data and non-annotated data.

The ZSL technique can reduce training and help systems be accurate when confronted with unexpected data.

Conclusion:

The Zero-Shot learning method is related to human vision in many ways. Instead of performing classification or recognition on a limited set of objects or data, using Zero-Shot Learning it is possible to classify or recognize every object. Unsupervised text classification with zero-shot model allows us to solve text sentiment detection tasks when you don't have training data to train the model. For specialized use cases, when text is based on specific words or terms it is better to go with a supervised classification model, based on the training set. As models continue to grow, we see a very slow decrease in the reliance on large amounts of annotated data for downstream tasks.

In Supervised learning, there is no understanding of wisdom. This can be easily proven by stating that we can easily identify a new object by having its description or using similarities of previously learned objects without needing data for those new objects.