Document classification is an example of Machine Learning (ML) in the form of Natural Language Processing (NLP). By classifying text, we are aiming to assign one or more classes or categories to a document, making it easier to manage and sort.

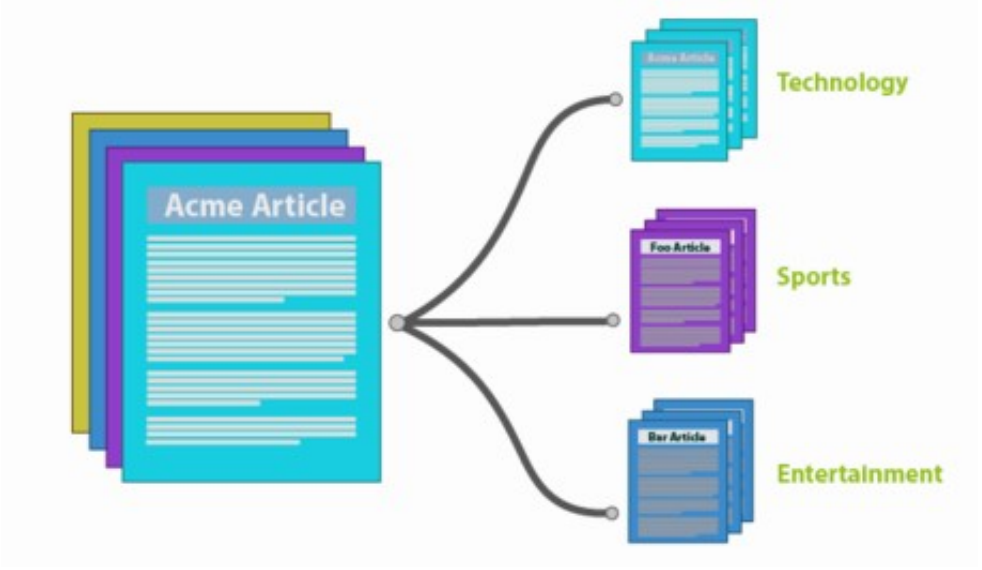
By Parsa Ghaffari.

Introduction

Document classification is an example of Machine Learning (ML) in the form of Natural Language Processing (NLP). By classifying text, we are aiming to assign one or more classes or categories to a document, making it easier to manage and sort. This is especially useful for publishers, news sites, blogs or anyone who deals with a lot of content.

Broadly speaking, there are two classes of ML techniques: supervised and unsupervised. In supervised methods, a model is created based on a training set. Categories are predefined and documents within the training dataset are manually tagged with one or more category labels. A classifier is then trained on the dataset which means it can predict a new document's category from then on.

Depending on the classification algorithm or strategy used, the classifier might also provide a confidence measure to indicate how confident it is that the classification label is correct.



A simple Illustration of Document Classification

We can use the words within a document as "features" to help us predict the classification of a document. For example, we could have three very short, trivial documents in our training set as shown below:

Reference Document Class 1	Reference Document Class 2	Reference Document Class 3
Some tigers live in the zoo	Green is a color	Go to New York city

To classify these documents, we would start by taking all of the words in the three documents in our training set and creating a table or vector from these words.

(some,tigers,live,in,the,zoo,green,is,a,color,go,to,new,york,city) class

Then for each of the training documents, we would create a vector by assigning

a 1 if the word exists in the training document and a 0 if it doesn't, tagging the document with the appropriate class as follows:

some	tigers	live	in	the	zoo	green	is	a	color	go	to	new	york	city	
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	**class 1**
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	**class 2**
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	**class 3**

When a new untagged document arrives for classification and it contains the words "Orange is a color" we would create a word vector for it by marking the words which exist in our classification vector.

some	tigers	live	in	the	zoo	green	is	a	color	go	to	new	york	city	
0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	**unknown class**

If we compare this vector for the document of unknown class, to the vectors representing our three document classes, we can see that it most closely resembles the vector for class 2 documents, as shown below:

< 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0 > Unknown class

< 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0 > class 1 (6 matching terms)
< 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0 > class 2 (14 matching terms - winner!!)
< 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1 > class 3 (7 matching terms)

It is then possible to label the new document as a class 2 document with an adequate degree of confidence. This is a very simple but common example of a statistical Natural Language Processing method.

A more detailed look at real world document classification

A real world classifier has three components to it, which we can look at in a bit more detail.

1. The dataset

The quality of the tagged dataset is by far the most important component of a statistical NLP classifier. The dataset needs to be large enough to have an adequate number of documents in each class. For 500 possible document categories, you may require 100 documents per category so a total of 50,000 documents may be required.

The dataset also needs to be of a high enough quality in terms of how distinct the documents in the different categories are from each other to allow clear delineation between the categories.

2. Preprocessing

In our simple examples, we have given equal importance to each and every word when creating document vectors. We could do some preprocessing and decide to give different weighting to words based on their importance to the document in question. A common methodology used to do this is TF-IDF (term

frequency - inverse document frequency). The TF-IDF weighting for a word increases with the number of times the word appears in the document but decreases based on how frequently the word appears in the entire document set.

3. Classification Algorithm and Strategy
In our example above, we classified the document by comparing the number of matching terms in the document vectors. In the real world numerous more complex algorithms exist for classification such as Support Vector Machines (SVMs), Naive Bayes and Decision Trees.

Additionally, we placed our document into just one category type but it's possible to assign multiple category types and even multiple labels within a given category type. For example, using IPTC International Subject News Codes to assign labels, we may give a document, two labels simultaneously such as "sports event - World Cup" and "sport - soccer". Where "sports event" and "sport" are the root category with "World Cup" and "soccer" being the child categories.

We may also have a hierarchical structure in our taxonomy and require the classifier to take that into account.

Summary
In supervised methods of document classification, a classifier is trained on a manually tagged dataset of documents. The classifier can then predict any new document's category and can also provide a confidence indicator. The biggest factor affecting the quality of these predictions is the quality of the training data set.

Bio: Parsa Ghaffari is CEO and Founder of AYLIEN, a startup that focuses on creating simple and intelligent applications in the news and media space. AYLIEN Text Analysis API is designed to help developers, data scientists, business people and academics extract meaning from text. You can try out the API by signing up for an *account* or visiting *sandbox*.

Related:

- More Data Mining with Weka
- Data Mining and Text Analytics of World Cup 2014
- Overcoming Text Analytics Barriers