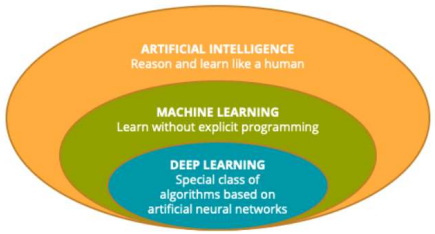
Applications of ML

Thursday, 13 August, 2020 12:09

As the diagram shows, all deep learning algorithms are particular cases of machine learning algorithms—but it's **not true** that all machine learning algorithms are deep learning algorithms.



Classical Machine Learning & Deep Learning Compared

ML trains an equation $y=mx+b$ Deep learning trains a graph

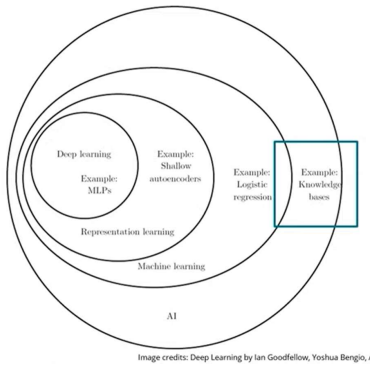


Image credits: Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville

What are the benefits of Deep Learning?

The non-parametric approach taken by neural nets allows them to learn arbitrarily complex functions

Hence, they can learn complex patterns without explicitly seeing them

Effective across various "incarnations" of data (numbers, images, text)

Work on (very) large sets of data

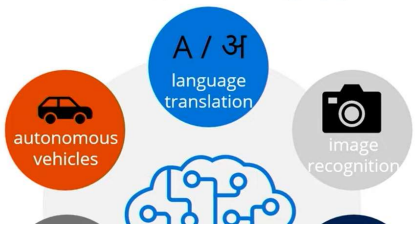
Can be distributed for parallel training

Lately, can learn time-related patterns (RNN = Recurrent Neural Network)

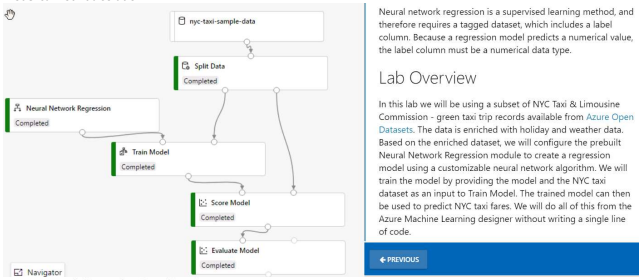
Capable of reaching on-par performance with certain human activities

Some of the area where Deep Learning are apply:

Where is Deep Learning applied?



Any class of statistical models can be termed a neural network if they use adaptive weights and can approximate non-linear functions of their inputs. Thus **neural network regression** is suited to problems where a more traditional regression model cannot fit a solution.



Score Model result visualization

Rows 3,520 Columns 15

nowDepth	precipTime	precipDepth	temperature	totalAmount	Scored Labels
6	0	-1.95	12.8	11.54942	
6	0	24.638136	9.8	10.709536	
24	51	21.695868	8.3	9.367613	
1	0	6.591071	12.8	14.810899	
.571429	24	9999	7.844094	11.3	13.343941

Statistics

Mean	14.8097
Median	10.1969
Min	7.2651
Max	98.9439
Standard deviation	9.593
Unique values	3516
Missing values	0
Feature type	Numeric Score

Evaluate Model result visualization

Rows 1 Columns 5

Mean_Absolute_Error	Root_Mean_Squared_Error	Relative_Squared_Error	Relative
2.628156	4.095484	0.158281	0.36335

Coefficient_of_Determination

Statistics

Mean	0.8417
Median	0.8417
Min	0.8417
Max	0.8417
Standard deviation	-
Unique values	1
Missing values	0

*If coefficient of Determination is closer to one then better the model is.

Similarity Learning: -

Introducing similarity learning

- Closely related to classification and regression
- Uses a different type of objective function
- Often used in recommendation systems
- Often used in solving verification problems (speech, face, and the like).

Similarity learning as a supervised learning approach

Similarity learning as a classification problem

The similarity function maps pairs of entities to a finite number of similarity levels (ranging from 0/1 to any number of levels).

Similarity learning as a regression problem

The similarity function maps pairs of entities to numerical values.

A variation of this approach where the supervision form is weakened from an exact measure to an ordering measure is known as **ranking similarity learning**. This approach is a better fit for real-life large-scale problems.

Suppose a streaming service uses similarity learning to compare movies and generate a continuous numerical value (e.g., 0.84) indicating how similar the movies are. This would be an example of similarity learning as *Regression*

Recommender systems - approaches

Content-based	Collaborative filtering
Makes use of features for both users and items	Uses only identifiers for users and items
User properties: age, gender, region etc.	Gets information from a matrix of ratings
Item properties: author, manufacturer etc.	Ratings can be explicit or implicit

https://github.com/sidooms/MovieTweetings - This dataset is the result of research conducted by [Simon Dooms]

Supervised feature learning

New features are learned using data that has already been labeled

Examples:

- Data sets that have multiple categorical features with high cardinality
- Image classification

Unsupervised feature learning

Based on learning the new features without having labeled input data

Clustering = a form of feature learning

Other algorithms:

- Principal component analysis (PCA)
- Independent component analysis
- Autoencoder (deep learning)
- Matrix factorization

Applications of **Feature learning** includes image classification and image search. **Feature embedding** is a categorical features with high cardinality.

Image classification with CNNs

CNN = Convolutional Neural Network

Step 1 – learn local patterns

Step 2 – down-sample and make translation invariant

Step 3 – densely connect all outputs to learn classification

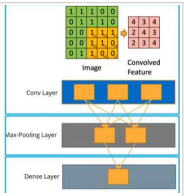


Image search with autoencoders

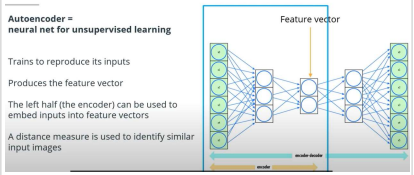
Autoencoder = neural net for unsupervised learning

Trains to reproduce its inputs

Produces the feature vector

The left half (the encoder) can be used to embed inputs into feature vectors

A distance measure is used to identify similar input images



Anomaly Detection

Datasets often contain a small number of items that deviate significantly from the norm. These anomalies can be of interest, since they may be the result of bad data, unusual behavior, or important exceptions to the typical trends. Anomaly detection is a machine learning technique concerned with finding these data points. Introducing anomaly detection;

- Given a set of entities, train a model to detect anomalies
- Usually, the number of abnormal entities << normal entities

This characteristic makes anomaly detection difficult

Supervised anomaly detection

Binary classification problem where entities must be classified as either **normal** or **anomaly**.

The **anomaly** and **normal** classes are highly imbalanced

Based on using a training data set that has already been labeled as normal/anomaly.

For e.g, in fraud detection, we have to provide data that already been identified as fraud transaction to train data in model creation

Unsupervised anomaly detection

A problem of identifying two major groups (clusters) of entities - the **normal** ones and the **abnormal** ones (the anomalies).

Again, the **anomaly** and **normal** classes are highly imbalanced

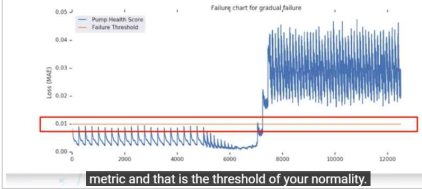
Based on using a training data set that has no normal/anomaly labels available.

For e.g., data provided has no knowledge/label that it is fraud data. Create model such that it can detect normal and abnormal patterns

Example: -

Anomaly detection in machinery maintenance

Train an autoencoder to recognize 'normality'



Mean Average Error metric help to set failure threshold (highlighted in Orange in picture). If value breach failure threshold value then it treated as abnormal.

Forecasting: -

Example of a forecasting problem would be: Given a set of ordered data points (such as sales data over a series of dates), we want to predict the next data points in the series (such as what sales will look like next week). Forecasting is a class of problems that deals with predictions in the context of orderable datasets. These orderable datasets can be time-series datasets, but they don't have to be—forecasting can be applied to other types of orderable sets as well.

Types of forecasting algorithms

ARIMA – AutoRegressive Integrated Moving Average

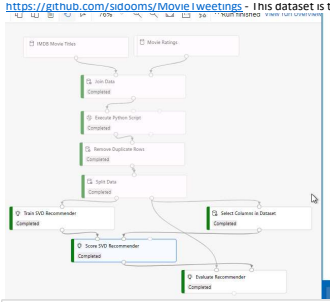
Multi-variate regression

Prophet

ForecastTCN

RNNs (Recurrent Neural Networks)

https://github.com/sidooms/MovieTweetings - This dataset is the result of research conducted by [Simon Dooms]

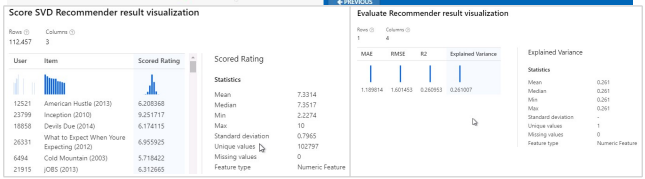


The SVD recommender module in Azure Machine Learning designer is based on the Singular Value Decomposition algorithm. It uses identifiers of the users and the items, and a matrix of ratings given by the users to the items. It's a typical example of collaborative recommender.

Lab Overview

In this lab, we make use of the Train SVD Recommender module available in Azure Machine Learning designer (preview), to train a movie recommender engine. We use the collaborative filtering approach: the model learns from a collection of ratings made by users on a subset of a catalog of movies. Two open datasets available in Azure Machine Learning designer are used the `IMDB Movie Titles` dataset joined on the movie identifier with the `Movie Ratings` dataset. The Movie Ratings data consists of approximately 225,000 ratings for 15,742 movies by 26,770 users, extracted from Twitter using techniques described in the original paper by Dooms, De Pessemier and Martens. The paper and data can be found on GitHub.

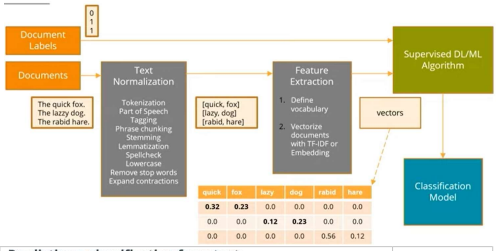
We will both train the engine and score new data, to demonstrate the different modes in which a recommender can be used and evaluated. The trained model will predict what rating a user will give to unseen movies, so we'll be able to recommend movies that the user is most likely to enjoy. We will do all of this from the Azure Machine Learning designer without writing a single line of code.



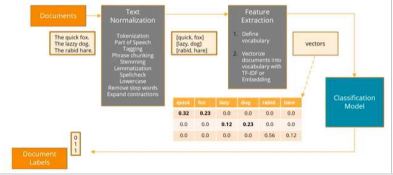
Text Classification: -

In text classification, the text first needs to be translated into some kind of numerical representation—a process known as **text embedding (word embedding and scoring[importance of word in text)**. The resulting numerical representation, which is usually in the form of vectors, can then be used as an input to a wide range of classification algorithms.

Training a classification model with text

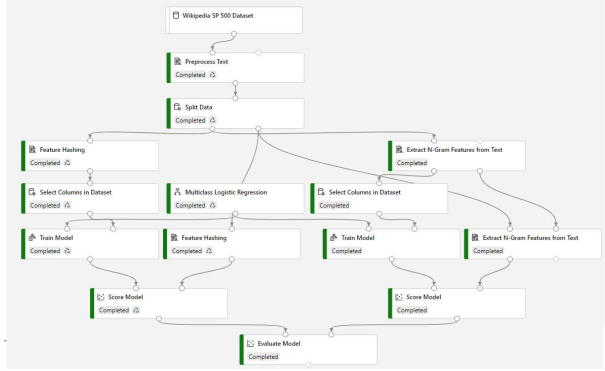


Predicting a classification from text



High level steps:
- Normalize Text
- Futurize text
- Convert to number

*Use same vocabulary for training and test data



Train a simple text classifier

In text classification scenarios, the goal is to assign a piece of text, such as a document, a news article, a search query, an email, a tweet, support tickets, customer feedback, user product review, to predefined classes or categories. Some examples of text classification applications are: categorizing newspaper articles into topics, organizing web pages into hierarchical categories, spam email filtering, sentiment analysis, predicting user intent from search queries, support tickets routing, and customer feedback analysis.

Lab Overview

In this lab we demonstrate how to use text analytics modules available in Azure Machine Learning designer (preview) to build a simple text classification pipeline. We will create a training pipeline and initialize a **multiclass logistic regression classifier** to predict the company category with Wikipedia SP 500 dataset derived from Wikipedia. The dataset manages articles of each S&P 500 company. Before uploading to Azure Machine Learning designer, the dataset was processed as follows: extracted text content for each specific company, removed wiki formatting, removed non-alphanumeric characters, converted all text to lowercase, known company categories added. Articles could not be found for some companies, so that's why the number of records is less than 500.

Exercise 1: Create New Training Pipeline

Task 1: Open Pipeline Authoring Editor

1. In Azure portal, open the available machine learning workspace.

Score Model result visualization

Rows 232 Columns 1,039

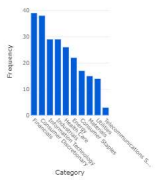
Title	Category	Text	Preprocessed Text	Preprocessed Text_HashingFeature.0	Preprocessed Text_HashingFeature.1	Preproc Text_Ha
Dollar General	Consumer Discretionary	br s p 500 s p 500 component foundation 1939 location city goodlettsville tennessee location country united states u s location locations 10 000 operating income united states dollar 1 5 billion small fy 2011 small in 40 u s states dollar general stores tend to locate	br s p ||| s p component foundation location city goodlettsville tennessee location country unite state ||| u s location location ||| operating income united state dollar small fy small u s state dollar general store tend locate small community walmart chairman cal turner	0.018703	0.009352	0
MeadWestvaco	Materials	br s p 500 s p 500 component fate predecessor successor foundation founder defunct location city location country location richmond virginia locations area served key people john a luke jr chairman official chairman chief executive officer ceo br james a br s p 500 s p 500 component fate predecessor successor foundation founder defunct location city	br s p ||| s p component fate predecessor successor foundation founder defunct location city location country location richmond virginia location area serve key people john luke jr chairman official chairman chief executive officer ceo br jam buzzard president br e mark br s p ||| s p component fate predecessor successor foundation founder defunct location city houston texas	0	0.051169	0.03837

Category

Statistics

Mean	-
Median	-
Min	-
Max	-
Standard deviation	-
Unique values	10
Missing values	0
Feature type	String Label

Visualizations



Evaluate Model result visualization

Rows 2 Columns 5

Overall_Accuracy	Micro_Precision	Macro_Precision	Micro_Recall	Macro_Recall
0.612069	0.612069	0.563826	0.612069	0.521984
0.711207	0.711207	0.754309	0.711207	0.645702

Overall_Accuracy

Statistics

Mean	0.6616
Median	0.6616
Min	0.6121
Max	0.7112
Standard deviation	0.0701
Unique values	2
Missing values	0
Feature type	Numeric Feature

Visualizations

Feature Hashing: transform a stream of English text into a set of integer features. You can then pass this hashed feature set to a machine learning algorithm to train a text analytics model. You must use Feature hashing on Test data also as shown in pipeline diagram above.
Preprocess Text module to clean and simplify text. This task will create *Preprocessed Text* column.
***Extract N-Gram** used on train data is set Vocabulary mode as "Create" and "Read-only" for test data so that it uses same vocabulary as it was created or used on train data.

https://docs.microsoft.com/en-us/azure/machine-learning/algorithm-module-reference/extract-n-gram-features-from-text