COMP47490: Machine Learning

Introduction

Deepak Ajwani

Slides largely prepared by:
Derek Greene, Aonghus Lawlor

School of Computer Science

Autumn 2019

Overview

- COMP47490
  - Practical Details
  - Module Outline
- Machine Learning
  - Common Applications
  - Supervised v Unsupervised Learning
  - Representing Data as Features

Practical Details

UCD CS Machine Learning module offerings in 2018/19:

Module	Coordinator	Semester	Credits
COMP47490 Machine Learning (Face-to-face)	Dr. Deepak Ajwani deepak.ajwani@ucd.ie	1	5
COMP47460 Machine Learning (Blended Delivery)	Dr. Aonghus Lawlor aonghus.lawlor@ucd.ie	1	5
COMP47750 Machine Learning w/ Python	Prof. Pádraig Cunningham padraig.cunningham@ucd.ie	1	5
COMP47590 Advanced Machine Learning	Dr. Brian Mac Namee brian.macnamee@ucd.ie	2	5
COMP47650 Deep Learning	Dr. Guenole Silvestre guenole.silvestre@ucd.ie	2	5

Practical Details

Lectures/Tutorials: Tuesdays 9am (B004 CS), Thursdays 2pm (B004 CS). Bring a laptop to all tutorials - not an iPad etc.

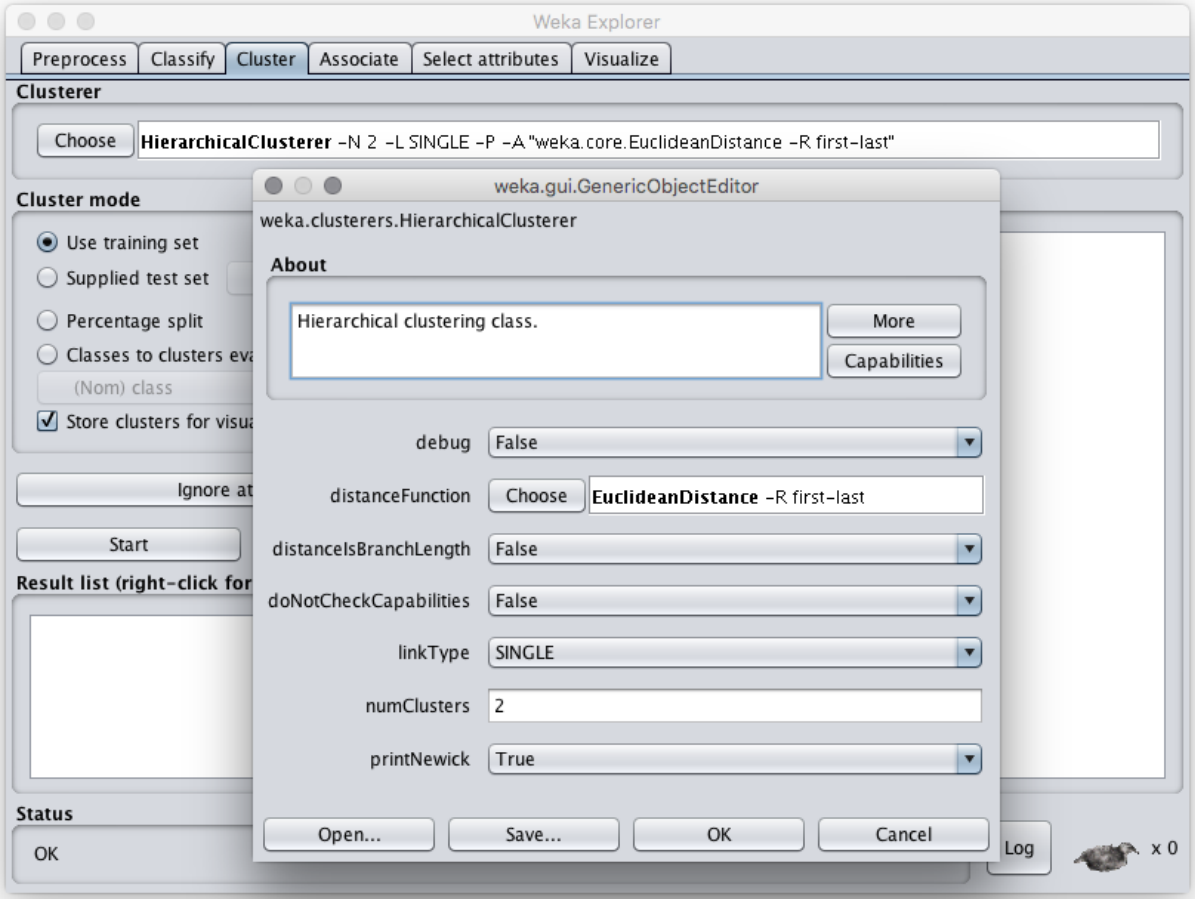
Notes, assignments, and additional material will be available on the Brightspace page for COMP47490

(https://brightspace.ucd.ie/d2l/home/55391)

All of you should be enrolled automatically.

Practical Details

Tutorials require laptop with Java *Weka Toolkit* - Version 3.8 Stable



http://www.cs.waikato.ac.nz/ml/weka

Practical Details

Module marks are based on assignments + final theory exam:

20%	Assignment 1: Weka Practical + Theory Questions
20%	Assignment 2: Weka Practical + Theory Questions
60%	End of Semester Theory Exam

Late Submissions Policy:

! All assignment deadlines are hard deadlines.

1-5 days late: 10% deduction from overall mark

6-10 days late: 20% deduction from overall mark

Not accepted after 10 without extenuating circumstances form or medical certificate.

Practical Details

CS grading scheme applies for this module. Pass mark is 40%.

Grade	Min	Max
A+	95	100
A	90	95
A-	85	90
B+	80	85
B	75	80
B-	70	75
C+	65	70
C	60	65
C-	55	60
D+	50	55
D	45	50
D-	40	45

Grade	Min	Max
E+	35	40
E	30	35
E-	25	30
F+	20	25
F	15	20
F-	10	15
G+	8	10
G	5	8
G-	2	5
NG	0	0

https://www.cs.ucd.ie/Grading

Plagiarism Policy

- Plagiarism is a **serious academic offence**
  - [Student Code, sections 6.2 & 6.3] or [UCD Registry Plagiarism Policy] or [CS Plagiarism policy and procedures]
- Our staff and demonstrators are **proactive** in looking for possible plagiarism in all submitted work.
- Suspected plagiarism is reported to the CS Plagiarism Subcommittee for investigation:
  - Usually includes an interview with student(s) involved
  - 1st offence: **usually** 0 or NG in the affected components
  - 2nd offence: referred to **University Disciplinary Committee**
- Students who enable plagiarism are equally responsible. See:

  http://www.ucd.ie/students/guide/academicregs.html

  http://libguides.ucd.ie/academicintegrity

Extenuating Circumstances

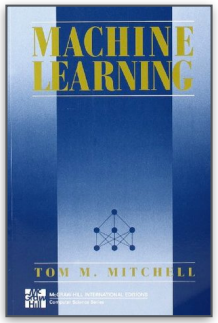
- Applications for Extenuating Circumstances
  - Refers to grave issues that occasionally arise such as:
    - Serious illness, hospitalisation, an accident.
    - Family bereavement.
    - Ongoing serious personal or emotional circumstances.
  - Extenuating Circumstances do not cover events which are foreseen (e.g. 21st party, wedding, personal travel etc).
- Minor Circumstances (absent for a few days)
  - These situations should be handled locally by making direct contact with the lecturer and or school administrator. Extenuating Circumstances do **not** apply in these cases.

Teaching Assistants

- Francisco J. Pena (francisco.pena@insight-centre.org)
- Erika Duriakova (erika.duriakova@ucd.ie)

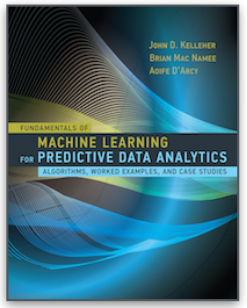
Both are Postdoctoral researchers at Insight Centre for Data Analytics working on problems leveraging machine learning and recommender systems

Additional Reading Materials



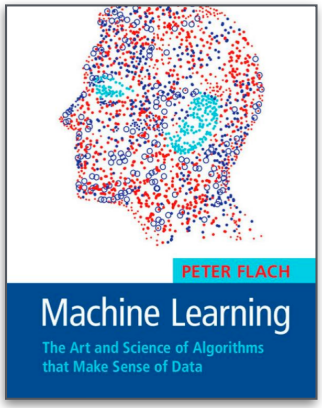
Machine Learning
McGraw-Hill

Tom M. Mitchell



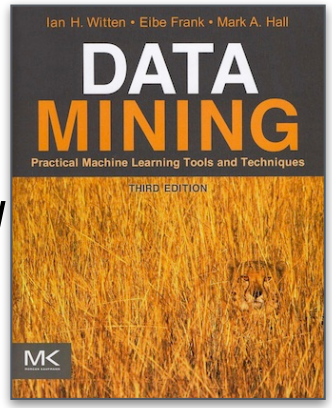
Fundamentals of Machine Learning for Predictive Data Analytics

John D. Kelleher, Brian Mac Namee, Aoife D'Arcy



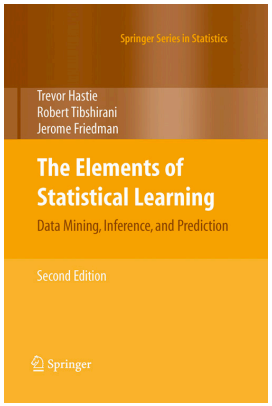
Machine Learning: The Art and Science of Algorithms that Make Sense of Data

Peter Flach



Data Mining: Practical Machine Learning Tools and Techniques, 3rd Ed

Ian H. Witten, Eibe Frank, Mark A. Hall



The Elements of Statistical Learning: Data Mining, Inference, and Prediction

Trevor Hastie, Robert Tibshirani and Jerome Friedman

Module Outline: Topics Covered

- ML Fundamentals
- Supervised Learning
  - Classification: KNNs, Decision trees, Naive Bayes
  - Neural networks
  - Linear regression
- Unsupervised Learning Algorithms
  - k-Means
  - Hierarchical clustering
- Working with Data
  - Dimensionality reduction, feature selection
- Evaluating the Performance of ML systems
- Further Topics in ML, including ensembles

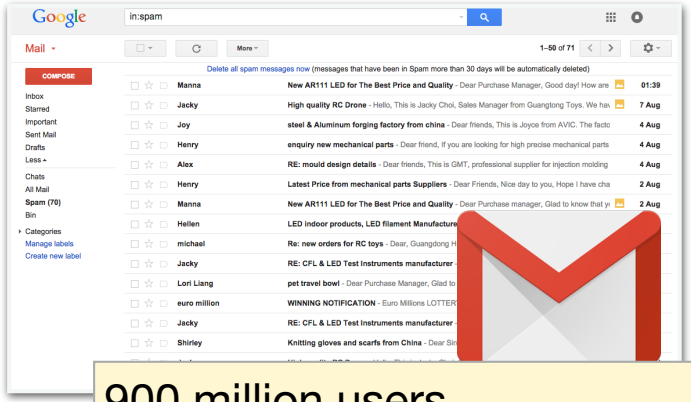
Module Outline: What this module will *not* cover?

- Theoretical foundations of machine learning
  - Computational complexity of learning, V-C dimension, High-dimensional geometry, Optimality guarantees of clustering algorithms, Efficiency and scalability of learning algorithms
- Statistical foundations of machine learning
  - Probability distribution functions, Statistical hypothesis testing
- Programming languages
  - Python, Java
- Data mining issues
  - Data cleaning, Data filtering, Data storage
- Advanced topics in machine learning
  - Reinforcement learning, Deep learning

Why Study Machine Learning?

- Explosion in rich, complex data to analyse - online and offline.
- Significant recent progress in algorithms and theory.
- Computational power is now available.
- Industry demand - Data scientists, Data engineers…
- New applications in many disciplines - Medicine, engineering, humanities, agriculture, physics…

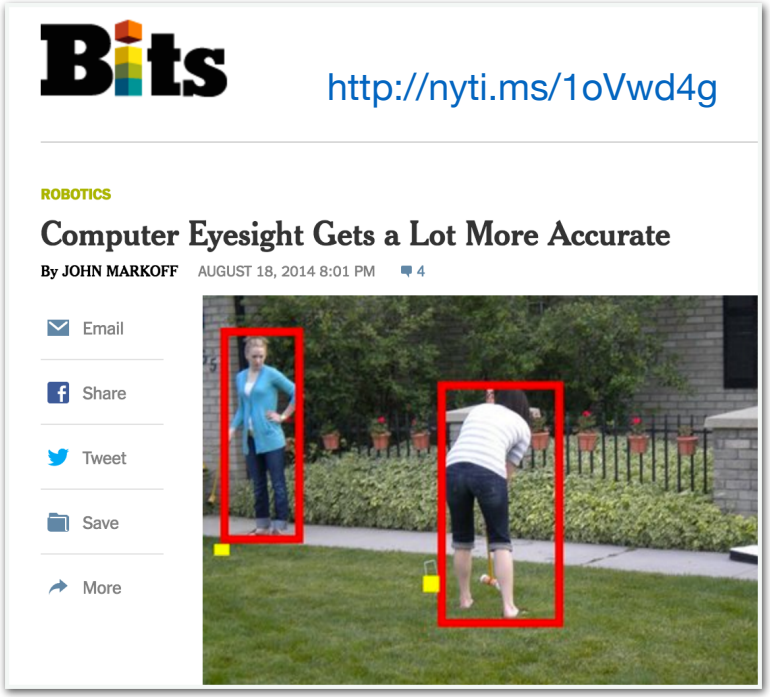
316 million monthly active users
500 million tweets per day
5 billion user sessions per day

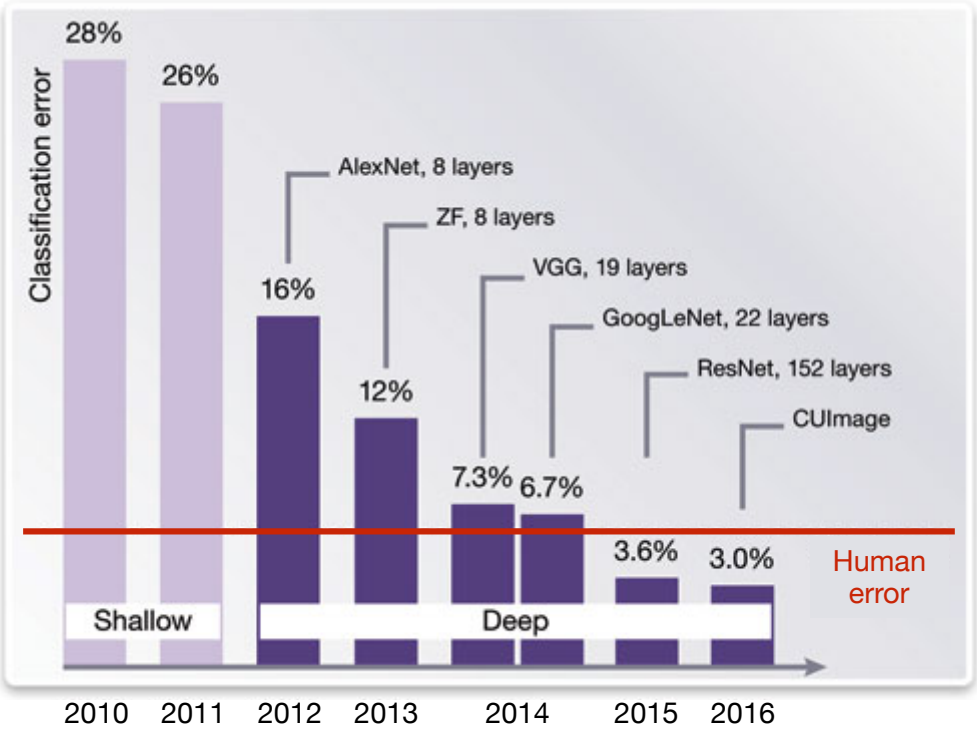


900 million users
Handles 2+ trillion mails per year

Progress in Machine Learning

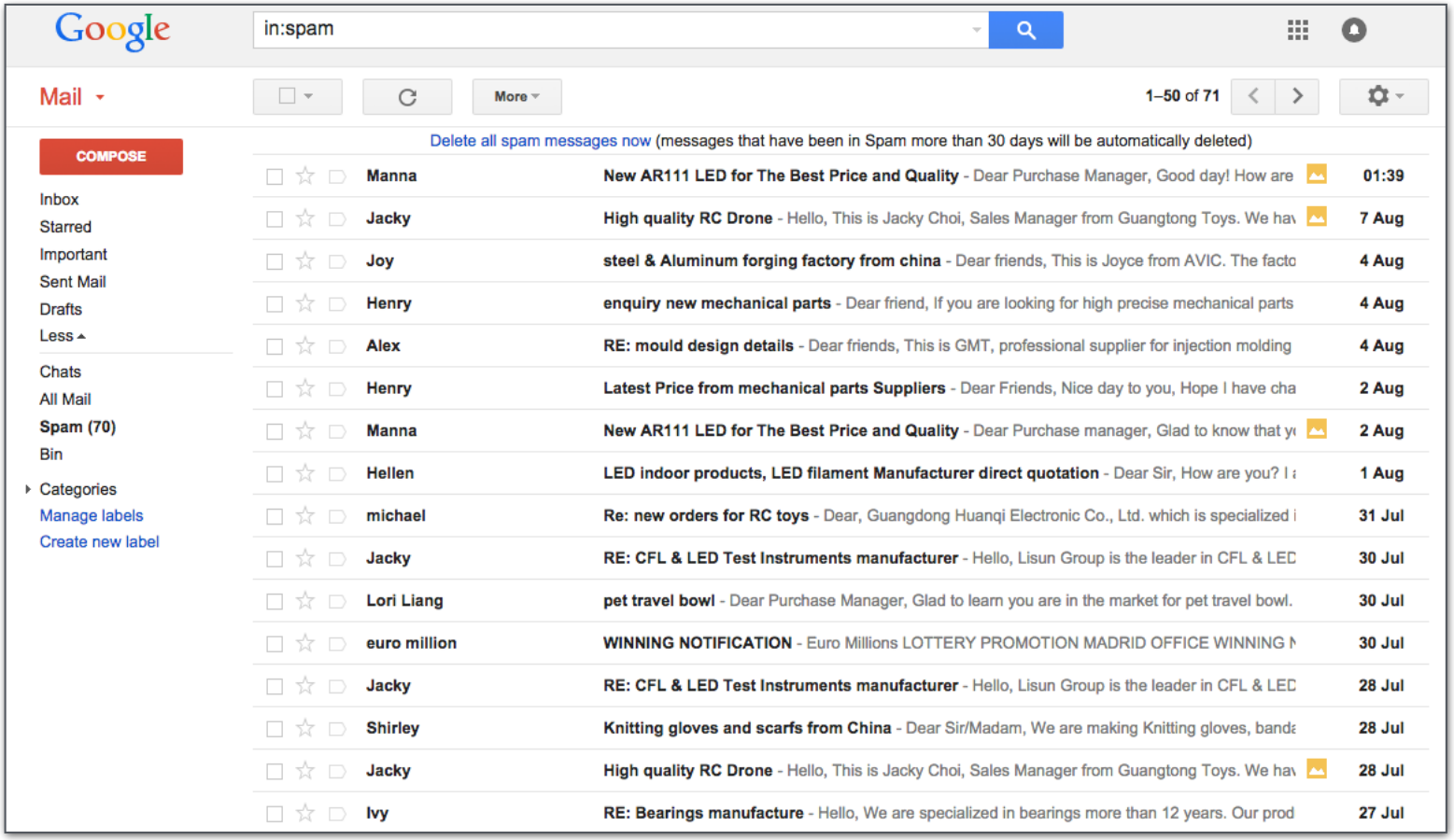
- The annual ImageNet competition compares algorithms for detecting and classifying specific items in images.
- In 2011 the best approach was able to correctly classify images with 25.7% error rate.
- In 2015 **Google** introduced a deep learning approach achieving < 5% error. In 2016 this was below "human error rate" at 3%.





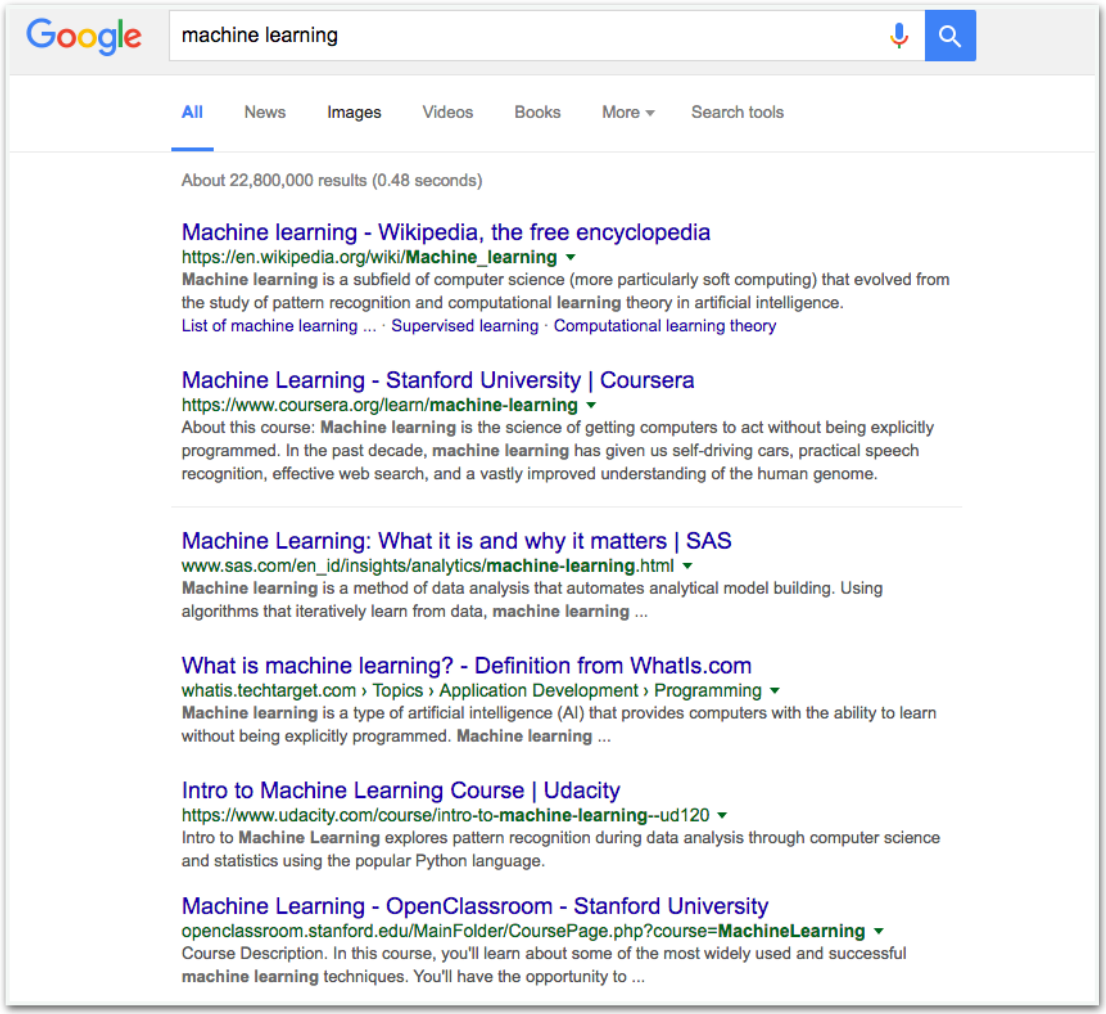
Application: Spam Classification

Apply a learning algorithm to automatically classify incoming emails as *spam* or *non-spam*, based on previous examples of legitimate and spam email.



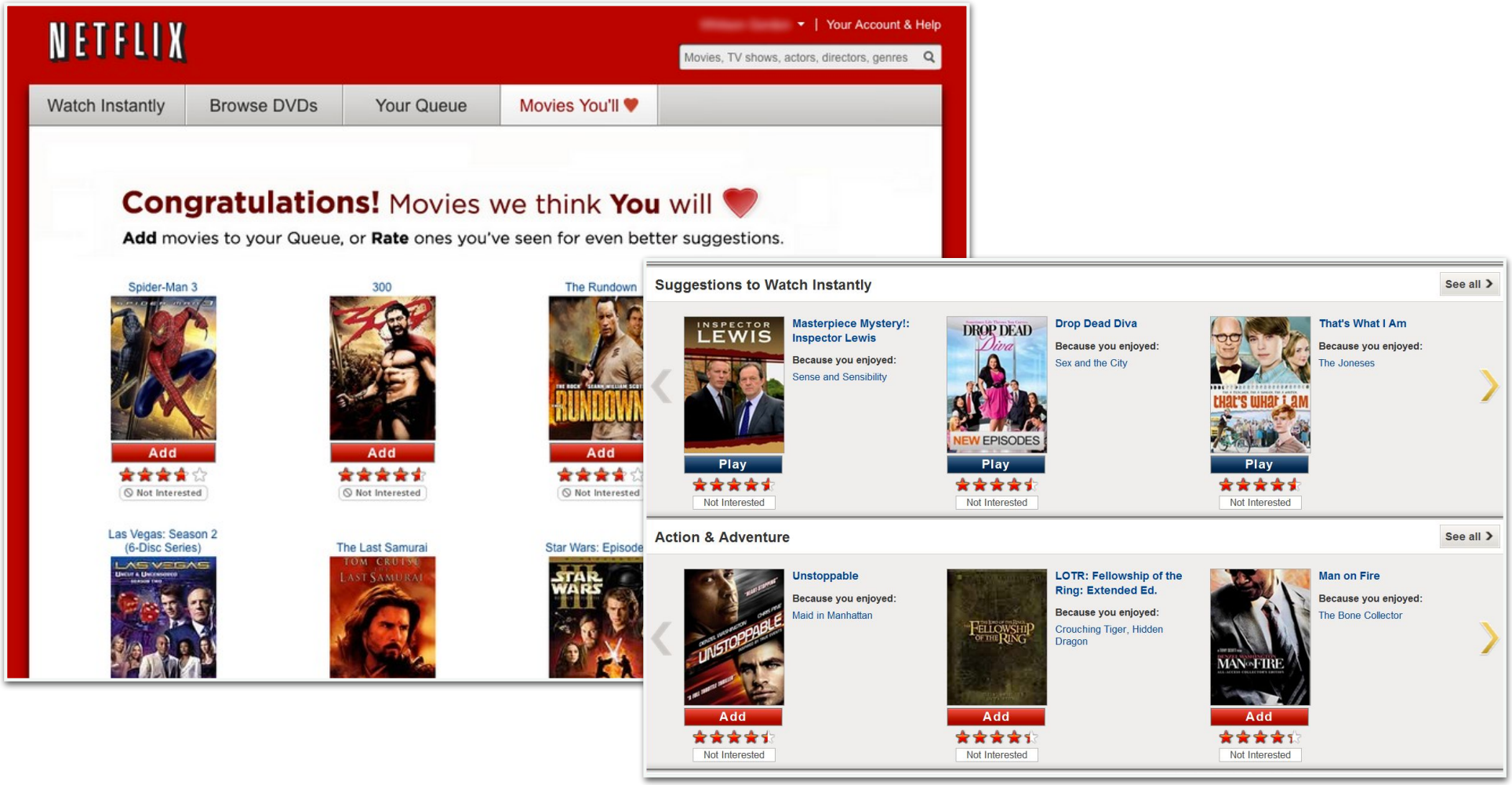
Application: Web Search

Submit a query to a search engine, it finds pages relevant to the query, and returns them ranked by relevance.



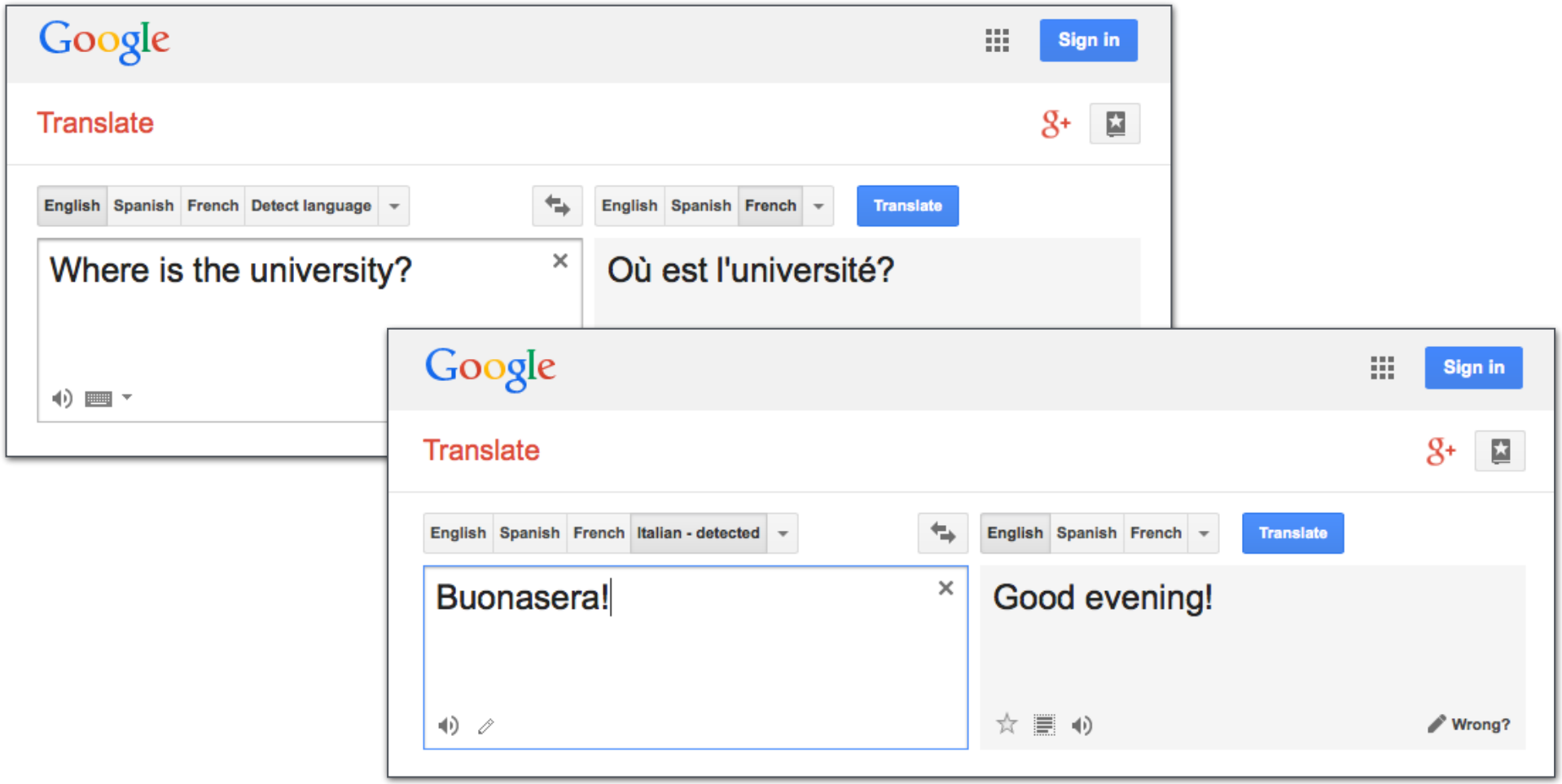
Application: Movie Recommendation

Netflix provides personalised recommendations for movies you might like, based on the previous ratings of other users.



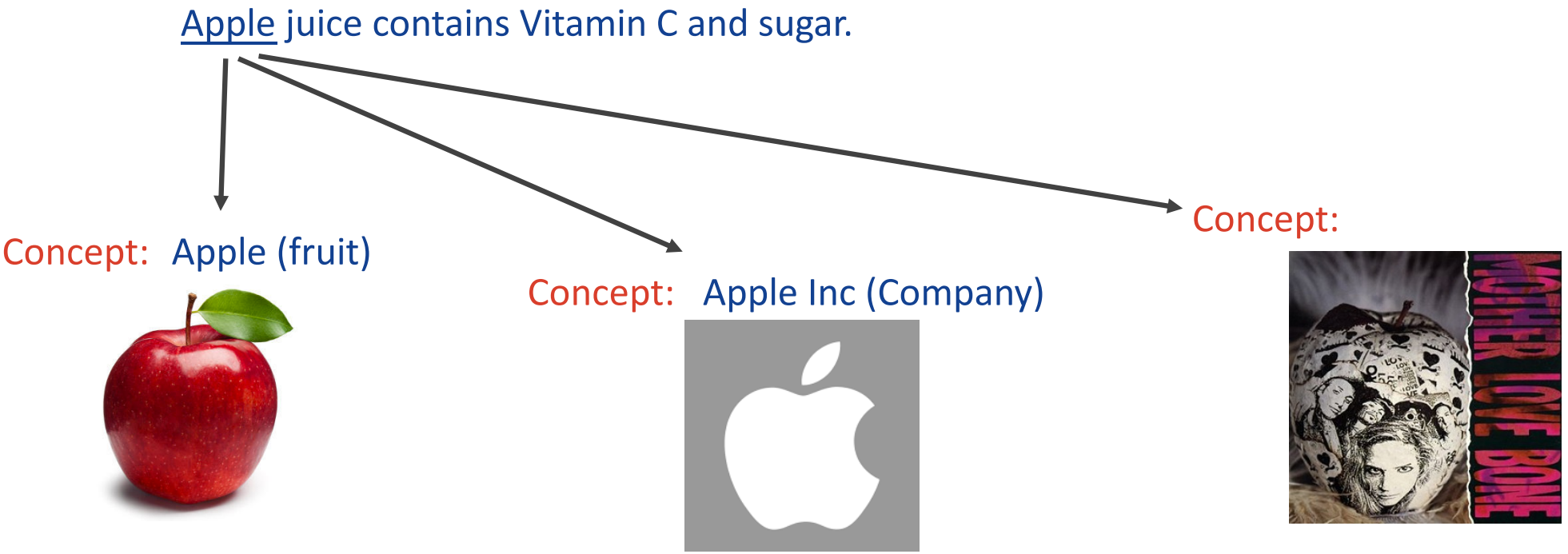
Application: Machine Translation

Use examples of translated documents to learn how to translate text between the two languages.



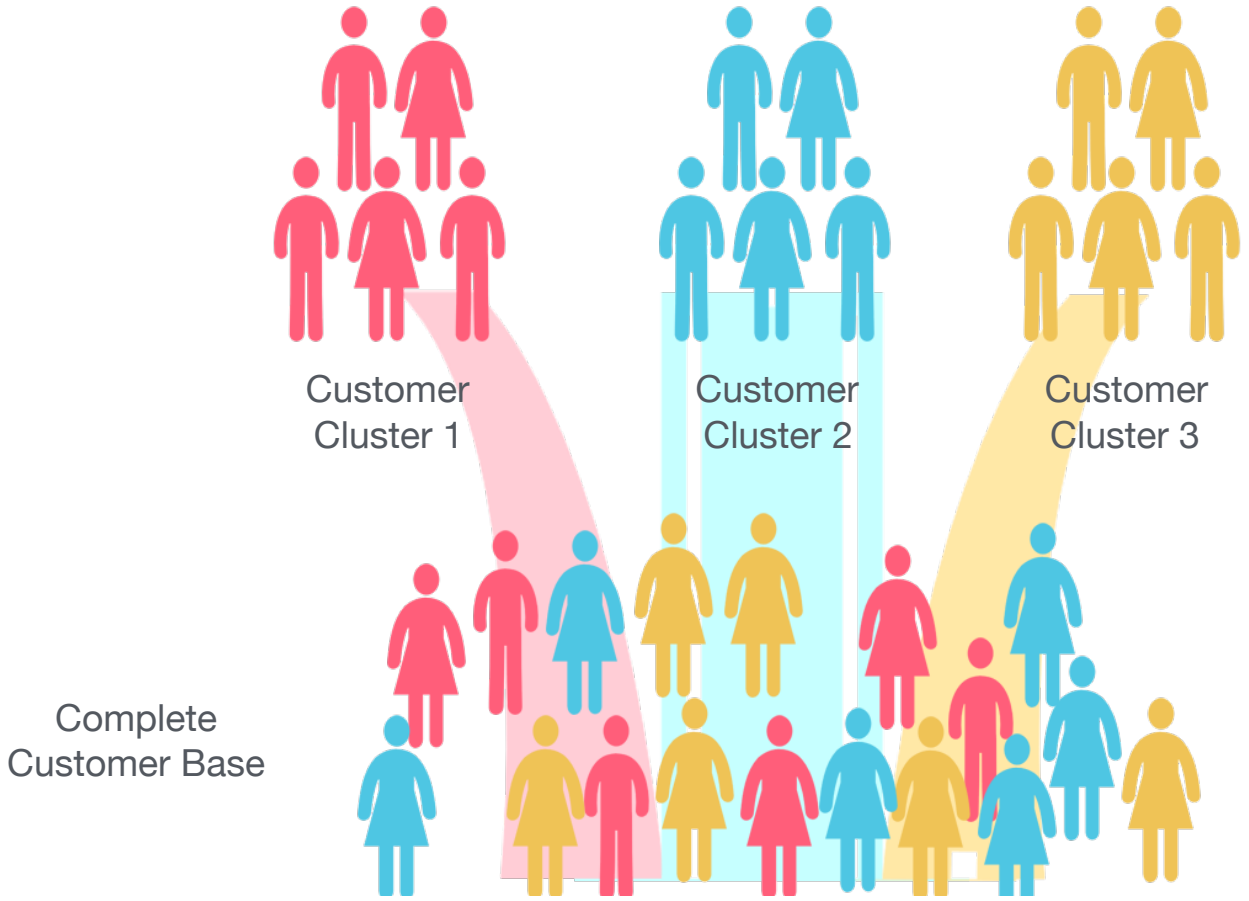
Application: Sense disambiguation

Map text phrases to their correct canonical representation in vocabulary



Application: Marketing

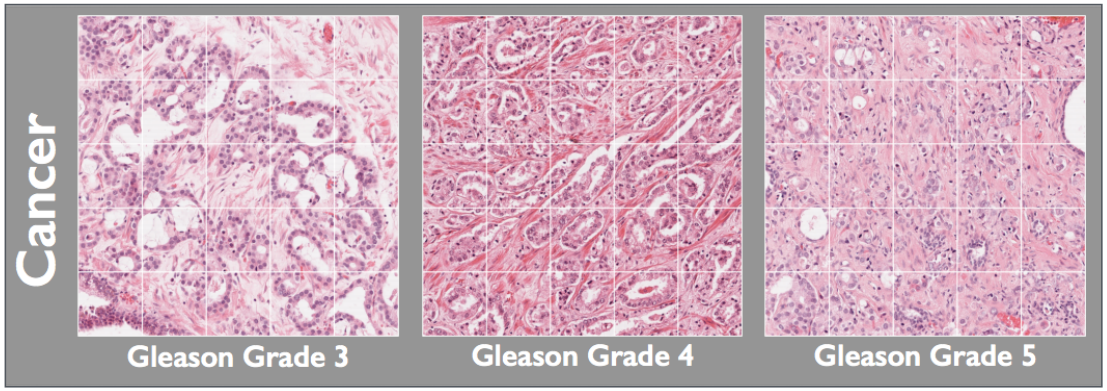
Automatically segment a large customer base into clusters of individuals that share similar characteristics, for target marketing.

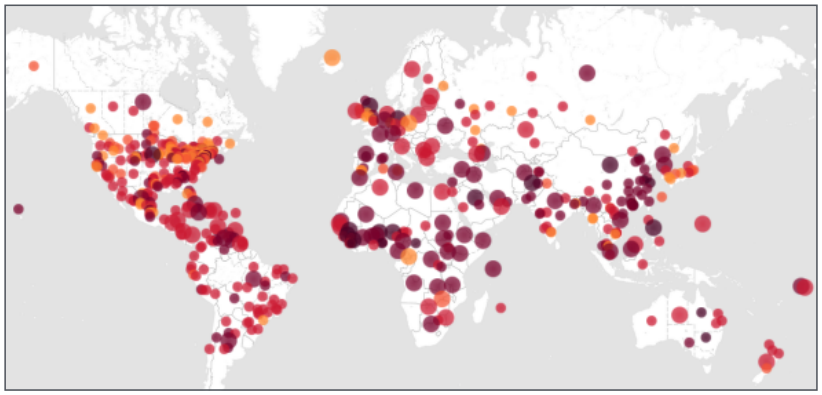


Application: Medicine

Machine Learning provides tools and support in diagnostic and prognostic tasks in a variety of medical domains, including:

- Disease identification and diagnosis
- Prediction of disease progression
- Medical image analysis and understanding
- Personalised medicine and behavioural modification
- Epidemic outbreak prediction

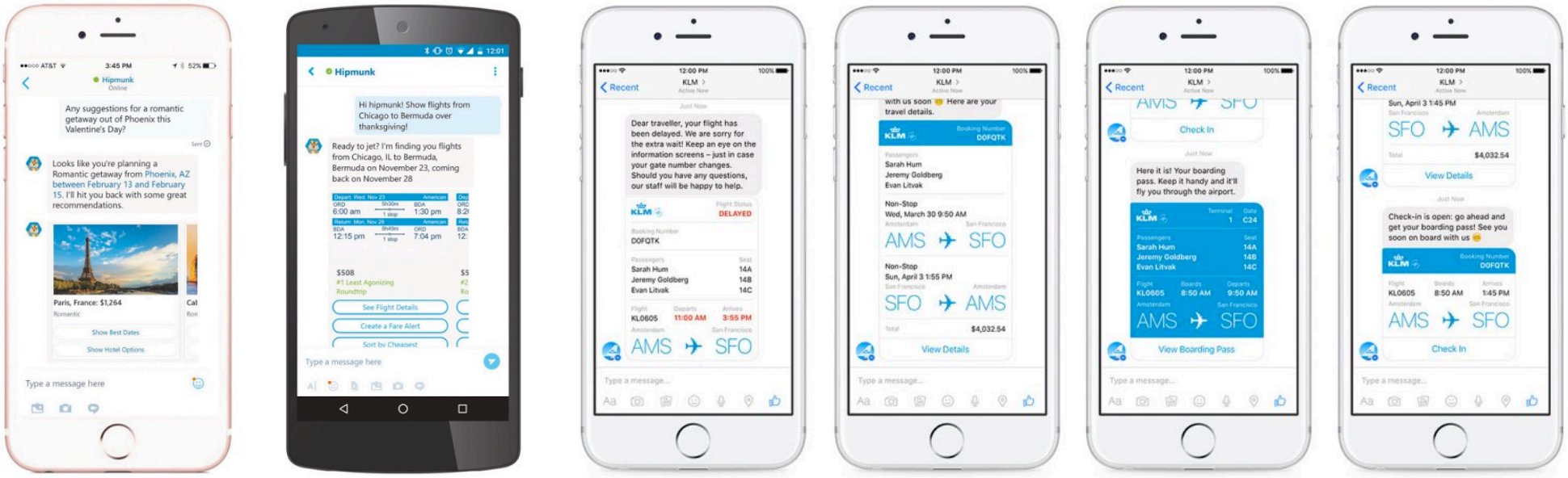




Application: Chatbots

Machine learning algorithms are now widely used in "chatbots", intelligent digital assistants designed to automatically respond to user requests via conversational interfaces. Examples include:

- Responding to customer service enquiries
- Personal travel and entertainment assistants
- Providing stock updates and breaking news
- Handling food orders and delivery



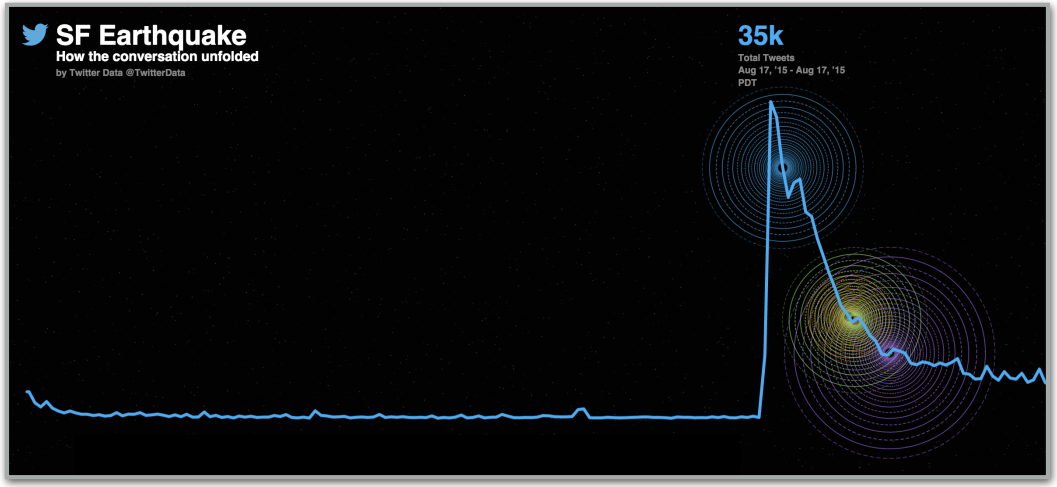
techtimes.com

whatbot.com 23

Application: Anomaly Detection

Algorithms can find patterns in data that don't conform to a model of “normal” behaviour in a system. In some systems, these are rare events. In other systems, these are unexpected bursts of activity.

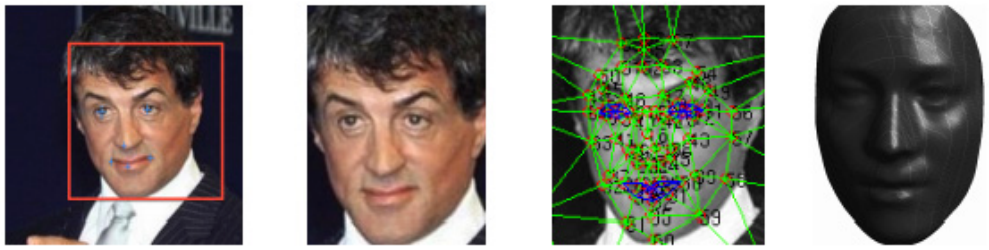
- *Cybersecurity*: Spike in number of false login attempts.
- *Payment systems*: High number of failed/incomplete payments.
- *Fraud detection*: Unusual patterns in financial networks.
- *Fault detection*: Timely detection and diagnosis of faults in aircraft.
- *Event detection*: Sudden spike in volume of social media posts.



Application: Face Recognition

Facebook tags photos by comparing them to profile pictures.

"We currently use facial recognition software that uses an algorithm to calculate a unique template based on someone's facial features, like the distance between the eyes, nose and ears. This template is based on your profile pictures and photos you've been tagged in on Facebook."



In 2013, Facebook revealed its users have uploaded > 250 billion photos, and are uploading 350 million new photos each day.

Using other cues (e.g. hair style, clothing), allows Facebook to accurately identify people, even when their face is obscured.

http://www.fastcolabs.com/3028414/how-facebooks-machines-got-so-good-at-recognizing-your-face

http://arstechnica.com/?p=695873

Application: Image Understanding

Deep learning techniques have been developed to automatically produce captions for complex situational images.

A **helicopter** in the **air** **dropping** an **unknown item** from the **helicopter** to the **land**. (Confidence: 93.8%)

A **person** **hoisting** a **flag** from the **land** using a **rope** **outdoors**. (1.2%)

Smoke **cresting** **outdoors**. (0.6%)

A **man** **barbecuing** **meat** **outdoors**. (Confidence: 29.2%)

A **man** **pushing** a **baby buggy** with his **hand** in a **hospital**. (10.1%)

A **man** **wheeling** **baggage** **outdoors**. (9.4%)

A **man** **bowing** **outside**. (7.7%)

http://nyti.ms/2cW1ng0

Application: Self-Driving Cars

- Car manufacturers and researchers are exploring the potential of self-driving cars. These involve the analysis of huge volumes of sensor data, categorised using ML approaches combined with human labelling.
- **2004:** Autonomous cars tried to navigate a 150 mile desert DARPA race. None of the 21 teams finished.
- **2015:** Google driverless test vehicles have driven nearly 1 million miles, with no accidents caused by a self-driving car. Prototypes launched on public roads.

http://googleblog.blogspot.ie/2015/05/self-driving-vehicle-prototypes-on-road.html

http://www.nature.com/news/autonomous-vehicles-no-drivers-required-1.16832

~~Future~~ Application: Self-Driving Cars

Self-driving cars are here

Drive.ai will offer a self-driving car service for public use in Frisco, Texas starting in July, 2018.

Self-driving cars are no longer a futuristic AI technology. They're here, and will soon make transportation cheaper and more convenient.

Supervised v Unsupervised Learning

- Supervised Learning:
  An algorithm that learns a function from examples of its inputs and outputs. It uses manually-labelled example data (i.e. a training set) to predict the correct answer for new unseen query inputs.

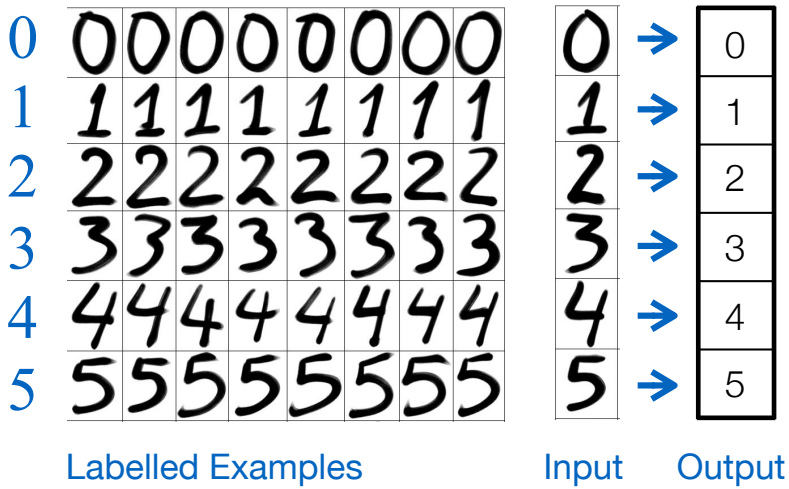
  e.g. Classification, regression algorithms

- Unsupervised Learning:
  An algorithm that finds structure in data where no manually labelled examples are available as inputs - i.e. there is no training set. These algorithms are more focused on data exploration and knowledge discovery.

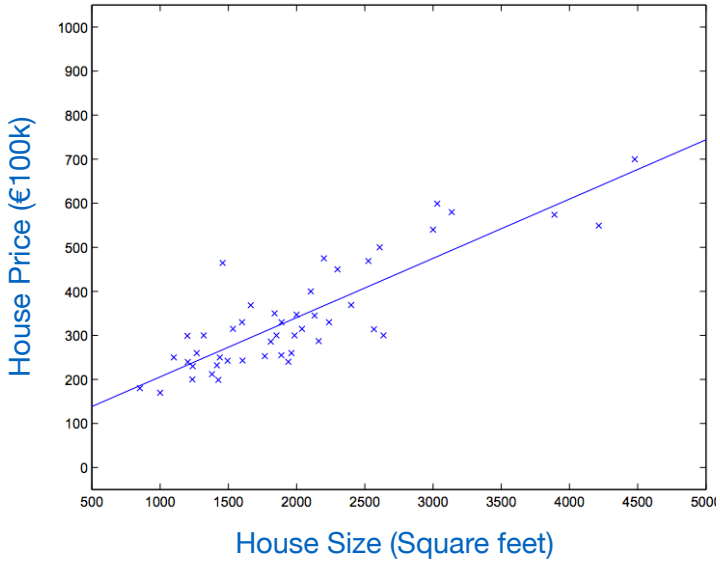
  e.g. Clustering, topic modelling algorithms

Supervised Learning

- **Classification:**
  Examples represented by a set of features, which help decide the target class to which a new query input belongs
  (i.e. the output is a class label).



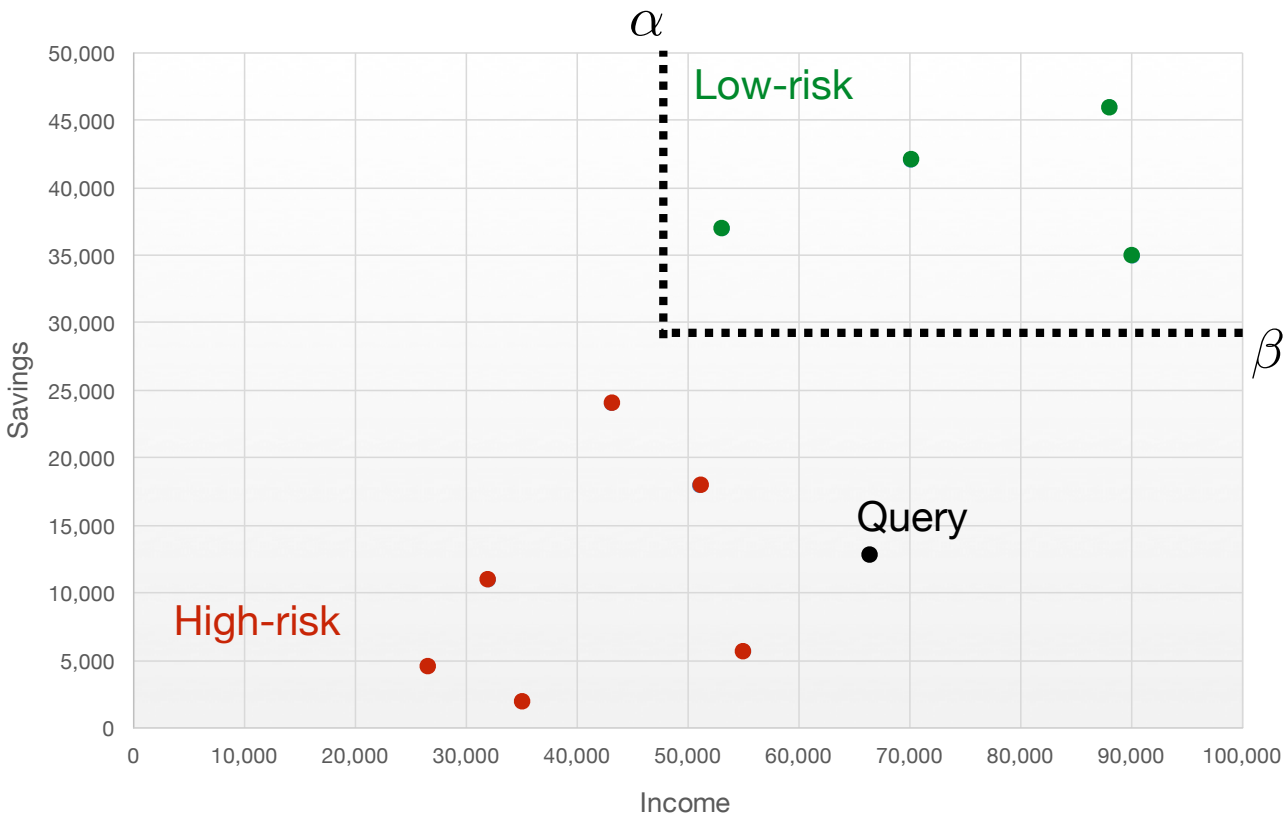
- **Regression:**
  Examples characterised by a set of features, which help decide the value of a continuous output variable
  (i.e. the output is a number).



Classification Tasks

Example:
Credit scoring

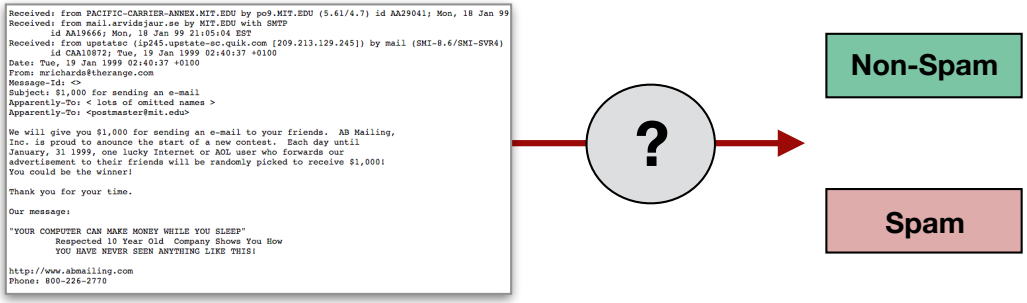
Manually classify customers into two categories (low-risk and high-risk) based on savings and income data.



Q. Can we train an algorithm to learn to automatically classify new customers as either low-risk or high-risk?
i.e. can we learn α and β?

Classification Tasks

- Binary classification: Assign a new query input to one of two possible target class labels.



Input

2 classes

- Multiclass classification: Assign a new query input to one of $M > 2$ different target class labels.

Input

4 classes

Representing Data

- Commonly we use a tabular structure to represent a dataset, often referred to as the analytics base table (ABT).
- Each row represents a different example, and is composed of a set of descriptive features.
- For classification, we have training data where each row also has a target class label - i.e. the "correct answer".

	Descriptive Features									Target Class
Examples	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Representing Data

- The descriptive features used to represent examples can be distinguished by the type and number of values they can take.
  - Binary: Takes only two values - a boolean True/False decision
    e.g. married={True,False}, test_result={Pass,Fail}
  - Categorical (Nominal): A feature that takes values from two or more categories, with no intrinsic ordering to the categories.
    e.g. blood_group={A,B,AB,O}, nationality={French,Irish,Italian}
  - Ordinal: Similar to a categorical variable, but there is a clear ordering of the variables.
    e.g. grade={A,B,C,D,E,F}, dosage={Low,Medium,High}
  - Continuous: Numeric measurements, with or without a fixed range for the values.
    e.g. temperature, weight, height, latitude, longitude etc.

Typical Classification Task

- Training set with N=10 examples (customers). Each is described by D=5 features: 3 continuous, 2 categorical
- Each example has one of two class labels = {High-risk,Low-risk}

Example	*Income*	*Savings*	*Married*	*Gender*	*Age*	*Class*
1	35,000	2,000	Y	M	32	High-risk
2	51,000	18,000	N	M	34	High-risk
3	70,000	42,000	Y	F	41	Low-risk
4	26,500	4,500	N	M	22	High-risk
5	32,000	11,000	N	F	25	High-risk
6	53,000	37,000	N	F	39	Low-risk
7	88,000	46,000	Y	M	48	Low-risk
8	55,000	5,700	N	M	55	High-risk
9	90,000	35,000	Y	F	61	Low-risk
10	43,000	24,000	Y	M	33	High-risk

Q. To which class does this new customer belong?

Example	*Income*	*Savings*	*Married*	*Gender*	*Age*	*Class*
X	66,000	13,000	Y	M	44	???

Typical Classification Task

- Training set with N=10 examples (fruit). Each is described by D=4 features: 3 continuous, 1 categorical
- Each has one of three class labels = {Apple,Pear,Orange}

Example	*Height*	*Width*	*Taste*	*Weight*	*Class*
1	60	62	Sweet	186	Apple
2	70	53	Sweet	180	Pear
3	55	50	Tart	152	Apple
4	76	40	Sweet	152	Pear
5	68	71	Tart	207	Orange
6	65	68	Sour	221	Apple
7	63	45	Sweet	140	Pear
8	55	56	Sweet	154	Apple
9	76	78	Tart	211	Orange
10	60	58	Sour	175	Apple

Q. To which class does this new fruit belong?

Example	*Height*	*Width*	*Taste*	*Weight*	*Class*
X	63	68	Sweet	168	???

Classification Tasks

- **Evaluation**: Standard approach for classification tasks is to split the set of examples into a *training set* and a *test set.*
- Training set: Examples provided to the classifier to build a model of the data. Each example has been manually assigned a class label.
- Test set: Examples held back from the classifier, which are used to evaluate the accuracy of the classifier. ***Test examples are completely separate from the training set.***
- Why not just train on all the data?
  - The test set is used to evaluate how well the model built by the classifier will generalise to new input examples.
  - Using the training data will give us over-optimistic results!

Classification Algorithms

- Many different learning algorithms exist for classification (e.g. *k*-nearest neighbour, decision tree, neural network, support vector machine).
- Problem dimensions will often determine which classification algorithm will be practically applicable, due to processing, memory, and storage constraints.

1. Number of input examples *N*.
   → Sometimes millions of input examples.
2. Number of features (dimensions) *D* representing each input example.
   → Often 10-1000, but sometimes far higher.
3. Number of target classes *M*.
   → Often small (binary), but sometimes far higher.