

Classification with the Perceptron

L101: Machine Learning for Language Processing
Andreas Vlachos



L101 Practicalities

Lecturer: Andreas Vlachos (av308@cam.ac.uk)

Additional Lecturer: Pushkar Mishra (pm576@cam.ac.uk)

Lectures/seminars on Tuesdays and Thursdays at noon.

Drop me an email with your @cam addresses to add you to the mailing list if you did not receive a message from it already

Materials and information about the assessment:

<https://www.cl.cam.ac.uk/teaching/2526/L101/>



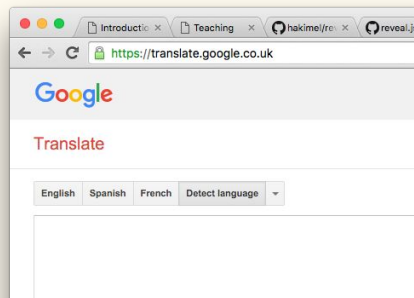
L101 Assessment (part 1)

- Students will receive one tick worth 5% for attendance at seminar sessions, reading of assigned material, and satisfactory contribution during seminars.
- Students will receive a second tick worth 5% for a satisfactory presentation of an assigned paper during a seminar
 - These start from the **5th of November**
 - List of papers under Course Materials tab
 - Full instructions there, but need to select papers and send them to me (av308@cam.ac.uk) by **22nd of October at noon**, to make the schedule

L101 Assessment (part 2)

- Students will undertake a small project to be agreed with the lecturer and write a project report of not more than 5000 words.
 - Pick a dataset/task
 - Literature survey
 - Implement a system motivated by the survey
 - Compare against previous work
- Send us (`{av308,pm576}@cam.ac.uk`) your ideas ASAP, and no later than the **21st of October** (need to have it agreed by the **3rd of November**)
- The report will be due on the **19th of January**, will be assessed by us, and will account for 90% of the module marks.

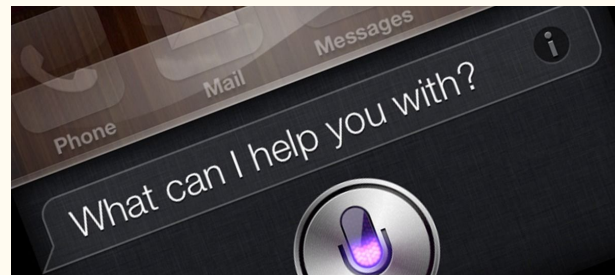
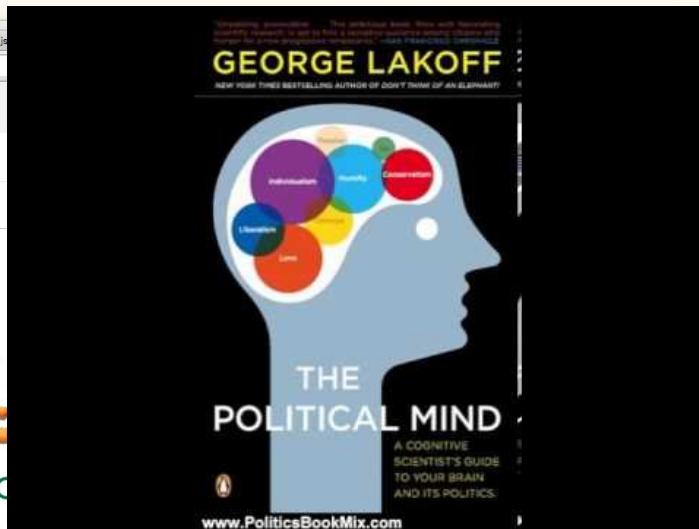
Why Natural Language Processing (NLP)?



Dan Jurafsky

Professor and Chair, Linguistics
Professor, Computer Science
Stanford University

wit.ai



Classification

Given a piece of text, assign a label from a predefined set

What could the labels be?

- Positive/negative sentiment
- Topical
- Author name (authorship identification)
- Biased or not
- etc.

Formulation

Given input instance x , parameters w , a classifier is a function f that predicts $\hat{y}$:

Binary (1/-1, sometimes 0/1): $\hat{y} = \textit{sign}(f(x; w))$

Multiclass (1 of N): $\hat{y} = \arg \max_{y \in \mathcal{Y}} f(x; w, y)$

Multilabel (a set of labels, possibly empty): $\hat{y} = \arg \max_{y \in \mathcal{P}(\mathcal{Y})} f(x; w, y)$

Features

Given a document, what features would you use to predict its:

- topic?
- sentiment?
- author name?
- factual correctness?

Some ideas:

- bag of words (n-grams)
- meta-data
- sub-word features
- external evidence, common sense...
- Often these are denoted with the feature function: $\phi(x) \in \mathfrak{R}^k$

Binary linear classifier

$$\hat{y} = \textit{sign}(w \cdot \phi(x))$$

The “linear” part refers to the function f :

$$w \cdot \phi(x)$$

How do we learn the weights w ?

Supervised learning

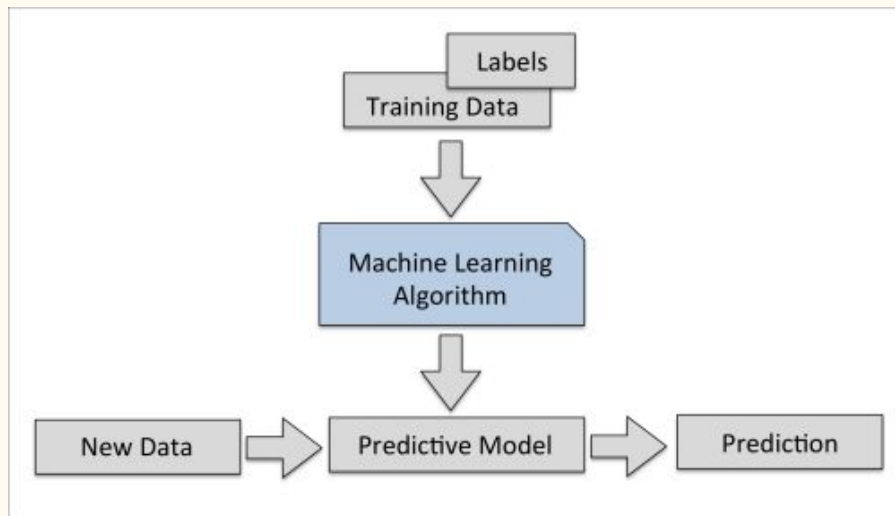


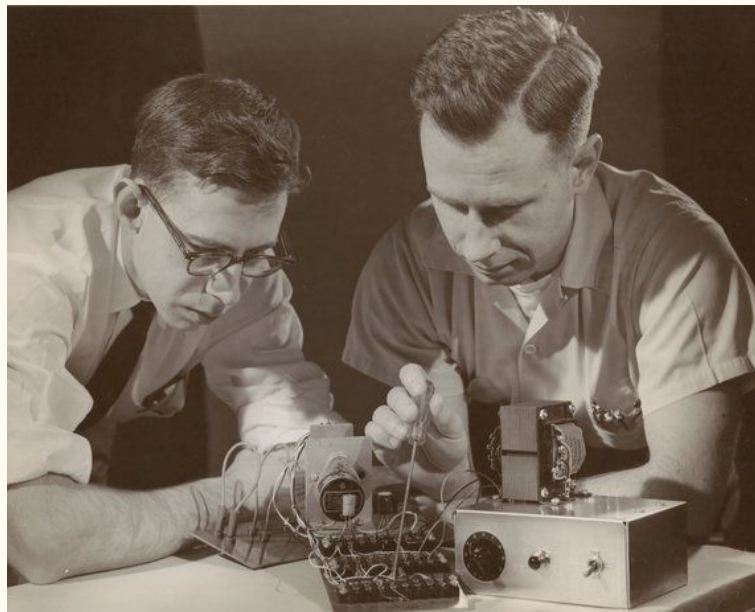
Image Credit:
Sebastian Raschka

Given labeled training data of the form:

$$D = \{(x^1, y^1), \dots (x^M, y^M)\}$$

Learn weights w that **generalize** well to new instances

The perceptron



Proposed by Rosenblatt in 1958, still close to state-of-the-art

The perceptron

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo
of Computer Designed to
Read and Grow Wiser

WASHINGTON, July. 7 (UPI)—The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's \$2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of \$100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human be-

ings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Without Human Controls

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying its surroundings without any human training or control."

The "brain" is designed to remember images and information it has perceived itself. Ordinary computers remember only what is fed into them on punch cards or magnetic tape.

Later Perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted.

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

1958 New York Times...

In today's demonstration, the "704" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

Learns by Doing

In the first fifty trials, the machine made no distinction between them. It then started registering a "Q" for the left squares and "O" for the right squares.

Dr. Rosenblatt said he could explain why the machine learned only in highly technical terms. But he said the computer had undergone a "self-induced change in the wiring diagram."

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

AI hype is not a new problem

We should always remember when we talk to the public

The perceptron algorithm

Input: training examples $\mathcal{D} = \{(x^1, y^1), \dots (x^M, y^M)\}$

Initialize weights $w = (0, \dots, 0)$

for $(x, y) \in \mathcal{D}$ **do**

 Predict label $\hat{y} = \text{sign}(w \cdot \phi(x))$

if $\hat{y} \neq y$ **then**

 Update $w = w + y\phi(x)$

end if

end for

error-driven, online learning

Testing our intuitions

Given the following tweets labeled with sentiment and assuming bag of words:

negative	Very sad about Iran.
negative	No Sat off...Need to work 6 days a week.
negative	I'm a sad panda today.
positive	such a beautiful satisfying day of bargain shopping. loves it.
positive	who else is in a happy mood??
positive	actually quite happy today.

- What weights do you expect your perceptron to learn?
- Do you think they would generalize well?

Sparsity

In NLP, no matter how large our training dataset, we will never see (enough of) all the words/features.

- features unseen in training are ignored in testing
- there are ways to ameliorate this issue (e.g. word clusters, word vectors), but it never goes away
- there will be texts containing only unseen words

Embeddings/continuous representations (wait for lecture 5) help a lot with these, but unseen words/features are more challenging than seen ones

Improving the perceptron

Input: training examples $\mathcal{D} = \{(x^1, y^1), \dots, (x^M, y^M)\}$

Initialize counter $c = 0$, weights $w_c = (0, \dots, 0)$

for $i = 1 \dots \text{maxIter}$ **do**

 Shuffle $\mathcal{D}$

for $(x, y) \in \mathcal{D}$ **do**

 Predict label $\hat{y} = \text{sign}(w_c \cdot \phi(x))$

if $\hat{y} \neq y$ **then**

 Update $w_{c+1} = w_c + y\phi(x)$

else

$w_{c+1} = w_c$

end if

$c = c + 1$

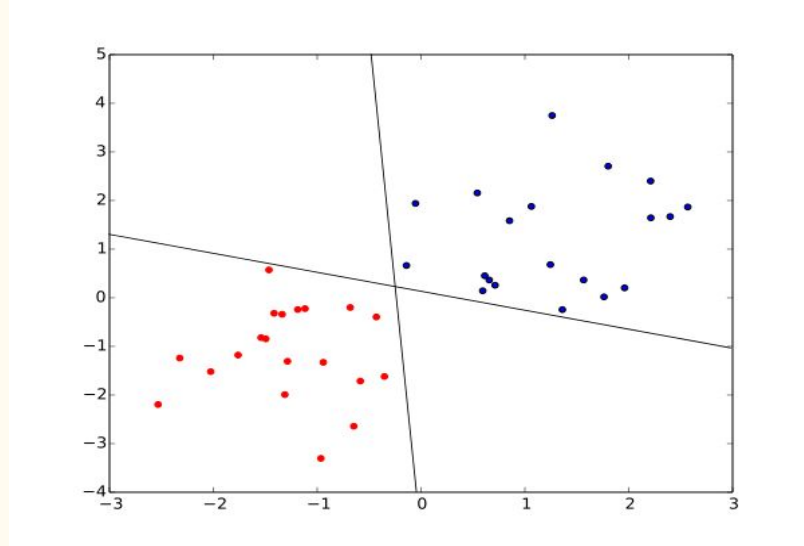
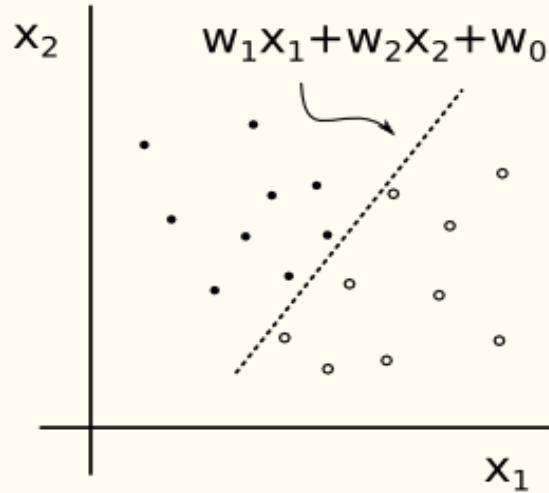
end for

end for

$w = \frac{1}{c} \sum w_0 + \dots + w_c$

- Multiple passes
- Shuffling
- Averaging

On separating hyperplanes



If data not linearly separable, perceptron does not converge (will always update)
Even if linearly separable, no guarantee of finding a good separating hyperplane

Image credits: <https://johnpatrickroach.com/2016/09/24/training-a-perceptron-model-in-python/> (left),
https://en.wikipedia.org/wiki/Perceptron#/media/File:Perceptron_cant_choose.svg (right)

Binary to multiclass

Binary: $\hat{y} = \textit{sign}(w \cdot \phi(x))$

Multiclass 1: $\hat{y} = \arg \max_{y \in \mathcal{Y}} w_y \cdot \phi(x)$

Input feature map: only describes x .

Multiclass 2: $\hat{y} = \arg \max_{y \in \mathcal{Y}} w \cdot \phi(x, y)$

Joint feature map: compatibility between x and y . More expressive but less intuitive.

Multiclass perceptron (two versions)

Input: training examples $\mathcal{D} = \{(x^1, y^1), \dots, (x^M, y^M)\}$

Initialize weights $w_y = (0, \dots, 0)$

Initialize weights $w = (0, \dots, 0)$

for $(x, y) \in \mathcal{D}$ **do**

Predict label $\hat{y} = \arg \max_{y \in \mathcal{Y}} w_y \cdot \phi(x) / w \cdot \phi(x, y)$

if $\hat{y} \neq y$ **then**

Update $w_y = w_y + \phi(x)$

Update $w_{\hat{y}} = w_{\hat{y}} - \phi(x)$

Update $w = w + \phi(x, y) - \phi(x, \hat{y})$

end if

end for

Multilabel classification

$$\hat{y} = \arg \max_{y \in \mathcal{P}(\mathcal{Y})} f(x; w, y)$$

Can be done in two ways:

- Binary relevance: build a binary classifier for each label in Y
 - Ignores label dependencies
- Multiclass: build a multiclass classifier for all members of the powerset $P(Y)$
 - Can be computationally expensive, training data can be sparse

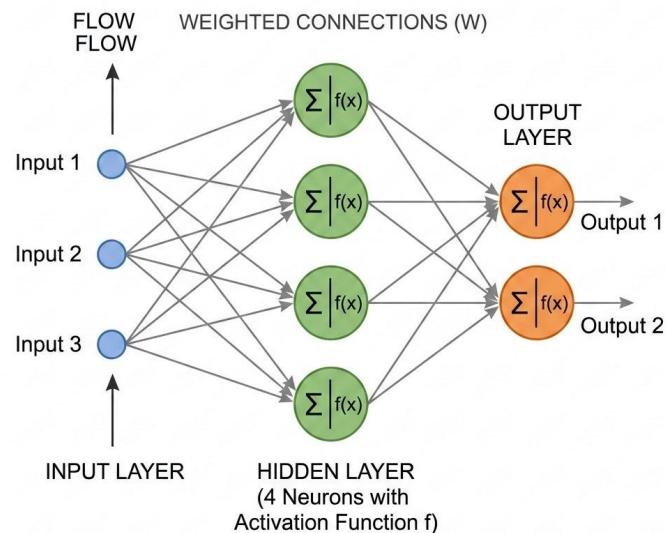
Both are instances of **reduction**: transform complex problems to simpler ones

More advanced methods exist, but always try these two first

Why talk about the perceptron today?

- The classification head of modern NNs/LLMs has the same capacity (linear)
- The main difference is the feature extraction/representation; perceptrons have been multilayer (a.k.a. deep!) too for a long time (Rosenblatt, 1958)
- When we ask an LLM to classify a text we are effectively doing the joint feature map multiclass classification
- More soon (lecture 4)

SIMPLE MULTILAYER PERCEPTRON (MLP) DIAGRAM



Evaluation

$$accuracy = \frac{correctPredictions}{allPredictions}$$

What can go wrong?

Imbalanced datasets: predicting one class gives 90% accuracy

Very common: most topics are not relevant to most documents, etc.

Evaluation

Predicted/Correct	Minority Class	Majority Class
Minority Class	True Positive	False Positive
Majority Class	False Negative	True Negative

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives}$$

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives}$$

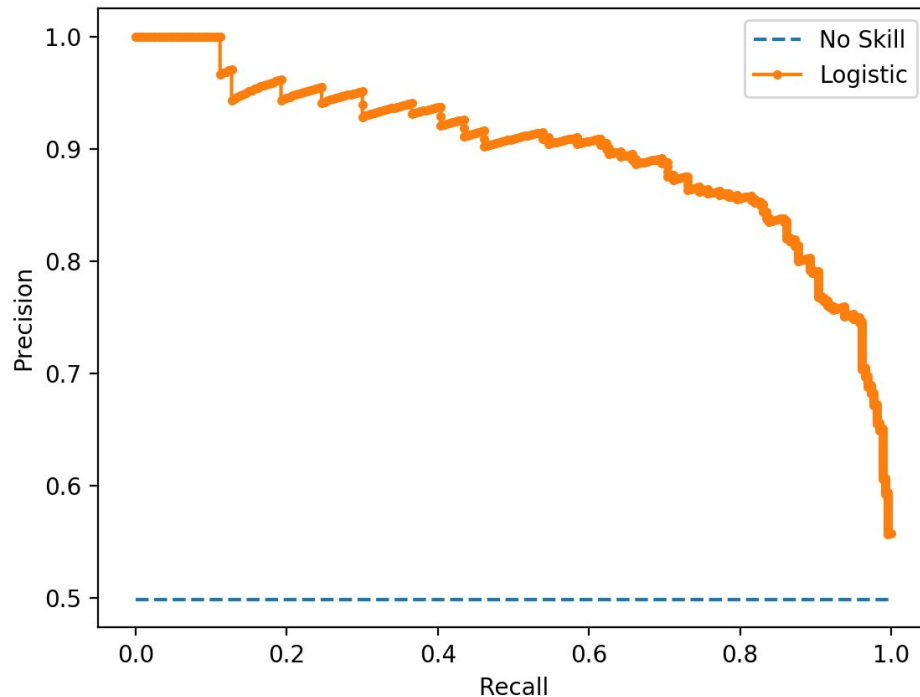
Often combined into their harmonic mean, aka F1-score

Evaluation (cont.)

So far we assumed a fixed decision threshold but often it makes sense to adjust it

Don't pick a single threshold, check them all!

Can summarize in the area under the precision-recall curve (AUC)



What sentiment classifier would you build?



The screenshot shows the IMDb page for the movie "The Godfather" (1972). The page includes a search bar at the top, navigation links for Movies, TV & Showtimes, Celebs, Events & Photos, News & Community, and Watchlist. The movie's title is prominently displayed, along with its rating of 9.2/10 and a Metascore of 100/100. The page also features a synopsis, director information (Francis Ford Coppola), and a list of writers and stars. A "Quick Links" section on the right provides access to various related content like the full cast and crew, plot summary, and user reviews.

IMDb Find Movies, TV shows, Celebrities and more... All  **IMDb Pro** | IMDb Apps | Help

Movies, TV & Showtimes | Celebs, Events & Photos | News & Community | Watchlist | Login

The Godfather (1972) 

 175 min - Crime | Drama - 24 March 1972 (USA)

Your rating: ★★★★★★★★★★ -/10

9.2 Ratings: 9.2/10 from 801,690 users | Metascore: 100/100
Reviews: 1,731 user | 184 critic | 14 from Metacritic.com

The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.

Director: [Francis Ford Coppola](#)

Writers: [Mario Puzo](#) (screenplay), [Francis Ford Coppola](#) (screenplay), [1 more credit](#) »

Stars: [Marlon Brando](#), [Al Pacino](#), [James Caan](#) | [See full cast and crew](#) »

[+ Watchlist](#)  [Watch Trailer](#) [Share...](#)

Quick Links

- [Full Cast and Crew](#)
- [Plot Summary](#)
- [Trivia](#)
- [Parents Guide](#)
- [Quotes](#)
- [User Reviews](#)
- [Awards](#)
- [Release Dates](#)
- [Message Board](#)
- [Company Credits](#)

[Explore More](#)

 **Like** 31,995 people like this. Be the first of your friends.

Related News

- [Academy's Song Nominee](#)

Type: Binary, multiclass, something else?

What features? What weights do you expect for them?

Bibliography

Hal Daumé III's [chapter](#) on the perceptron from his book on machine learning

For more background reading on classification, Kevin Murphy's [introductory book](#) touches upon most important concepts in ML (see also the [second book](#) for more)

For some theoretical analysis on the perceptron and its connection to gradient-based optimization see this [blog post](#) by Francesco Orabona