

Web/App Intelligence Part I: Supervised Machine Learning

Shan-Hung Wu
shwu@cs.nthu.edu.tw

Department of Computer Science,
National Tsing Hua University, Taiwan

Software Design & Studio

Outline

① Web/App Intelligence

② What's Machine Learning?

③ Post Toxicity Detection

Outline

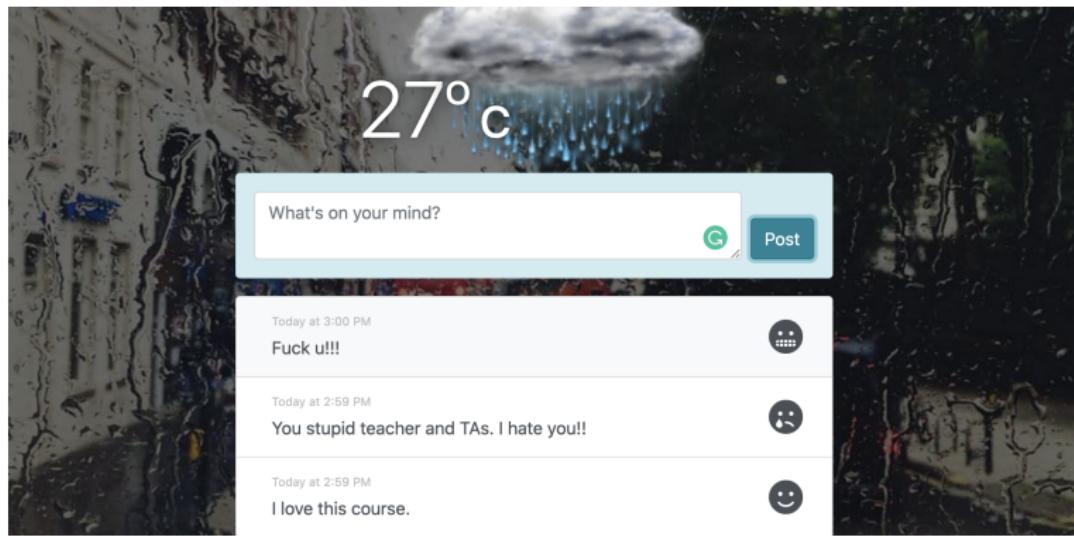
① Web/App Intelligence

② What's Machine Learning?

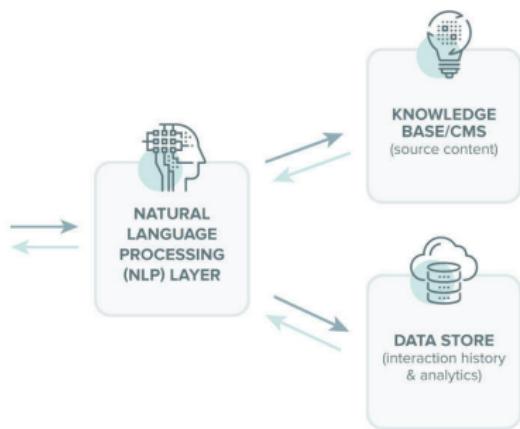
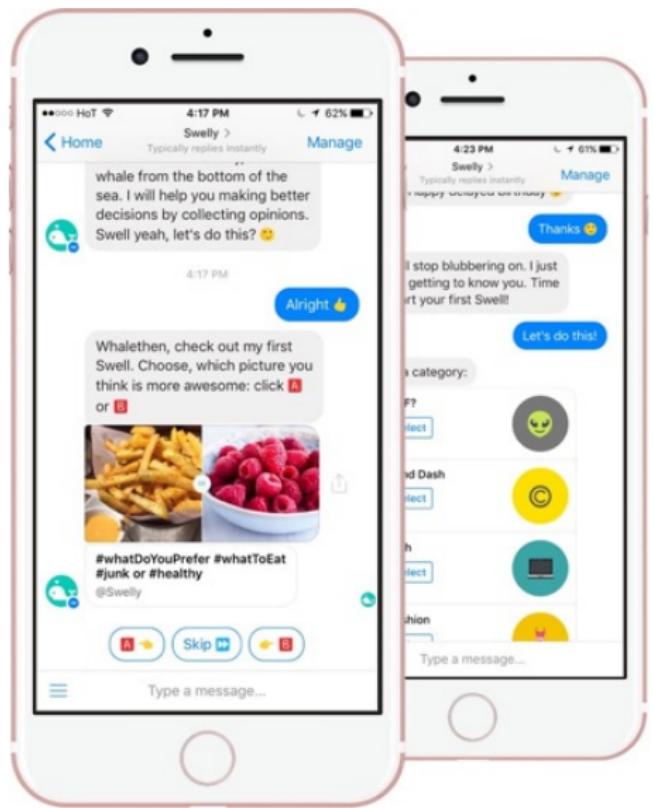
③ Post Toxicity Detection

Let's Make WeatherMood More Intelligent...

```
$ git clone weathermood-toxicity-detection  
$ npm install  
$ npm run start
```



Customer Service Automation



Spam Detection

 Donald J. Trump ✅ @realDonaldTrump · 12h
There is NO WAY (ZERO!) that Mail-In Ballots will be anything less than substantially fraudulent. Mail boxes will be robbed, ballots will be forged & even illegally printed out & fraudulently signed. The Governor of California is sending Ballots to millions of people, anyone.....

 Get the facts about mail-in ballots

 31.2K  29.2K  100.8K 

 Donald J. Trump ✅ @realDonaldTrump

....living in the state, no matter who they are or how they got there, will get one. That will be followed up with professionals telling all of these people, many of whom have never even thought of voting before, how, and for whom, to vote. This will be a Rigged Election. No way!

 Get the facts about mail-in ballots

Product Recommendations

Frequently Bought Together



Price For All Three: \$258.02

Add all three to Cart

- [This item: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition \(Springer Series in Statistics\) by Trevor Hastie](#)
- [Pattern Recognition and Machine Learning \(Information Science and Statistics\) by Christopher M. Bishop](#)
- [Pattern Classification \(2nd Edition\) by Richard O. Duda](#)

Customers Who Bought This Item Also Bought



All of Statistics: A Concise Course in Statist... by Larry Wasserman
★★★★★ (8) \$60.00



Pattern Classification (2nd Edition) by Richard O. Duda
★★★★★ (27) \$117.25



Data Mining: Practical Machine Learning Tools and Algorithms, Second Edition (Texts in Computer Science) by Ian H. Witten and Eibe Frank
★★★★★ (29) \$41.55



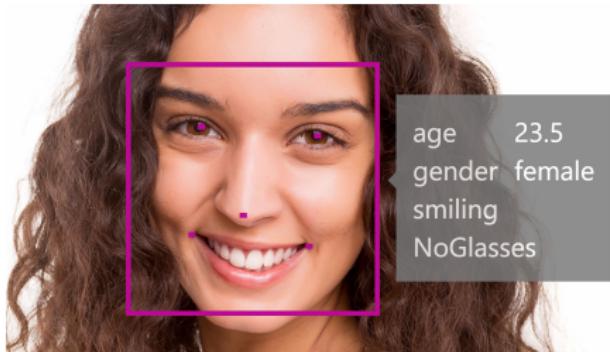
Bayesian Data Analysis, Second Edition (Texts in Computer Science) by Andrew Gelman and John Carlin
★★★★★ (10) \$56.20



Data Analysis Using Regression and Multilevel Methods by Andrew Gelman and Jennifer Hill
★★★★★ (13) \$39.59

The screenshot shows a Spotify interface for a user named Moorissa Tjokro. On the left, there's a sidebar with navigation links like 'Browse', 'Radio', 'YOUR MUSIC', 'Your Daily Mix', 'Songs', 'Albums', 'Artists', 'Stations', 'Local Files', 'PLAYLISTS', and a 'Discover Weekly...' button. The main area features a 'Discover Weekly' section with a photo of Moorissa and a description: 'MADE FOR MOORISSA Discover Weekly Your weekly mixtape of fresh music. Enjoy new discoveries and deep cuts chosen just for you. Updated every Monday, so save your favourites!' Below this, a track is playing: 'Made for Moorissa Tjokro by Spotify • 30 songs, 2 hr 6 min'. There are buttons for 'PAUSE', 'FOLLOWING', and a three-dot menu. At the bottom, there's a 'Filter' dropdown, a 'Download' toggle, and a list of tracks: 'Home (feat. Jeremy Camp)', 'Adam Cappa, Jeremy C...', '2 days ago', and '...'. To the right, there's a 'FIND FRIENDS' button and a 'See what your friends are playing' section showing friends like Julia Eger and Ben Khan.

Customer Profiling

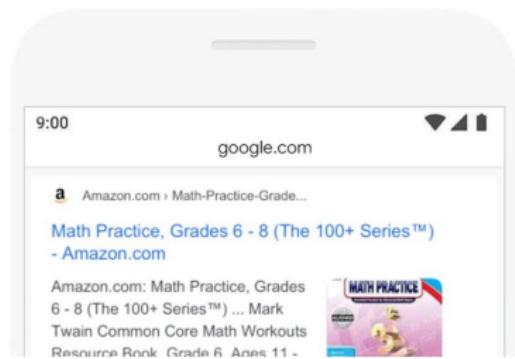


Intention Identification

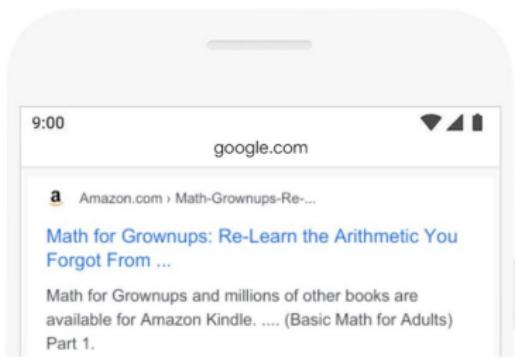


math practice books for adults

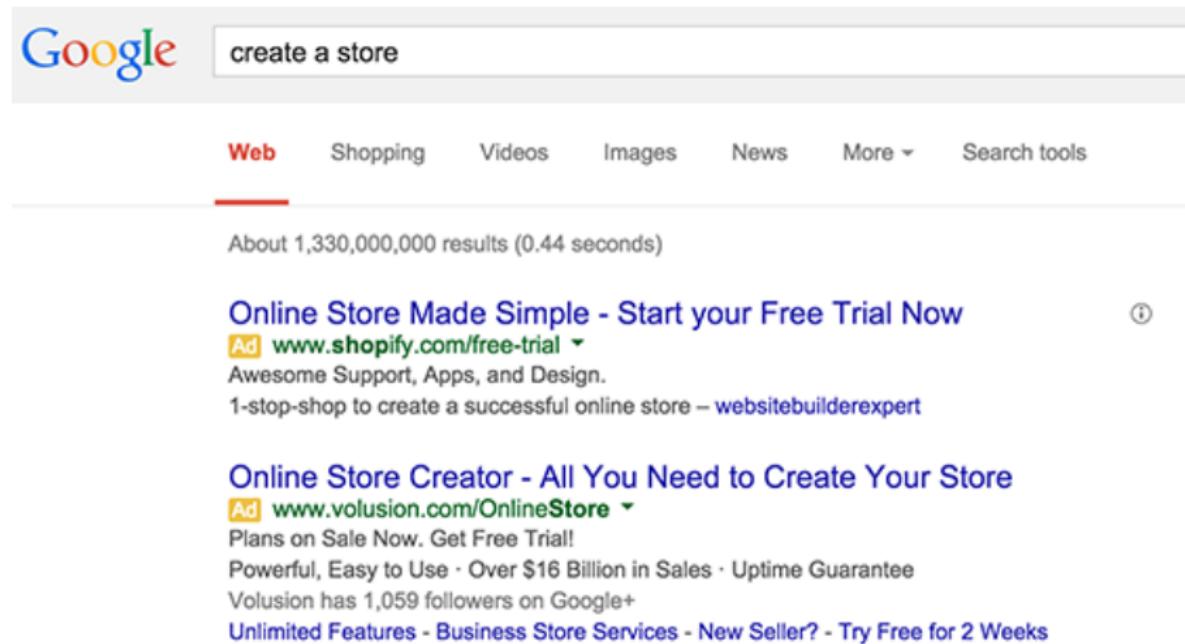
BEFORE



AFTER



Marketing & Advertisement



A screenshot of a Google search results page. The search query "create a store" is entered in the search bar. The results are filtered by "Web". Below the search bar, it says "About 1,330,000,000 results (0.44 seconds)". The first result is an advertisement for Shopify, titled "Online Store Made Simple - Start your Free Trial Now". It includes a link to www.shopify.com/free-trial, a snippet "Awesome Support, Apps, and Design.", and a description "1-stop-shop to create a successful online store – websitebuilderexpert". The second result is another advertisement for Volusion, titled "Online Store Creator - All You Need to Create Your Store". It includes a link to www.volusion.com/OnlineStore, a snippet "Plans on Sale Now. Get Free Trial!", and a description "Powerful, Easy to Use · Over \$16 Billion in Sales · Uptime Guarantee". It also mentions "Volusion has 1,059 followers on Google+", "Unlimited Features - Business Store Services - New Seller? - Try Free for 2 Weeks", and a small "①" icon.

- And much more...

How to do it?

How to do it?

Machine Learning

or Data Mining, Deep Learning, NLP, CV, etc.

Outline

① Web/App Intelligence

② What's Machine Learning?

③ Post Toxicity Detection

Prior vs. Posteriori Knowledge

- To solve a problem, we need an algorithm
 - E.g., sorting

Prior vs. Posteriori Knowledge

- To solve a problem, we need an algorithm
 - E.g., sorting
 - ***A priori knowledge*** is enough

Prior vs. Posteriori Knowledge

- To solve a problem, we need an algorithm
 - E.g., sorting
 - ***A priori knowledge*** is enough
- For some problem, however, we do not have the a priori knowledge
 - E.g., to tell if a post is toxic or not
 - The correct answer varies in time and from site to site

Prior vs. Posteriori Knowledge

- To solve a problem, we need an algorithm
 - E.g., sorting
 - ***A priori knowledge*** is enough
- For some problem, however, we do not have the a priori knowledge
 - E.g., to tell if a post is toxic or not
 - The correct answer varies in time and from site to site
- Machine learning algorithms use the ***a posteriori knowledge*** to solve problems

Prior vs. Posteriori Knowledge

- To solve a problem, we need an algorithm
 - E.g., sorting
 - ***A priori knowledge*** is enough
- For some problem, however, we do not have the ***a priori knowledge***
 - E.g., to tell if a post is toxic or not
 - The correct answer varies in time and from site to site
- Machine learning algorithms use the ***a posteriori knowledge*** to solve problems
 - Takes ***examples*** as extra input

Example Data \mathbb{X} as Extra Input

- Unsupervised:

$$\mathbb{X} = \{\mathbf{x}^{(i)}\}_{i=1}^N, \text{ where } \mathbf{x}^{(i)} \in \mathbb{R}^D$$

- E.g., $\mathbf{x}^{(i)}$ a post

Example Data \mathbb{X} as Extra Input

- Unsupervised:

$$\mathbb{X} = \{\mathbf{x}^{(i)}\}_{i=1}^N, \text{ where } \mathbf{x}^{(i)} \in \mathbb{R}^D$$

- E.g., $\mathbf{x}^{(i)}$ a post

- Supervised:

$$\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^N, \text{ where } \mathbf{x}^{(i)} \in \mathbb{R}^D \text{ and } \mathbf{y}^{(i)} \in \mathbb{R}^K,$$

- E.g., label $y^{(i)} \in \{0, 1\}$ indicates if the post $\mathbf{x}^{(i)}$ is toxic

3 General Types of Learning (1/2)

- **Supervised learning**: learn to predict the labels of future data points

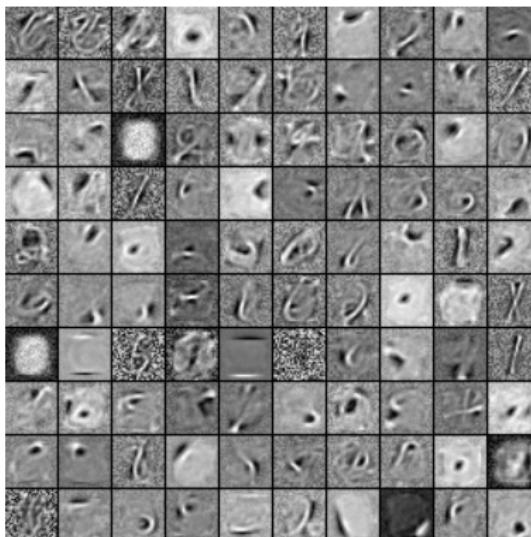
$$X \in \mathbb{R}^{N \times D} : \begin{matrix} 6 \\ 1 \\ 9 \\ 4 \\ 2 \end{matrix} \quad x' \in \mathbb{R}^D : \begin{matrix} 5 \end{matrix}$$
$$y \in \mathbb{R}^{N \times K} : [e^{(6)}, e^{(1)}, e^{(9)}, e^{(4)}, e^{(2)}] \quad y' \in \mathbb{R}^K : ?$$

3 General Types of Learning (1/2)

- **Supervised learning**: learn to predict the labels of future data points

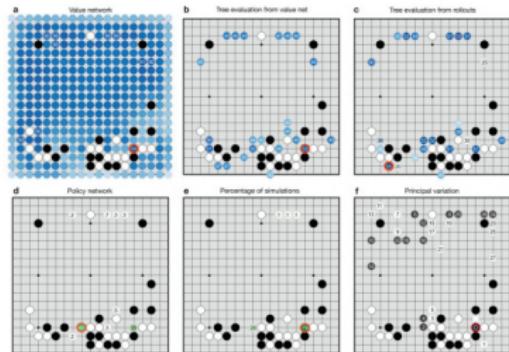
$$X \in \mathbb{R}^{N \times D} : \begin{matrix} 6 & 1 & 9 & 4 & 2 \end{matrix} \quad x' \in \mathbb{R}^D : \begin{matrix} 5 \end{matrix}$$
$$y \in \mathbb{R}^{N \times K} : [e^{(6)}, e^{(1)}, e^{(9)}, e^{(4)}, e^{(2)}] \quad y' \in \mathbb{R}^K : ?$$

- **Unsupervised learning**: learn patterns or latent factors in X



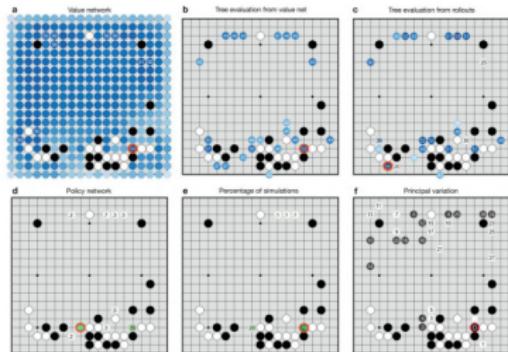
3 General Types of Learning (2/2)

- **Reinforcement learning:** learn from “good”/“bad” feedback of actions (instead of correct labels) to maximize the goal



3 General Types of Learning (2/2)

- **Reinforcement learning:** learn from “good”/“bad” feedback of actions (instead of correct labels) to maximize the goal



- AlphaGo is a hybrid of reinforcement learning and supervised learning
 - Supervised learning from the game records
 - Then, reinforcement learning from self-play

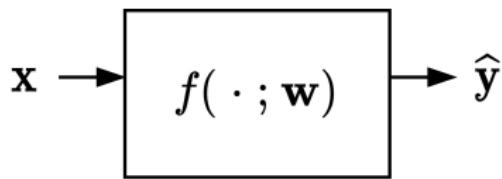
Supervised ML Step 1: Data Pre-processing

- ① Data collection and exploration
- ② Data preprocessing (e.g., integration, cleaning, etc.)



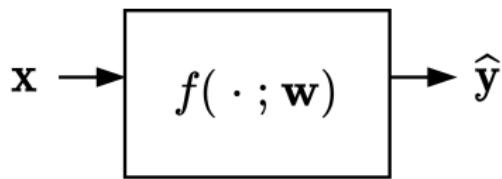
$$\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^N$$

Supervised ML Step 2: Model Development



- ① Assume a **model** $\{f(\cdot; \mathbf{w})\}_{\mathbf{w}}$ that is a collection of candidate functions f 's
 - Each f predicts label \hat{y} given an input x
 - f is assumed to be parametrized by \mathbf{w}

Supervised ML Step 2: Model Development

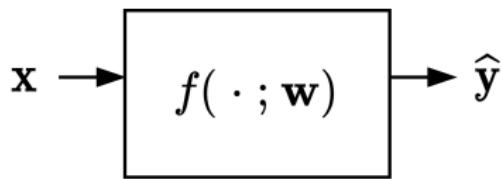


- ① Assume a **model** $\{f(\cdot; \mathbf{w})\}_{\mathbf{w}}$ that is a collection of candidate functions f 's
 - Each f predicts label \hat{y} given an input x
 - f is assumed to be parametrized by \mathbf{w}
- ② Define a **cost function**

$$C(\mathbf{w}; \mathbb{X})$$

that measures “how good a particular $f(\cdot; \mathbf{w})$ can explain the training data \mathbb{X} ” (posteriori knowledge)

Supervised ML Step 2: Model Development



- ① Assume a **model** $\{f(\cdot; \mathbf{w})\}_{\mathbf{w}}$ that is a collection of candidate functions f 's

- Each f predicts label \hat{y} given an input x
- f is assumed to be parametrized by \mathbf{w}

- ② Define a **cost function**

$$C(\mathbf{w}; \mathbb{X})$$

that measures “how good a particular $f(\cdot; \mathbf{w})$ can explain the training data \mathbb{X} ” (posteriori knowledge)

- ③ **Training**: employ an algorithm that solves

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} C(\mathbf{w}; \mathbb{X})$$

Supervised ML Step 3: Testing & Deployment

- ① **Testing**: evaluate the performance of the learned $f(\cdot; \mathbf{w}^*)$ using another, *unseen* test dataset \mathbb{X}'
 - Examples in \mathbb{X}' should have the same distribution with those in \mathbb{X}

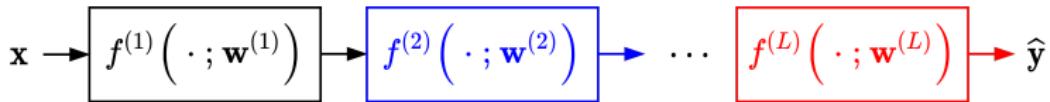
Supervised ML Step 3: Testing & Deployment

- ① **Testing**: evaluate the performance of the learned $f(\cdot; \mathbf{w}^*)$ using another, *unseen* test dataset \mathbb{X}'
 - Examples in \mathbb{X}' should have the same distribution with those in \mathbb{X}
- ② If $f(\cdot; \mathbf{w}^*)$ has a good test performance, deploy it in a real world system

What is Deep Learning?

- ML where an $f(\cdot; \mathbf{w})$ has many (deep) layers

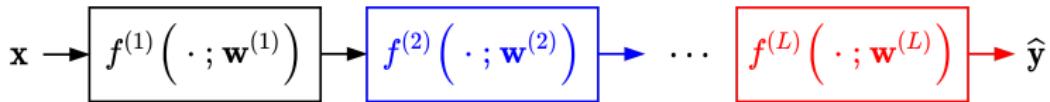
$$\hat{\mathbf{y}} = f^{(L)}(\dots f^{(2)}(f^{(1)}(\mathbf{x}; \mathbf{w}^{(1)}); \mathbf{w}^{(2)}) \dots; \mathbf{w}^{(L)})$$



What is Deep Learning?

- ML where an $f(\cdot; \mathbf{w})$ has many (deep) layers

$$\hat{\mathbf{y}} = f^{(L)}(\dots f^{(2)}(f^{(1)}(\mathbf{x}; \mathbf{w}^{(1)}); \mathbf{w}^{(2)}) \dots; \mathbf{w}^{(L)})$$



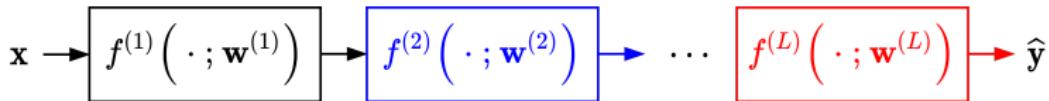
- Pros:

- Learns to pre-process data automatically
- Learns a complex function (e.g., visual objects to labels)

What is Deep Learning?

- ML where an $f(\cdot; \mathbf{w})$ has many (deep) layers

$$\hat{\mathbf{y}} = f^{(L)}(\dots f^{(2)}(f^{(1)}(\mathbf{x}; \mathbf{w}^{(1)}); \mathbf{w}^{(2)}) \dots; \mathbf{w}^{(L)})$$



- Pros:

- Learns to pre-process data automatically
- Learns a complex function (e.g., visual objects to labels)

- Cons:

- Usually needs large data to train a model well
- High computation costs (at both training and test time)

Outline

① Web/App Intelligence

② What's Machine Learning?

③ Post Toxicity Detection

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

¹ $1(\text{condition}) = 1$ if condition is true; otherwise 0.

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

- ② Model development

- ① **Model:** $\{f : f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}\}_{\mathbf{w}}$

¹ $1(\text{condition}) = 1$ if condition is true; otherwise 0.

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

- ② Model development

- ① **Model:** $\{f : f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}\}_{\mathbf{w}}$
- ② **Cost function:** $C(\mathbf{w}; \mathbb{X}) = \sum_{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \in \mathbb{X}} \mathbf{1}(f(\mathbf{x}^{(i)}; \mathbf{w}) \neq \mathbf{y}^{(i)})^1$

¹ $\mathbf{1}(\text{condition}) = 1$ if condition is true; otherwise 0.

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

- ② Model development

- ① **Model**: $\{f : f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}\}_{\mathbf{w}}$
- ② **Cost function**: $C(\mathbf{w}; \mathbb{X}) = \sum_{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \in \mathbb{X}} \mathbf{1}(f(\mathbf{x}^{(i)}; \mathbf{w}) \neq \mathbf{y}^{(i)})^1$
- ③ **Training**: to solve $\mathbf{w}^* = \arg \min_{\mathbf{w}} C(\mathbf{w}; \mathbb{X})$

¹ $\mathbf{1}(\text{condition}) = 1$ if condition is true; otherwise 0.

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

- ② Model development

- ① **Model**: $\{f : f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}\}_{\mathbf{w}}$
 - ② **Cost function**: $C(\mathbf{w}; \mathbb{X}) = \sum_{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \in \mathbb{X}} 1(f(\mathbf{x}^{(i)}; \mathbf{w}) \neq \mathbf{y}^{(i)})^1$
 - ③ **Training**: to solve $\mathbf{w}^* = \arg \min_{\mathbf{w}} C(\mathbf{w}; \mathbb{X})$
- ④ **Testing**: accuracy $\frac{1}{|\mathbb{X}'|} \sum_{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)}) \in \mathbb{X}'} 1(f(\mathbf{x}'^{(i)}; \mathbf{w}^*) = \mathbf{y}'^{(i)})$

¹ $1(\text{condition}) = 1$ if condition is true; otherwise 0.

Supervised Learning

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

- ① Get and preprocess a dataset, e.g., [civil comments](#)

- ① Training dataset: $\mathbb{X} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_i$
- ② Testing dataset: $\mathbb{X}' = \{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)})\}_i$

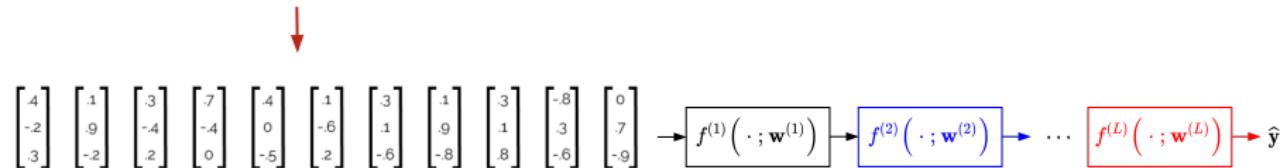
- ② Model development

- ① **Model**: $\{f : f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}\}_{\mathbf{w}}$
 - ② **Cost function**: $C(\mathbf{w}; \mathbb{X}) = \sum_{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) \in \mathbb{X}} 1(f(\mathbf{x}^{(i)}; \mathbf{w}) \neq \mathbf{y}^{(i)})^1$
 - ③ **Training**: to solve $\mathbf{w}^* = \arg \min_{\mathbf{w}} C(\mathbf{w}; \mathbb{X})$
- ③ **Testing**: accuracy $\frac{1}{|\mathbb{X}'|} \sum_{(\mathbf{x}'^{(i)}, \mathbf{y}'^{(i)}) \in \mathbb{X}'} 1(f(\mathbf{x}'^{(i)}; \mathbf{w}^*) = \mathbf{y}'^{(i)})$
- ① Then, integrate $f(\cdot; \mathbf{w}^*)$ into WeatherMood

¹ $1(\text{condition}) = 1$ if condition is true; otherwise 0.

Pre-trained Toxicity Classifier

The chef who ran to the store was out of food



- [Google's Pre-trained Toxicity Classifier on GitHub](#)
 - It's free
- Deep model:
 - ① Transforms each word into a fixed-length vector
 - ② Sums then normalizes the word vectors
 - ③ Feeds the sum into a deep classification model

Using the Pre-trained Toxicity Classifier

```
// installation
$ npm install @tensorflow/tfjs @tensorflow-models/toxicity

// usage in code
const toxicity = require('@tensorflow-models/toxicity');
const model = await toxicity.load(0.9); // threshold
const inputs = [ 'We're dudes on computers, moron...'];
const classes = await model.classify(inputs);
inputs.forEach((text, i) => {
  console.log(text);
  classes.forEach(cls => {
    console.log(cls.label, cls.results[i].match);
  });
});
```

text	identity attack	insult	obscene	severe toxicity	sexual explicit	threat	toxicity
We're dudes on computers, moron. You are quite astonishingly stupid.	false	true	false	false	false	false	true

About Your Final Project...

- Implement the prototype shown in the midterm design pitch
- Final project demo:
 - On 6/30
 - 4 min for team (strict)
 - 30 sec for QA

About Your Final Project...

- Implement the prototype shown in the midterm design pitch
- Final project demo:
 - On 6/30
 - 4 min for team (strict)
 - 30 sec for QA
- Evaluation:
 - Completeness (60%)
 - Complexity (20%)
 - Design (20%)

Completeness (60%)

- How many main features have you completed?
- Demo each of them
- How well does your final implementation match your mid-term design?
(10%)

Complexity (20%)

- Explain *one or two* most
 - challenging features you implemented, or
 - difficult issues you resolved

Design (20%)

- UX (user experience)
- UI (user interface)
- Animation effect
- Overall design

Bonus

- Best Minimal Viable Products (MVPs)
 - **+30%**, **+20%**, and **+10%** for #1, #2, and #3, respectively
- Peer review
 - Each team has three non-self votes
 - Judged by completeness, complexity, and design

