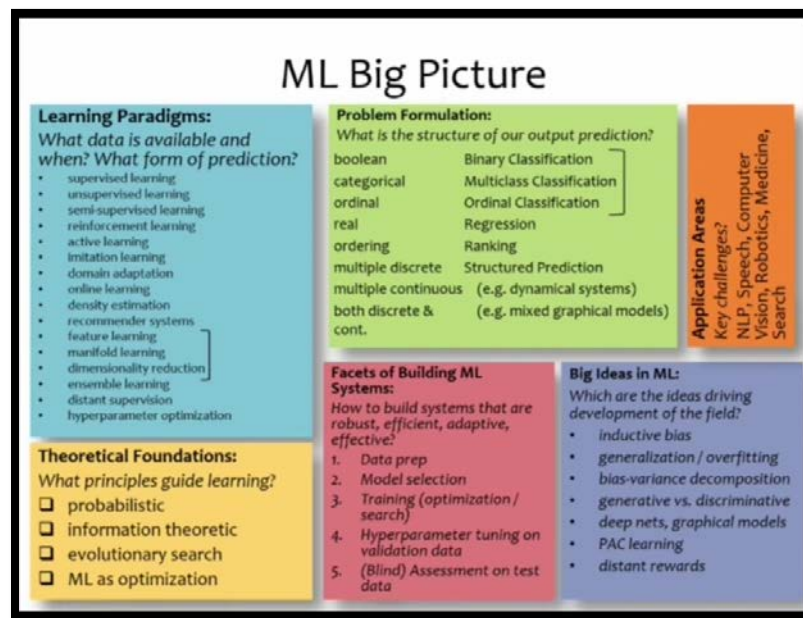


كلها انتردكشن .. لما هيجي حاجه مفيده هبدأ أكتب .. المواضيع اللي معانا ..

Topics	
<ul style="list-style-type: none">• Foundations<ul style="list-style-type: none">– Probability– MLE, MAP– Optimization• Classifiers<ul style="list-style-type: none">– KNN– Naive Bayes– Logistic Regression– Perceptron– SVM• Regression<ul style="list-style-type: none">– Linear Regression• Important Concepts<ul style="list-style-type: none">– Kernels– Regularization and Overfitting– Experimental Design• Unsupervised Learning<ul style="list-style-type: none">– K-means / Lloyd's method– PCA– EM / GMMs	<ul style="list-style-type: none">• Neural Networks<ul style="list-style-type: none">– Feedforward Neural Nets– Basic architectures– Backpropagation– CNNs, LSTMs• Graphical Models<ul style="list-style-type: none">– Bayesian Networks– HMMs– Learning and Inference• Learning Theory<ul style="list-style-type: none">– Statistical Estimation (covered right before midterm)– PAC Learning• Other Learning Paradigms<ul style="list-style-type: none">– Matrix Factorization– Reinforcement Learning– Information Theory

بس انت هتعوز تركز علي الصورة الكبيره .. اللي هي ديه



انهارده احنا هنسيب الصورة الكبيره شويه وبعدين نتحرك ونرجعلها تاني .. اول موضوع هنتكلم عنه هو ان عندنا شوية كلاسيفاييرز مختلفين ...

انهارده بقي هنعمل ايه .. هن Define learning problems ... معظم الكورس هيبقي ازاي اصلا نحل المشاكل ... بس دلوقت احنا هنعرف ايه هي المشاكل ...

Well-Posed Learning Problems

Three components $\langle T, P, E \rangle$:


1. Task, T
2. Performance measure, P
3. Experience, E

Definition of learning:

A computer program **learns** if its performance at tasks in T , as measured by P , improves with experience E .

لو خدت كورس أندرو فهتبقا فاكرا الديفيشن ده .. بيقولك ان الكمبيوتر بيتعلم لو البرفورمنس بتاعه في التاسك اللي اسمو T .. وانت بتقيس البرفورمنس ده برقم و الرقم ده بيعبر عن برفورمنس P ... و البرفورمنس بيتحسن مع ال Experience .. بمعنى انت مش بس عاوز تعرف تمااب الداتا اللي انت هتشتغل منها .. وليه الخبره هنا؟ .. عشان انت مش عاوز بس تعرف هي ايه الداتا الي معاك .. انت عاوز تعرف الداتا نفسها فين .. جايه منين .. تعال ناخذ مثال .. لازم البرفورمنس ميجرمنت بيقا aligned مع المشكله اللي عندنا ... من ضمن الطرق ان ممكن حاجه تبقي قيمتها أكبر من حاجه تانيه .. فممكن تضحي بحاجه مقابل حاجه ... بالنسبة للخبره .. انت تقدر تخلي الكمبيوتر يلعب شويه مع نفسه .. فأول حل كان انك تشوف الإكسبيرتس بيقولو ايه وتبدأ تحط رولز .. بص عالصورتين اللي جايين

Capturing the Knowledge of Experts



Solution #1: Expert Systems

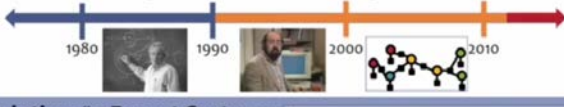
- Over 20 years ago, we had rule based systems
- Ask the expert to
 1. Obtain a PhD in Linguistics
 2. Introspect about the structure of their native language
 3. Write down the rules they devise

Give me directions to Starbucks
If: "give me directions to X"
Then: directions(here, nearest(X))

How do I get to Starbucks?
If: "how do i get to X"
Then: directions(here, nearest(X))

Where is the nearest Starbucks?
If: "where is the nearest X"
Then: directions(here, nearest(X))

Capturing the Knowledge of Experts



Solution #1: Expert Systems

- Over 20 years ago, we had rule based systems
- Ask the expert to
 1. Obtain a PhD in Linguistics
 2. Introspect about the structure of their native language
 3. Write down the rules they devise

I need directions to Starbucks
If: "I need directions to X"
Then: directions(here, nearest(X))

Starbucks directions
If: "X directions"
Then: directions(here, nearest(X))

Is there a Starbucks nearby?
If: "Is there an X nearby"
Then: directions(here, nearest(X))

بس في الآخر هتلاقي عندك كمية كبيره جداً من الروولز .. عامله ريستركشنز عاليه أوي علي حاجات كتير ...
 في حل تاني .. ما تجيب الإكسبريتس نفسهم .. خد شوية داتا و خليه يقولوك مين الصح و مين الغلط او انه يرد
 هنا وانه يرد هنا .. فانت هنا مش محتاج رولز انت محتاج رد بس من الإكسبريتس ..

Capturing the Knowledge of Experts

Solution #2: Annotate Data and Learn

1. Collect raw sentences $\{x_1, \dots, x_n\}$
2. Experts annotate their meaning $\{y_1, \dots, y_n\}$

x_1 : How do I get to Starbucks?	x_3 : Send a text to John that I'll be late
y_1 : directions(here, nearest(Starbucks))	y_3 : txtmsg(John, I'll be late)
x_2 : Show me the closest Starbucks	x_4 : Set an alarm for seven in the morning
y_2 : map(nearest(Starbucks))	

be late

تعال نبص علي التاسك ده: ...

Example Learning Problems

Learning to **respond to voice commands (Siri)**

1. Task, T :
 → predicting action from speech
2. Performance measure, P :
 → percent of correct actions taken in user pilot study
3. Experience, E :
 → examples of (speech, action) pairs