Introduction to Machine Learning

Stephen Clark
(based heavily on slides from Mark Gales)

Lent 2013

Machine Learning for Language Processing: Lecture 1

MPhil in Advanced Computer Science

Decision Making

In this world nothing can be said to be certain, except death and taxes.

- Benjamin Franklin

- We make decisions under uncertainty all the time

  gambling (not recommended), weather forecasting (not very successfully)
  insurance (risk assessment), stock market

Need to formalise "intuitive decisions" mathematically

- Basically, how to quantify and manipulate uncertainty.
- Various tasks we can consider
  - classification: predict class from observations
  - regression (prediction): predict value from observations
  - clustering: group observations into "meaningful" groups

Machine Learning

- One definition is (Mitchell):

  "A computer program is said to learn from experience (E) with some class of tasks (T) and a performance measure (P) if its performance at tasks in T as measured by P improves with E"

  alternatively

  "Systems built by analysing data sets rather than by using the intuition of experts"

- Multiple specific conferences:
  - {International,European} Conference on Machine Learning;
  - Neural Information Processing Systems;
  - International Conference on Pattern Recognition etc etc;
- as well as sessions in other conferences:
  - ICASSP - machine learning for signal processing.

Natural Language Processing Applications

- Many possible applications:
  - spam email detection;
  - named-entity recognition;
  - machine translation;
  - relation extraction;
  - information retrieval;
  - sentiment analysis.
- Generally need to structure and annotate vast quantities of text data
  - sometimes used in combination with speech and image processing

Machine Translation

Rafales de marque - lecteur dans la technologie de... http://66.249.91.104/translate_c?hl=en&langpai...

UNIVERSITY OF CAMBRIDGE DEPARTMENT OF ENGINEERING

Marquer les rafales

Les rafales de marque est un lecteur dans la technologie de l'information dans le laboratoire d'intelligence de machine (autrefois le groupe de vision et de robotique de la parole (SVR)) et un camarade de l'université d'Emmanuel. Il est un membre du groupe de recherche de la parole ainsi que les jeunes de Steve de membres de personnel de corps enseignant, la régfion boisée et la facture Byrne de Phil.

Une brève biographie est accessible en ligne.

[Recherche | projets | publications | étudiants | enseignant | contact]

Intérêts de recherches

- Reconnaissance de la parole continue de grand vocabulaire
- Reconnaissance de la parole robuste
- Adaptation d'orateur
- Étude de machine (en particulier choix modèle et méthodes grain-basées)
- Identification et vérification d'orateur

Une brève introduction à la reconnaissance de la parole est accessible en ligne.
dessus

Projets de recherche

Projets en cours :

- Bruit ASR robuste (Europe Ltd de recherches de Toshiba placée)
- Traitement averti d'environnement rapide et robuste (Europe Ltd de recherches de Toshiba placée)
  - NEW Position d'associé de recherches disponible
- AGILE (projet placé par GALE de DARPA)
- Version 3 de HTK - HTK_V3.4 et exemples sont disponibles.

Projets récemment réalisés :

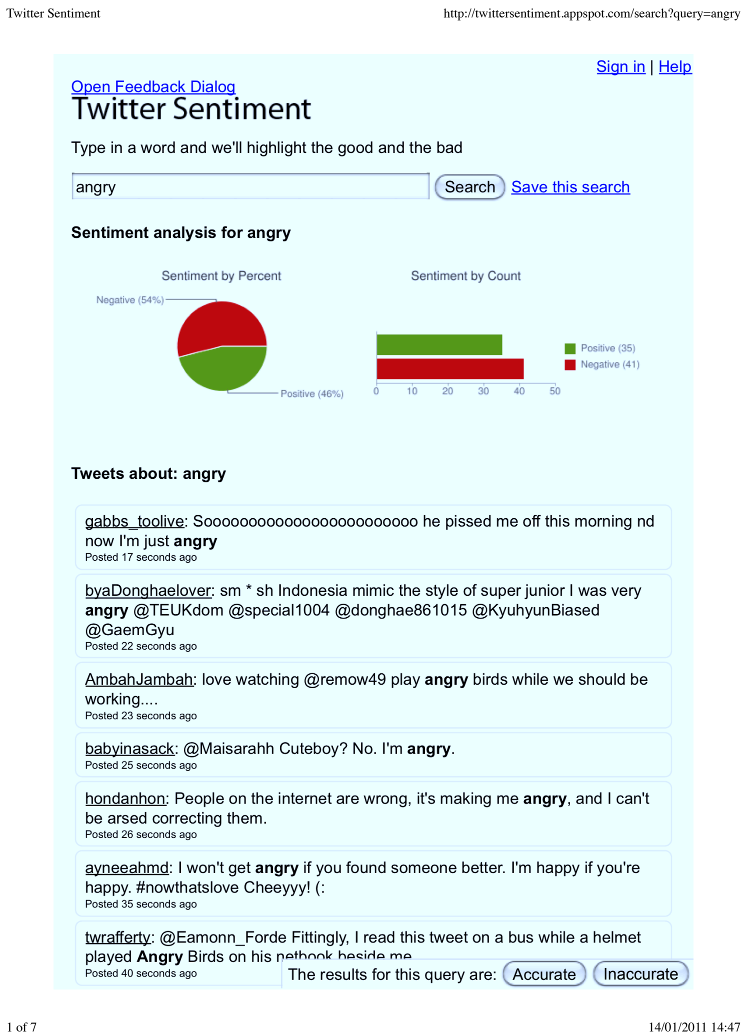
- CoreTex (améliorant la technologie de reconnaissance de la parole de noyau)
- Transcription audio riche de HTK(Projet placé par OREILLES de DARPA) - pages Web locaux

dessus

1 of 3 17/09/07 15:08

- Statistical approaches work well
- Part of this MPhil course.

Sentiment Analysis



- Statistical approaches again work well
- Lots of interest in this from companies, esp. with the advent of social media

Natural Language Processing

Why is natural language processing an interesting machine learning task?

- "Standard" machine learning tasks are of the form
  - clearly defined set of observations $\boldsymbol{x}$
  - "reasonable" number of classes $\omega_1, \ldots, \omega_K$

- Consider statistical machine translation with source vocabulary $V_{\rm s}$ target vocabulary $V_{\rm t}$
  - for target sentence of 10 words $V_{\rm t}^{10}$ possible sentences
  - $V_{\rm s}$ word features, $V_{\rm s}^2$ word-pair features, $V_{\rm s}^3$ word-tuple features, ...
  - vast number of possible classes, vast number of possible features

- The first 2 lectures on classification will not address these problems directly
  - standard machine learning described
  - language processing extensions will be described in future lectures

Basic Discrete Probability

- Discrete random variable x takes one value from the set, with probabilities

$$\mathcal{X} = \omega_1, \ldots, \omega_K; \quad p_j = \Pr(x = \omega_j), \quad j = 1, \ldots, K$$

Probability mass function, $P(x)$, describes the set of probabilities, satisfies

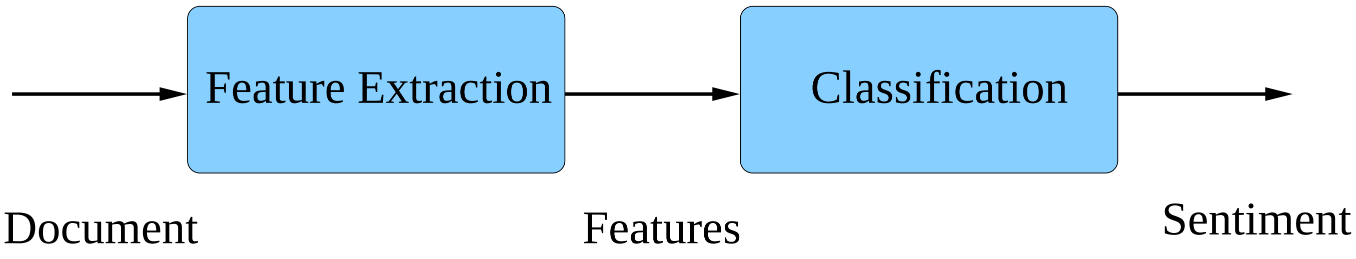
$$\sum_{x \in \mathcal{X}} P(x) = 1, \quad P(x) \geq 0$$

Probability density function, $p(x)$, equivalent for continuous random variables

- For random variables x, y, z need

conditional distribution: $P\left(x|y\right) = \frac{P(x,y)}{P(y)}$
joint distribution $P\left(x,\, y\right)$
marginal distribution $P\left(x\right) = \sum_{y \in \mathcal{Y}} P(x, y)$
chain rule $P(x, y, z) = P(x|y, z)\; P(y|z)\; P(z)$

Machine Learning Framework



- There are two stages in a pattern recognition framework:
  - feature extraction: a feature vector, $\boldsymbol{x}$, is derived from the "observations";
  - classification: a class ω is identified given the feature vector $\boldsymbol{x}$.
- Example: sentiment analysis
  - $\boldsymbol{w}$ is the document (words)
  - $\boldsymbol{x}$ is a binary vector indicating whether a particular word is in the document
  - ω is the sentiment (e.g. angry)
- Need to design a suitable feature vector and classifier for the task in hand.

Training and Evaluation Data

- The basic machine learning framework has two sets of data:

  1. Training data: is used to train the classifier - data may be:
     - supervised: the correct classes of the training data are known
     - unsupervised: the correct classes of the training data are not known
     - reinforcement learning: don't learn a model - directly learn an action!
  2. Test data: held-out data for evaluating the classifier

  Supervised training data will be mostly considered in this course

- It is important that the training and test data do not overlap
  - performance on training data better than on held-out data
  - becomes more important as the classifiers become more complex
  - development data sometimes used to tune parameters

- Aim to build a classifier that performs well on held-out data; generalise.

Machine Learning-Based Decisions

- Consider a system where
  - observation: feature vector of dimension d, $\boldsymbol{x}$
  - class labels: there are K classes, denoted by $\omega_1,\ \omega_2, ...,\ \omega_K$.
- Classifiers for making decisions can be broadly split as:
  - Generative models: a model of the joint distribution of observations and classes is trained, $P(\boldsymbol{x}, \omega_j)$.
  - Discriminative models: a model of the posterior distribution of the class given the observation is trained, $P(\omega_j|\boldsymbol{x})$.
  - Discriminant functions: a mapping from an observation $\boldsymbol{x}$ to class ω_j is directly trained. No posterior probability, $P(\omega_j|\boldsymbol{x})$, generated just class labels.

Generative Models

- For generative models the joint distribution is found - often expressed as

$$P(\boldsymbol{x}, \omega_j) = P(\boldsymbol{x}|\omega_j)P(\omega_j)$$

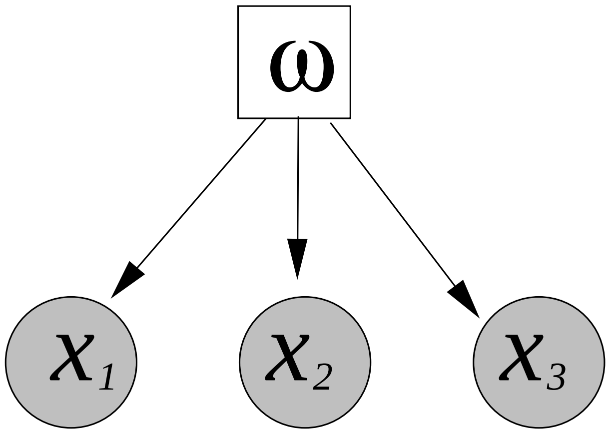
- Form of classifier considered has two parts
  - prior probabilities: an idea of how frequent each class is, $P(\omega_1), \ldots, P(\omega_K)$.
  - class-conditional (likelihood) probability: the probability of the feature vector for each class $P(\boldsymbol{x}|\omega_1), \ldots, P(\boldsymbol{x}|\omega_K)$.

- For an unknown observation $\boldsymbol{x}$, Bayes' rule allows the calculation of posterior probability of class membership.

$$P(\omega_j|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|\omega_j)P(\omega_j)}{\sum_{k=1}^{K} P(\boldsymbol{x}|\omega_k)\; P(\omega_k)}, \quad j = 1, 2, ..., K$$

Naive Bayes' Classifier

- Simple form of generative model:
  - joint distribution: $P(\boldsymbol{x}, \omega_j) = P(\omega_j) \prod_{i=1}^{d} P(x_i|\omega_j)$
  - classification: $P(\omega_j|\boldsymbol{x}) \propto P(\omega_j) \prod_{i=1}^{d} P(x_i|\omega_j)$
- Elements of the feature vector conditionally independent given the class



- write as a Bayesian Network (BN)
  - shaded observed variable
  - unshaded unobserved variable
  - circle continuous variable
  - square discrete variable
- More on Bayesian Networks (and Graphical Models) later in the module

Probability Distributions

- For generative models need to decide form of conditional distribution $P(\boldsymbol{x}|\omega_j)$
  - (d-dimensional) feature vector may be discrete or continuous

- Discrete distributions (probability mass functions) - primary interest here
  - Multivariate-Bernoulli distribution: $x_i \in \{0, 1\}$,

$$P(\boldsymbol{x}|\omega_j) = \prod_{i=1}^{d} p_{ji}^{x_i}(1-p_{ji})^{1-x_i}; \qquad 0 \leq p_{ji} \leq 1$$

  - Multinomial distribution: $x_i \in \{0, \ldots, n\}$

$$P(\boldsymbol{x}|\omega_j) = \frac{n!}{\prod_{i=1}^{d} x_i!}\prod_{i=1}^{d} p_{ji}^{x_i}, \qquad n = \sum_{i=1}^{d} x_i, \quad \sum_{i=1}^{d} p_{ji} = 1, \quad p_{ji} \geq 0$$

- Continuous distribution, $x_i \in [-\infty, \infty]$, less interest on this module

Maximum Likelihood Training

- The class-conditional distribution $P(\boldsymbol{x}|\omega_j)$ needs to be trained
  - for class ω_j with n training examples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$

$$\hat{\boldsymbol{\lambda}}_j = \operatorname*{argmax}_{\boldsymbol{\lambda}} \left\{ \prod_{\tau=1}^{n} P(\boldsymbol{x}_\tau|\boldsymbol{\lambda}) \right\} = \operatorname*{argmax}_{\boldsymbol{\lambda}} \left\{ \sum_{\tau=1}^{n} \log\left(P(\boldsymbol{x}_\tau|\boldsymbol{\lambda})\right) \right\}$$

- For the multivariate Bernoulli distribution: $\boldsymbol{\lambda}_j = \{p_{j1}, \ldots, p_{jd}\}$

$$\hat{\boldsymbol{\lambda}}_j = \operatorname*{argmax}_{\boldsymbol{\lambda}_j} \left\{ \sum_{\tau=1}^{n} \sum_{i=1}^{d} x_{\tau i} \log(p_{ji}) + (1 - x_{\tau i}) \log(1 - p_{ji}) \right\}$$

Differentiating wrt $\boldsymbol{\lambda}_j$ and equating to zero yields: $p_{ji} = \frac{1}{n} \sum_{\tau=1}^{n} x_{\tau i}$

Improving the Basic Model

- Incorporating a Prior: What happens if a count is zero?
  - simplest solution to initialise counts with a constant α: for Bernoulli

$$p_{ji} = \frac{1}{\alpha + n}\left(\alpha + \sum_{\tau=1}^{n} x_{\tau i}\right)$$

  - more details on this topic in discussion of language models
- Mixture Model: more "powerful" distribution combining multiple distributions:

$$P(\boldsymbol{x}|\omega_j) = \sum_{m=1}^{M} P(c_m|\omega_j)P(\boldsymbol{x}|c_m, \omega_j)$$

  - component c_m has prior, $P(c_m|\omega_j)$ and probability distribution, $P(\boldsymbol{x}|c_m, \omega_j)$
  - more details on this topic in the lectures on graphical models