Applied Natural Language Processing

Info 256

Lecture 5: Text classification (Feb 5, 2019)

David Bamman, UC Berkeley

Data

Classification

A mapping *h* from input data x (drawn from instance space $\mathcal{X}$) to a label (or labels) y from some enumerable output space $\mathcal{Y}$

$\mathcal{X}$ = set of all documents
$\mathcal{Y}$ = {english, mandarin, greek, …}

x = a single document
y = ancient greek

Classification

$$h(x) = y$$

$$h(\mu\tilde{\eta}\nu\iota\nu\ \ddot{\alpha}\epsilon\iota\delta\epsilon\ \theta\epsilon\grave{\alpha}) = \text{ancient grc}$$

Classification

Let h(x) be the “true” mapping. We never know it.
How do we find the best $\hat{h}(x)$ to approximate it?

One option: rule based

if x has characters in unicode point range 0370-03FF:
$\hat{h}(x)$ = greek

Classification

Supervised learning

Given training data in the form of <x, y> pairs, learn $\hat{h}(x)$

Text categorization problems

task	x	y
language ID	text	{english, mandarin, greek, …}
spam classification	email	{spam, not spam}
authorship attribution	text	{jk rowling, james joyce, …}
genre classification	novel	{detective, romance, gothic, …}
sentiment analysis	text	{postive, negative, neutral, mixed}

Sentiment analysis

- Document-level SA: is the entire text positive or negative (or both/neither) with respect to an implicit target?

- Movie reviews [Pang et al. 2002, Turney 2002]

Training data

positive

"… is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius"

Roger Ebert, Apocalypse Now

- "I hated this movie. Hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it."

Roger Ebert, North

negative

- Implicit signal: star ratings
- Either treat as ordinal regression problem ({1, 2, 3, 4, 5} or binarize the labels into {pos, neg}

★★★★★ **I recommend it.**
This book introduces readers to important topics in NLP. In places where it needs to go deeper it seems like it compiles information from relevant published papers and provides... Read more
Published 1 year ago by Renat Bekbolatov

★★★★☆ **It's presented easily and accessibly**
It can be dense sometimes, but it's one of the most helpful textbooks I've had for computational linguistics. Read more
Published on April 28, 2015 by Vanessa A.

★★★★★ **Five Stars**
I love this book.
It was easy to follow and a great read.
Published on December 28, 2014 by Stefan Melforth Gulbrandsen

★★★★★ **Five Stars**
I needed the book for my natural language processing class. needless to say, I learnt a lot.
Published on November 27, 2014 by Kamran

★★★☆☆ **Encyclopedic Treatment of NLP**
Daniel Jurafsky and James Martin have assembled an incredible mass of information about natural language processing.Foundations of Statistical Natural Language Processing Read more
Published on April 25, 2012 by John M. Ford

Sentiment analysis

- Is the text positive or negative (or both/ neither) with respect to an explicit target within the text?

Feature: **picture**

Positive: 12

- Overall this is a good camera with a really good picture clarity.
- The pictures are absolutely amazing - the camera captures the minutest of details.
- After nearly 800 pictures I have found that this camera takes incredible pictures.

...

Negative: 2

- The pictures come out hazy if your hands shake even for a moment during the entire process of taking a picture.
- Focusing on a display rack about 20 feet away in a brightly lit room during day time, pictures produced by this camera were blurry and in a shade of orange.

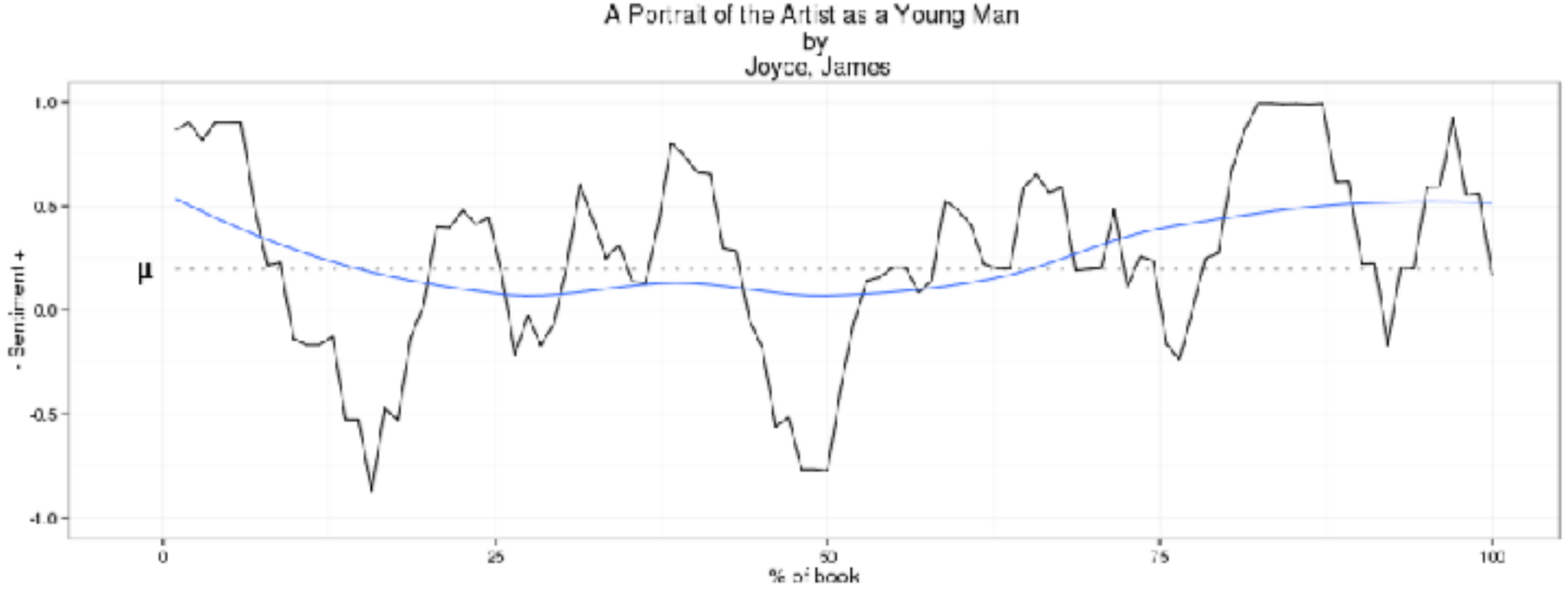
Hu and Liu (2004), “Mining and Summarizing Customer Reviews”

Sentiment as tone

- No longer the speaker's attitude with respect to some particular target, but rather the positive/negative tone that is evinced.

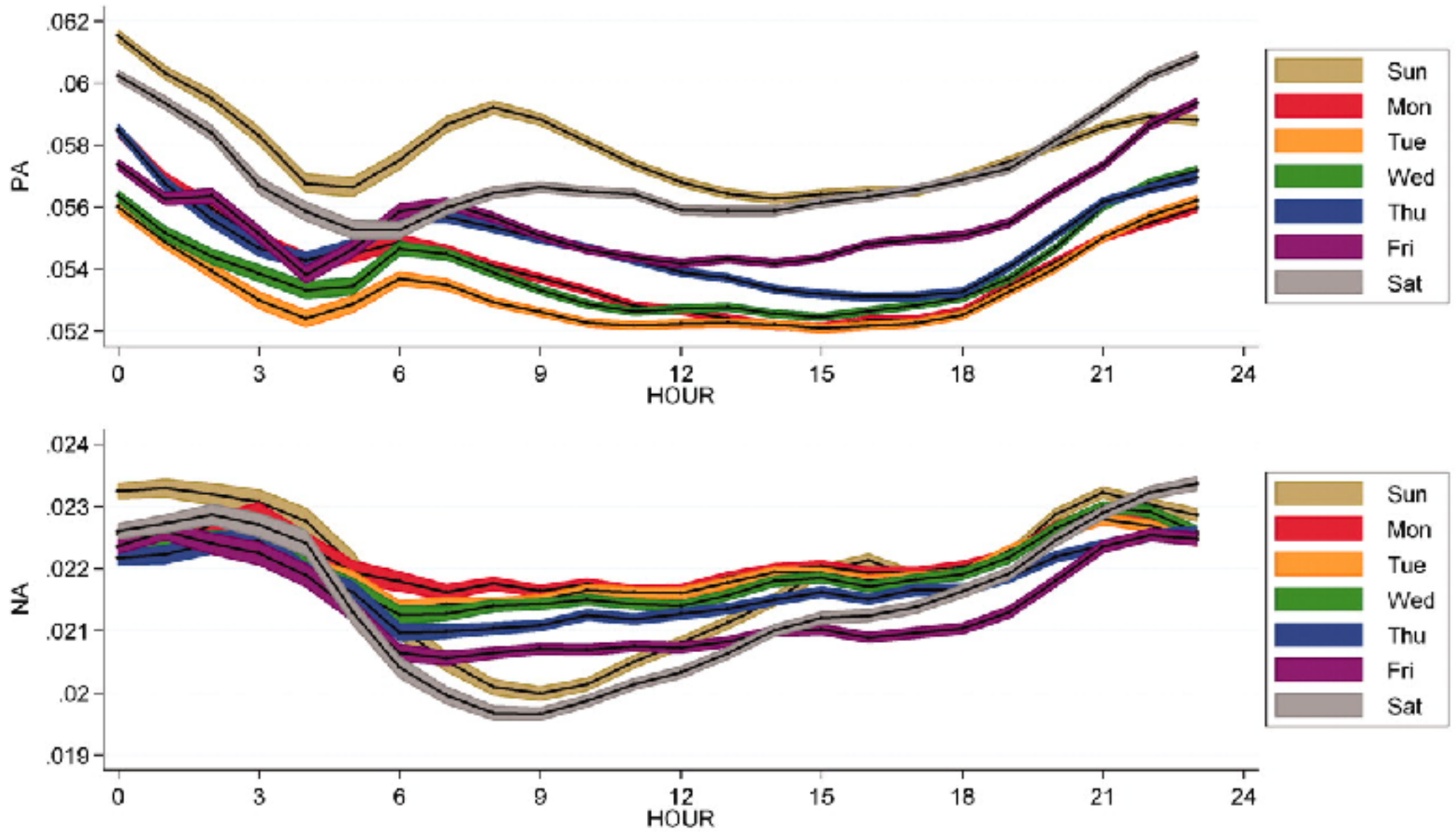
Sentiment as tone

“Once upon a time and a very good time it was there was a moocow coming down along the road and this moocow that was coming down along the road met a nicens little boy named baby tuckoo…"



http://www.matthewjockers.net/2014/06/05/a-novel-method-for-detecting-plot/

Sentiment as tone



Golder and Macy (2011), "Diurnal and Seasonal Mood Vary with Work, Sleep, and Daylength Across Diverse Cultures," *Science.* Positive affect (PA) and negative affect (NA) measured with LIWC.

Why is SA hard?

- Sentiment is a measure of a speaker's private state, which is unobservable.

- Sometimes words are a good indicator of sentence (love, amazing, hate, terrible); many times it requires deep world + contextual knowledge

> "*Valentine's Day* is being marketed as a Date Movie. I think it's more of a First-Date Movie. If your date likes it, do not date that person again. And if you like it, there may not be a second date."
>
> Roger Ebert, Valentine's Day

Classification

Supervised learning

Given training data in the form of <x, y> pairs, learn $\hat{h}(x)$

x	y
loved it!	positive
terrible movie	negative
not too shabby	positive

$\hat{h}(x)$

- The classification function that we want to learn has two different components:
  - the formal structure of the learning method (what's the relationship between the input and output?) → Naive Bayes, logistic regression, convolutional neural network, etc.
  - the representation of the data

Classification

Representation for SA

- Only positive/negative words in dictionaries (MPQA)
- Only words in isolation (bag of words)
- Conjunctions of words (sequential, skip ngrams, other non-linear combinations)
- Higher-order linguistic structure (e.g., syntax)

"… is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius"

Roger Ebert, Apocalypse Now

"I hated this movie. Hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it."

Roger Ebert, North

Bag of words

Representation of text only as the counts of words that it contains

	Apocalypse now	North
the	1	1
of	0	0
hate	0	9
genius	1	0
bravest	1	0
stupid	0	1
like	0	1
…		

Refresher

$$\sum_{i=1}^{F} x_i \beta_i = x_1 \beta_1 + x_2 \beta_2 + \ldots + x_F \beta_F$$

$$\prod_{i=1}^{F} x_i = x_i \times x_2 \times \ldots \times x_F$$

$$\exp(x) = e^x \approx 2.7^x \qquad \exp(x+y) = \exp(x)\exp(y)$$

$$\log(x) = y \rightarrow e^y = x \qquad \log(xy) = \log(x) + \log(y)$$

Logistic regression

$$P(y = 1 \mid x, \beta) = \frac{1}{1 + \exp\left(-\sum_{i=1}^{F} x_i \beta_i\right)}$$

output space $\mathcal{Y} = \{0, 1\}$

x = feature vector

Feature	Value
the	0
and	0
bravest	0
love	0
loved	0
genius	0
not	0
fruit	1
BIAS	1

β = coefficients

Feature	β
the	0.01
and	0.03
bravest	1.4
love	3.1
loved	1.2
genius	0.5
not	-3.0
fruit	-0.8
BIAS	-0.1

	BIAS	love	loved
β	-0.1	3.1	1.2

	BIAS	love	loved	$a=\sum x_i\beta_i$	exp(-a)	1/(1+exp(-a))
x^1	1	1	0	3	0.05	95.2%
x^2	1	1	1	4.2	0.015	98.5%
x^3	1	0	0	-0.1	1.11	47.4%

Features

- As a discriminative classifier, logistic regression doesn't assume features are independent (like e.g. Naive Bayes does)

- Its power partly comes in the ability to create richly expressive features with out the burden of independence.

- We can represent text through features that are not just the identities of individual words, but any feature that is scoped over the entirety of the input.

features
contains like
has word that shows up in positive sentiment dictionary
review begins with "I like"
at least 5 mentions of positive affectual verbs (like, love, etc.)

Features

- Features are where you can encode your own domain understanding of the problem.

feature classes
unigrams (“like”)
bigrams (“not like”), higher order ngrams
prefixes (words that start with “un-”)
has word that shows up in positive sentiment dictionary

Features

Task	Features
Sentiment classification	Words, presence in sentiment dictionaries, etc.
Keyword extraction	
Fake news detection	
Authorship attribution	

Features

Feature	Value
the	0
and	0
bravest	0
love	0
loved	0
genius	0
not	1
fruit	0
BIAS	1

Feature	Value
like	1
not like	1
did not like	1
in_pos_dict_MPQA	1
in_neg_dict_MPQA	0
in_pos_dict_LIWC	1
in_neg_dict_LIWC	0
author=ebert	1
author=siskel	0

How do we get good values for β?

β = coefficients

Feature	β
the	0.01
and	0.03
bravest	1.4
love	3.1
loved	1.2
genius	0.5
not	-3.0
fruit	-0.8
BIAS	-0.1

Conditional likelihood

$$\prod_{i}^{N} P(y_i \mid x_i, \beta)$$

For all training data, we want the probability of the true label y for each data point x to be high

	BIAS	love	loved	$a=\sum x_i\beta_i$	exp(-a)	1/(1+exp(-a))	true y
x^1	1	1	0	3	0.05	95.2%	1
x^2	1	1	1	4.2	0.015	98.5%	1
x^3	1	0	0	-0.1	1.11	47.5%	0

Conditional likelihood

$$\prod_{i}^{N} P(y_i \mid x_i, \beta)$$

For all training data, we want the probability of the true label y for each data point x to be high

Pick the values of parameters β to maximize the conditional probability of the training data <x, y> using gradient ascent.

Evaluation

- For all supervised problems, it's important to understand how well your model is performing

- What we try to estimate is how well you will perform in the future, on new data also drawn from $\boldsymbol{x}$

- Trouble arises when the training data <x, y> you have does not characterize the full instance space.

    - n is small
    - sampling bias in the selection of <x, y>
    - x is dependent on time
    - y is dependent on time (concept drift)

$\mathcal{X}$

instance space

labeled data

$\mathcal{X}$

instance space

train

test

Train/Test split

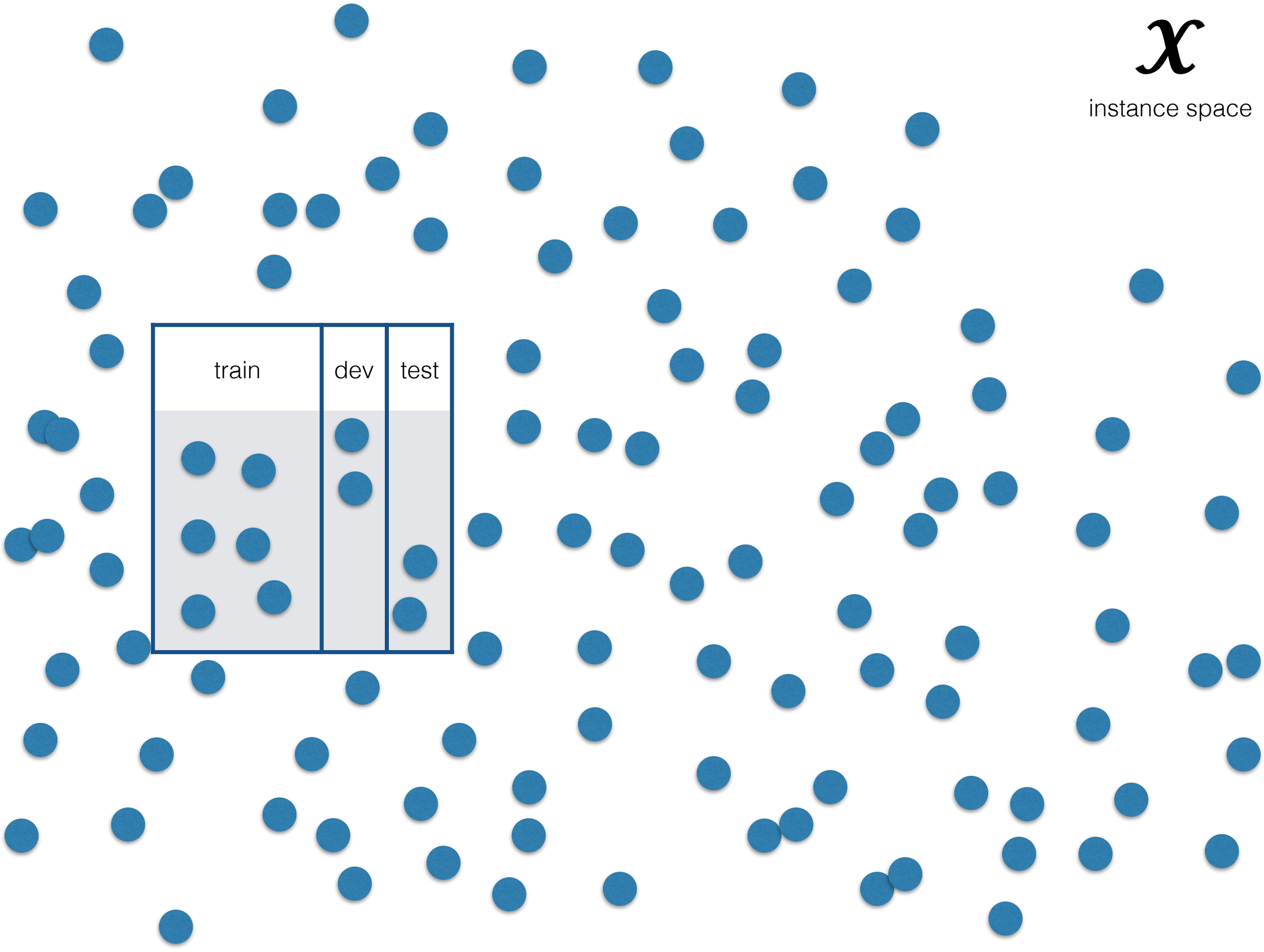
- To estimate performance on future unseen data, train a model on 80% and test that trained model on the remaining 20%
- What can go wrong here?

$\mathcal{X}$

instance space

train

test


x
instance space
train
dev
test

Experiment design

	training	development	testing
size	80%	10%	10%
purpose	training models	model selection	evaluation; never look at it until the very end

Binary classification

https://twitter.com/teenybiscuit/status/705232709220769792/photo/1

- Binary classification: $|y| = 2$

[one out of 2 labels applies to a given x]

x	y
image	{puppy, fried chicken}

Accuracy

$$\text{accuracy} = \frac{\text{number correctly predicted}}{N}$$

$$\frac{1}{N}\sum_{i=1}^{N} I[\hat{y}_i = y_i] \qquad I[x] = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

Perhaps most intuitive single statistic when the number of positive/negative instances are comparable

Majority class baseline

- Pick the label that occurs the most frequently in the training data. (Don't count the test data!)

- Predict that label for every data point in the test data.

Activity

- Implement majority class baseline for your data

- Explore the impact of hyperparameter choices on accuracy with a bag-of-words model

Parameters vs. Hyperparameters

Parameters whose values are *learned*

Feature	β
the	0.01
and	0.03
bravest	1.4
love	3.1
loved	1.2
genius	0.5
BIAS	-0.1

Hyperparameters whose values are *chosen*

Hyperparameter	value
minimum word frequency	5
max vocab size	10000
lowercase	TRUE
regularization strength	1.0