UNIVERSITY OF MANNHEIM

Web Mining

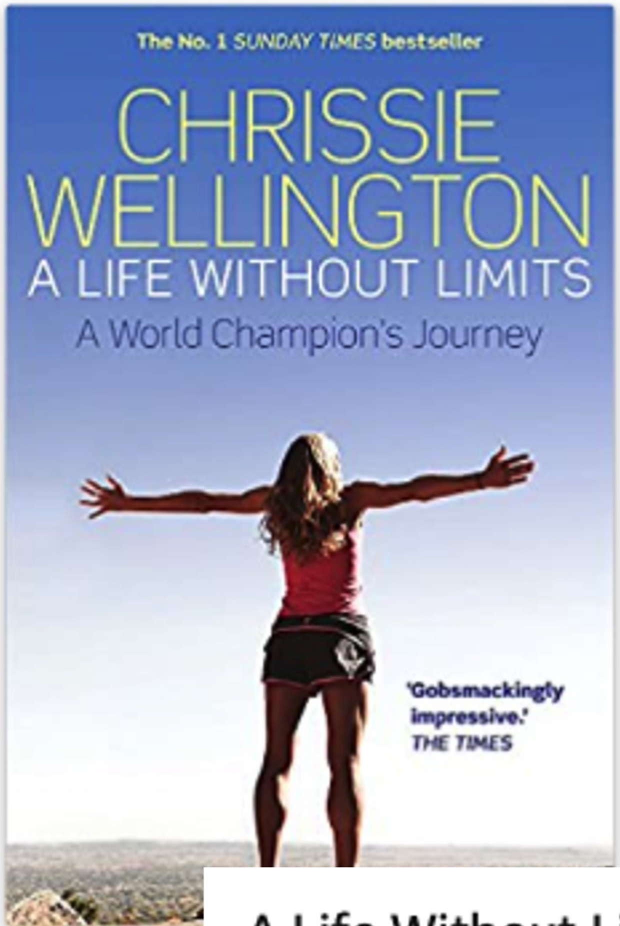
Web Content Mining:
Detecting sentiment, sarcasm, hate

Simone Paolo Ponzetto

FSS 2024

Detecting orientation on Web data

Blick ins Buch

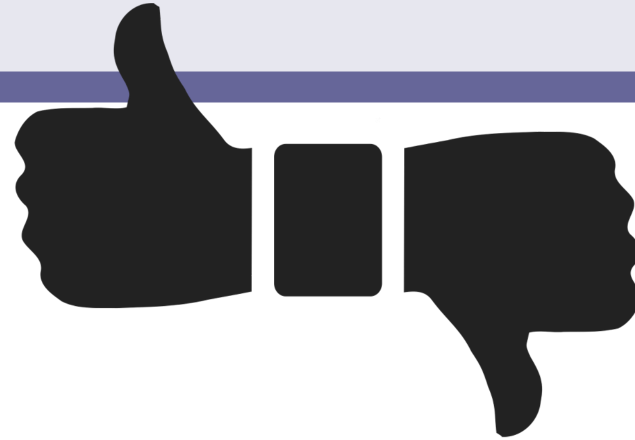


A Life Without Limits: A World Champion's Journey Paperback

English edition | by Chrissie Wellington (Autor)

★★★★☆ 628 ratings

Amazon reviews



P. Schmitt

Go Girl!

ermany on 7 September 2012

Verified Purchase

Das Buch gibt einen schönen Einblick in das Leben von Chrissie Wellington. Mir gefällt das Buch, es ist offenherzig, teilweise selbstkritisch und - das finde ich besonders gut - ohne irgendwelchen "Dann habe ich mich an XY erinnert, habe mich zusammen gerissen und bin einfach weiter gelaufen/gefahren/geflogen", wie es viele Motivationsbücher beinhalten.
Es ist eine Biografie, kein Trainingsbuch und kein ausgewiesenes Motivationsbuch. Doch gerade das macht es für mich zu einem solchen....

Micaela

Inspiring!

he United Kingdom on 29 April 2019

Verified Purchase

I've been a sporty person all my life and I have a competitive personality. I read this book and it inspired me to train hard despite my age. I only started training for marathons and triathlons after 30. This book is great for it covers a great life story, but it is also really interesting for those of us who live for sports.

Discussions on social media (Twitter)

Outline

1. Introduction to Sentiment Analysis / Opinion Mining
2. Constructing Sentiment Lexicons
3. Sentiment Classification
4. Sarcasm Detection
5. Hate Speech Detection

Sentiment Analysis and Opinion Mining

- **Opinionated text is unavoidable on the web:**
  - **Social media posts, product/service reviews**

I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. However, my mother was mad with me as I did not tell her before I bought it. She also thought the phone was too expensive, and wanted me to return it to the shop.

- **Detection of stances and opinions towards people, companies, and products/services has a tremendous business value**
  - **Improving products and services, targeted advertising, revealing trends in election campaigns, ...**

Sentiment Analysis and Opinion Mining

- Sentiment analysis or opinion mining is the computational study of people's opinions, appraisals, attitudes, and emotions towards
  - Entities, individuals, issues, events, topics, and their attributes (aspects)

- Technically, it is very challenging, but practically very useful

- A general sentiment analysis framework aims to answer
  1. Who is the opinion holder?
  2. Towards whom or what is opinion/sentiment expressed?
  3. What is the polarity and intensity of the opinion?
  4. Is an opinion associated with a time-span?

Sentiment Analysis and Opinion Mining

I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. However, my mother was mad with me as I did not tell her before I bought it. She also thought the phone was too expensive, and wanted me to return it to the shop.

Opinion holder	Opinion clue	Target
I	nice	phone
(I)	really cool	touch screen
(I)	clear	voice quality
mother	mad	me
She	too expensive	phone

Sentiment Analysis and Opinion Mining

Formally, an opinion is a quintuple

$$(e_i, a_{ij}, oo_{ijkl}, h_k, t_l)$$

- e_i – the name of the entity which is the target of the expressed sentiment (e.g., iPhone)
- a_{ij} – is the aspect of the entity e_i towards which an opinion is directed (e.g., screen)
- h_k – is the person expressing the opinion (i.e., the person expressing the opinion, for instance I or my girlfried)
- t_l – is the is the time when the opinion towards a_{ij} is expressed by h_k (or the time period during this opinion holds)
- oo_{ijkl} – is the orientation (possibly with intensity) of the opinion (e.g., negative)

- Most opinion mining studies opinions from a large number of opinion holders (⇒ need for opinion summarization)

Outline

1. Introduction to Sentiment Analysis / Opinion Mining
2. Constructing Sentiment Lexicons
3. Sentiment Classification
4. Sarcasm Detection
5. Hate Speech Detection

Sentiment Lexicons

- Sentiment clues (opinion words, sentiment-bearing words) – words and phrases used to express some desired or undesired state
  - Positive clues: *good*, *amazing*, *beautiful*
  - Negative clues: *bad*, *awful*, *terrible*, *poor*

- Sentiment clues are often domain-dependent
  - Quiet speaker phone vs. quiet car engine
  - Separate sentiment lexicons need to be constructed for different domains
    - General lexicons contain words for which the sentiment does not vary across domains

- Q: How would you automatically construct a sentiment lexicon?

Automated acquisition of sentiment lexicons

- Automated acquisition of sentiment lexicon is most often semi-supervised (or weakly supervised)
  1. Start from a small seed lexicon of sentiment words
  2. Iteratively augment the lexicon based on links between words already in the lexicon and words in the large general lexicon or large corpus
  3. Stop when there are no more reliable candidate words to be added to the lexicon
- Approaches for constructing sentiment lexicons are either
  1. Dictionary-based or
  2. Corpus-based
- Often there is a final step of manual cleansing of automatically derived sentiment lexicons

Dictionary-Based Sentiment Lexicon Acquisition

- Bootstrapping using a small seed sentiment lexicon
  - E.g., 10 positive and 10 negative sentiment words
- Idea: exploit semantic links between words in the general lexicon
  - E.g., synonymy and antonymy links in WordNet
  - The procedure is typically iterative
- Additional information can be used to make better lists
  - WordNet glosses
  - Machine learning (classification based on concept definitions)

- Q: What is the shortcoming of dictionary-based approaches?

WordNet

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: estimable   Search WordNet

Display Options: (Select option to change)   Change

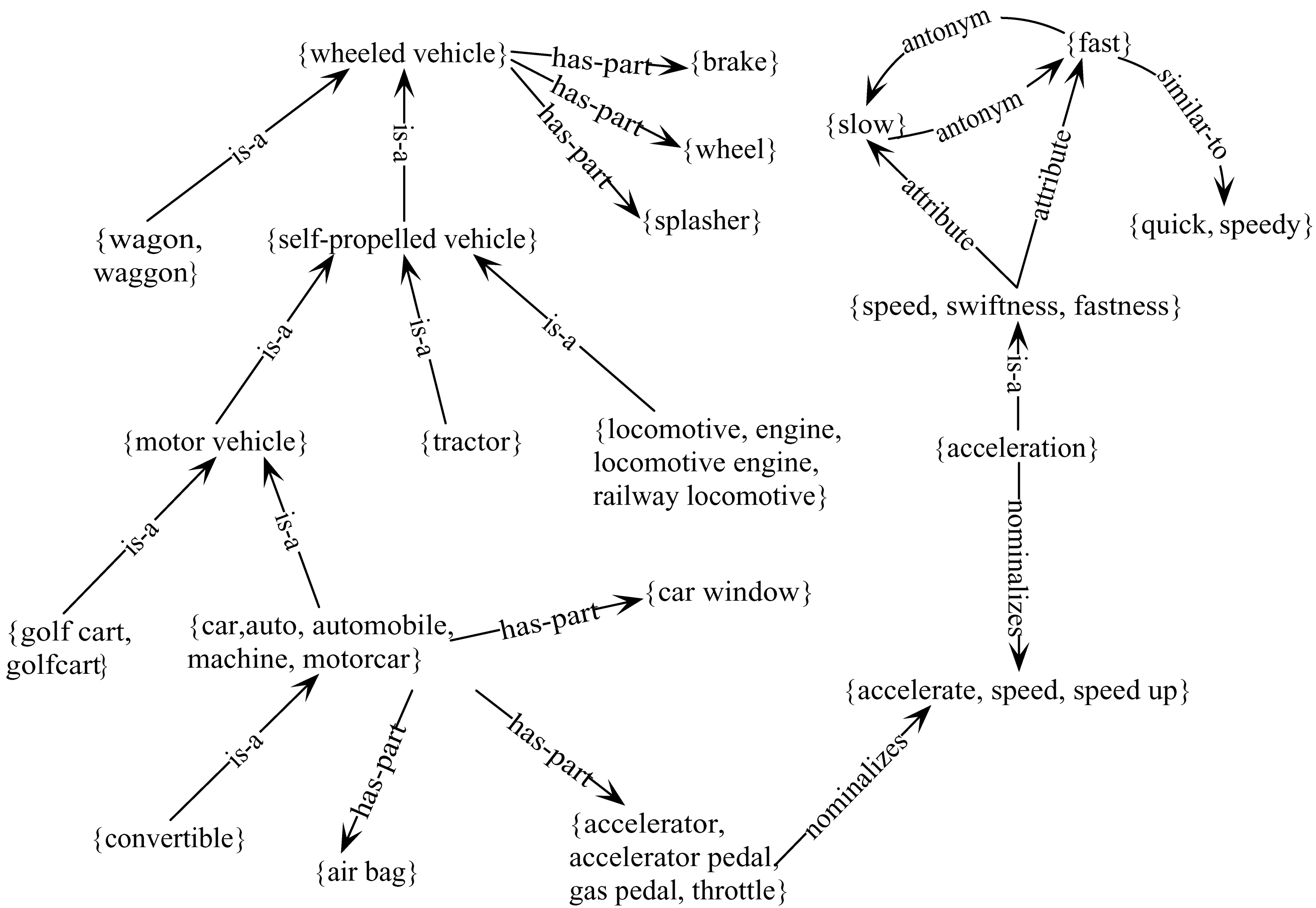
Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations

Display options for sense: (gloss) "an example sentence"

Adjective

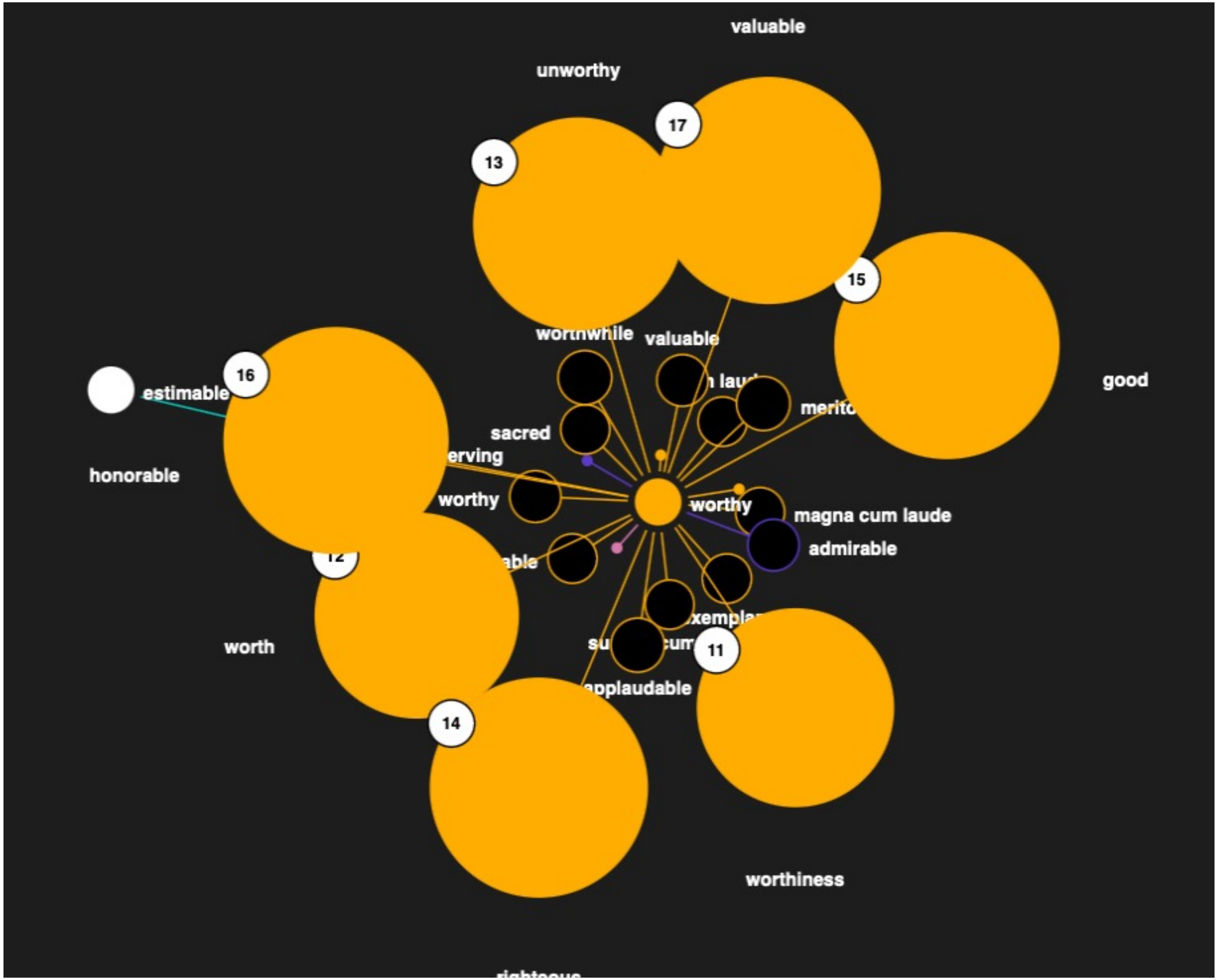
- S: (adj) **estimable** (deserving of respect or high regard)
- S: (adj) **estimable**, good, honorable, respectable (deserving of esteem and respect) *"all respectable companies give guarantees"; "ruined the family's good name"*
- S: (adj) computable, **estimable** (may be computed or estimated) *"a calculable risk"; "computable odds"; "estimable assets"*

WordNet



Source: Navigli (2009)

WordNet

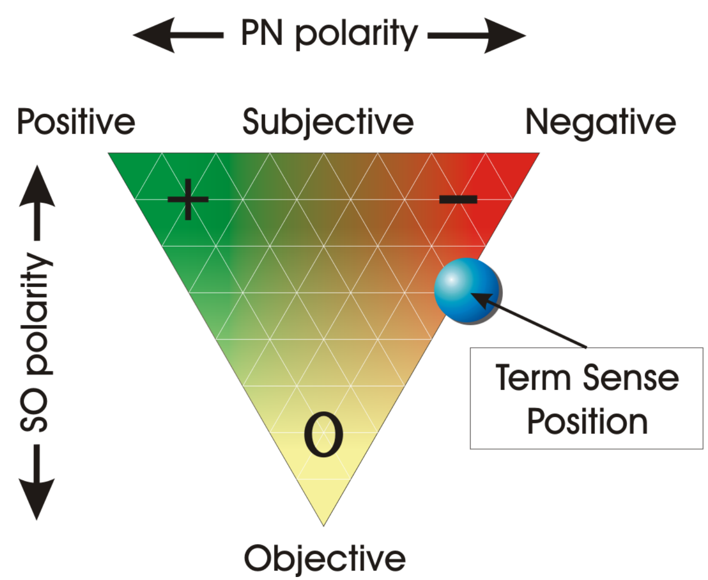


valuable
unworthy
17
13
15
good
16
estimable
honorable
sacred
worthy
worthy
magna cum laude
admirable
valuable
applaudable
11
14
worth
worthiness

SentiWordNet

- **SentiWordNet is a general sentiment lexicon derived from WordNet**
  - **Esuli and Sebastiani (2006); Bacianella et al., (2010)**
- **It contains automated annotations of all WordNet synsets with sentiment scores:**
  - **Positivity score: *Pos(s)***
  - **Negativity score*: Neg(s)***
  - **Objectivity score*: Obj(s)***
  - **For each synset *s:***

***Pos(s) + Neg(s) + Obj(s)* = 1**



- **[estimable(J,3)] "may be computed or estimated"**

```
Pos  0    Neg 0   Obj 1
```

- **[estimable(J,1)] "deserving of respect or high regard"**

```
Pos .75   Neg 0   Obj .25
```

SentiWordNet

First step: Semi-supervised learning

1. **Small positive and negative seed sets (7 synsets each)**
2. **Seed set expansion via WordNet relations: *also-see*, *direct antonymy***
3. **Expanded seed sets used as training data for a ternary classifier *(Pos, Neg, Obj)***
   - **Synset glosses used as bag-of-words features for a classifier**
   - **Classification performed for all WordNet synsets**

Second step: The random walk

1. **Construct a WordNet graph based on definiens-definiendum relation**
2. **Run a label propagation algorithm on the induced WordNet graph**
   - **Two runs: one for positive *Pos(s)* and another for negative *Neg(s)* labels**
3. **Normalize *Pos(s)* and *Neg(s)* over all synsets**
4. **Compute the objective scores, *Obj(s)* = 1 – *Pos(s)* – *Neg(s)***

Corpus-Based Sentiment Lexicon Acquisition

- **Methodologically, corpus-based induction of sentiment lexicons resembles to the dictionary-based:**
  1. **Semi-supervised learning from small initial seed sets**
  2. **Graph-based propagation of positive and negative sentiment**

- **Difference:**
  - **Graph for label propagation is computed from word co-occurrences in a large corpus**
  - **The resulting lexicon specific to the domain of the corpus**

- **Some (simple) approaches:**
  - **Sentiment consistency, conjunction of adjectives (Hatzivassiloglou & McKeown, 1997)**
  - **Pointwise mutual information (PMI) of candidate words with seed set words (Turney & Littman, 2002)**
  - **PMI-induced graph with PageRank label propagation and supervised learning (Glavaš and Šnajder, 2012)**

Hatzivassiloglou & McKeown (1997)

- Adjectives conjoined by “*and*” have same polarity
  - Fair and legitimate, corrupt and brutal
  - *fair and brutal, *corrupt and legitimate
- Adjectives conjoined by “*but*” do not
  - fair but brutal
- Step 1: Label seed set of 1336 adjectives (all >20 in 21-million-word WSJ corpus)
  - 657 positive: adequate central clever famous intelligent remarkable reputed sensitive slender thriving…
  - 679 negative: contagious drunken ignorant lanky listless primitive strident troublesome unresolved unsuspecting…

Hatzivassiloglou & McKeown (1997)

- **Step 2: Expand seed set to conjoined adjectives**
  - **Look in the corpus (or now, on the Web) for conjunctions of adjectives**

Google "was nice and"

Nice location in Porto and the front desk staff **was nice and** helpful ...
www.tripadvisor.com/ShowUserReviews-g189180-d206904-r12068... +1
Mercure Porto Centro: Nice location in Porto and the front desk staff **was nice and** helpful - See traveler reviews, 77 candid photos, and great deals for Porto, ...

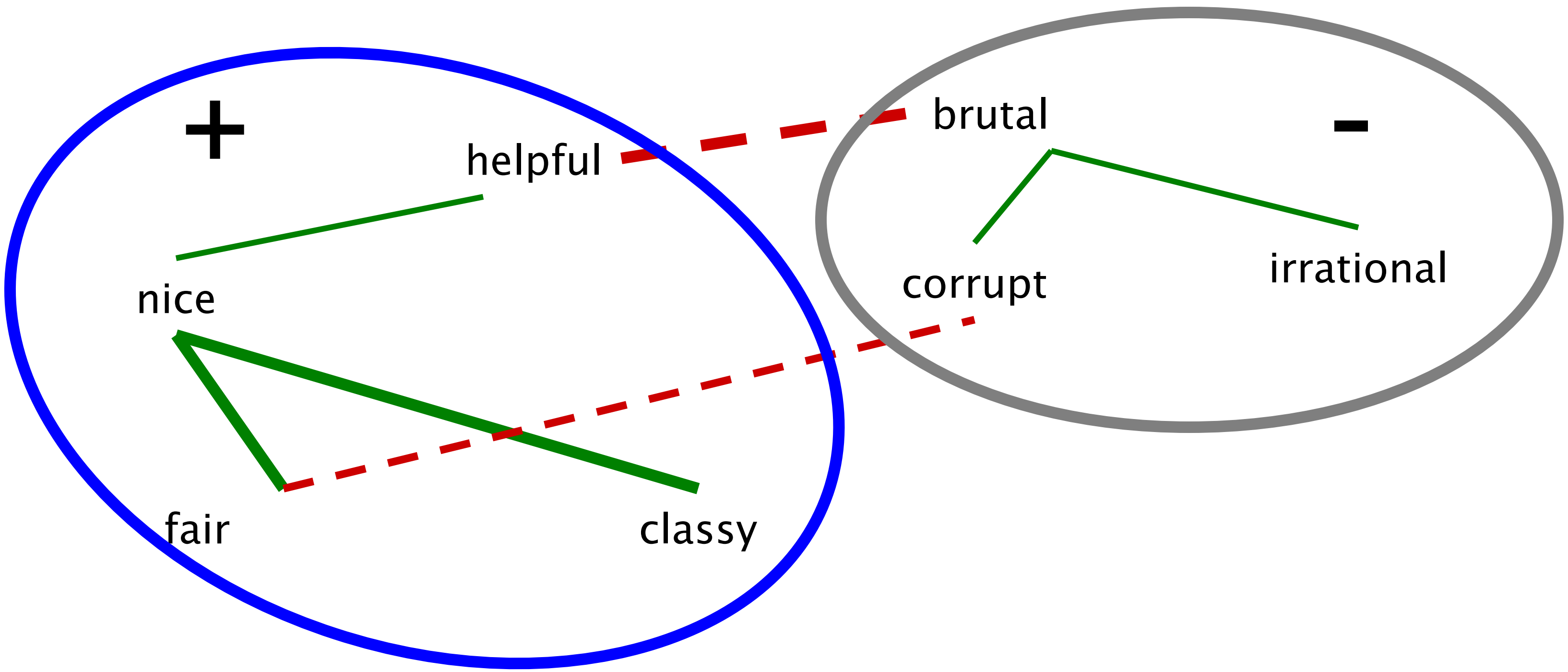
nice, helpful

If a girl **was nice and** classy, but had some vibrant purple dye in ...
answers.yahoo.com › Home › All Categories › Beauty & Style › Hair +1
4 answers - Sep 21
Question: Your personal opinion or what you think other people's opinions might ...
Top answer: I think she would be cool and confident like katy perry :)

nice, classy

Hatzivassiloglou & McKeown (1997)

- **Step 3: Supervised classifier assigns “polarity similarity” to word pair**
- **Step 4: Clustering for partitioning the graph into two**



Turney (2002)

1. Extract a *phrasal lexicon* from reviews
2. Learn polarity of each phrase
3. Rate a review by the average polarity of its phrases

Turney (2002)

- **Extract two-word phrases with adjectives**

First Word	Second Word
JJ	NN or NNS
RB, RBR, RBS	JJ
JJ	JJ
NN or NNS	JJ
RB, RBR, or RBS	VB, VBD, VBN, VBG

- **Positive phrases co-occur more with *"excellent"***
- **Negative phrases co-occur more with *"poor"***
- **But how to measure co-occurrence?**

Turney (2002)

- **PMI between two words:**
  - **How much more do two words co-occur than if they were independent?**

$$\mathrm{PMI}(word_1, word_2) = \log_2 \frac{P(word_1, word_2)}{P(word_1)P(word_2)}$$

- **Counts collected using a search engine:**
  - **$P(word_1, word_2)$ estimated by `hits(word1 NEAR word2)/N`**
  - **`P(word)` estimated by `hits(word)/N`**

$$\mathrm{PMI}(word_1, word_2) = \log_2 \frac{\frac{1}{N} hits(word_1 \text{ NEAR } word_2)}{\frac{1}{N} hits(word_1) \frac{1}{N} hits(word_2)}$$

Does phrase appear more with “poor” or “excellent”?

$$\text{Polarity}(\textit{phrase}) = \text{PMI}(\textit{phrase}, \text{"excellent"}) - \text{PMI}(\textit{phrase}, \text{"poor"})$$

$$= \log_2 \frac{\frac{1}{N}\textit{hits}(\textit{phrase} \text{ NEAR "excellent"})}{\frac{1}{N}\text{hits}(\textit{phrase})\frac{1}{N}\text{hits}(\text{"excellent"})} - \log_2 \frac{\frac{1}{N}\text{hits}(\textit{phrase} \text{ NEAR "poor"})}{\frac{1}{N}\text{hits}(\textit{phrase})\frac{1}{N}\text{hits}(\text{"poor"})}$$

$$= \log_2 \frac{\text{hits}(\textit{phrase} \text{ NEAR "excellent"})}{\text{hits}(\textit{phrase})\text{hits}(\text{"excellent"})} \frac{\text{hits}(\textit{phrase})\text{hits}(\text{"poor"})}{\text{hits}(\textit{phrase} \text{ NEAR "poor"})}$$

$$= \log_2 \left(\frac{\text{hits}(\textit{phrase} \text{ NEAR "excellent"})\text{hits}(\text{"poor"})}{\text{hits}(\textit{phrase} \text{ NEAR "poor"})\text{hits}(\text{"excellent"})} \right)$$

Phrases from a thumbs-up review

Phrase	POS tags	Polarity
online service	JJ NN	2.8
online experience	JJ NN	2.3
direct deposit	JJ NN	1.3
local branch	JJ NN	0.42
…		
low fees	JJ NNS	0.33
true service	JJ NN	-0.73
other bank	JJ NN	-0.85
inconveniently located	JJ NN	-1.5
Average		0.32

Phrases from a thumbs-down review

Phrase	POS tags	Polarity
direct deposits	JJ NNS	5.8
online web	JJ NN	1.9
very handy	RB JJ	1.4
…		
virtual monopoly	JJ NN	-2.0
lesser evil	RBR JJ	-2.3
other problems	JJ NNS	-2.8
low funds	JJ NNS	-6.8
unethical practices	JJ NNS	-8.5
Average		-1.2

Outline

1. Introduction to Sentiment Analysis / Opinion Mining
2. Constructing Sentiment Lexicons
3. Sentiment Classification
4. Sarcasm Detection
5. Hate Speech Detection

Sentiment classification

- The goal is to classify an opinionated portion of text (e.g., product review) as expressing (dominantly) positive or negative sentiment
  - Typically, we classify a document, but paragraphs and sentences have been addressed as well
- Assumption: entire text portion addresses a single entity
  - Holds for product reviews but not for social media posts
- Capturing the overall sentiment expressed toward the entity
  - Sentiment toward specific aspects of the entity ignored

- Methodological approaches:
  1. Supervised learning (i.e., supervised text classification; dominantly)
  2. Unsupervised learning

Supervised sentiment classification

- Typically formulated as a ternary (Positive, Negative, Neutral) text classification task
- Training and testing data – typically product reviews
  - Labels often readily available via user ratings (e.g., 1 to 5 stars)
- Classification:
  - Feature-design algorithms

    The usual suspects: logistic regression, SVM, …

    Features

    - Bag of words, POS tags, opinion clues and phrases (from dictionary)
    - Negations (change opinion orientation) and syntactic dependencies
  - Semantic representation-based algorithms
    - CNNs, RNNs, Autoencoders, Recursive NN (for sentiment classification)
    - Raw text input (word or character embeddings), no need for manually designed features

Intro to logistic regression

- Let us focus on the binary case (positive vs. negative)
- Goal: we would like to build a model that computes the probability of an input to belong a certain (here, binary {0,1}) class as a linear combination of the input features and their weights
- For each feature x_i, weight w_i tells us the importance of x_i
- Note: there is also a term w_0 (also called the bias b).
- Just like we do for linear regression, we sum up all the weighted features and the bias

$$z = \left(\sum_{i=1}^{n} w_i x_i\right) + b$$

$$z = w \cdot x + b$$

- If this sum is high, we say $y = 1$, if low, then $y = 0$

Logistic regression as a probabilistic classifier

- What we are after is a classifier that gives us the *probability* of the positive and negative classes given the observed instance, i.e., $P(y=1|x,\boldsymbol{w})$ and $P(y=0|x,\boldsymbol{w})$

- But the linear combination of features and coefficients isn't a probability, it's just a number!
- Since weights are real-valued, the output might even be negative; z ranges from $-\infty$ to ∞.

- Solution: use a function of *z* that goes from 0 to 1

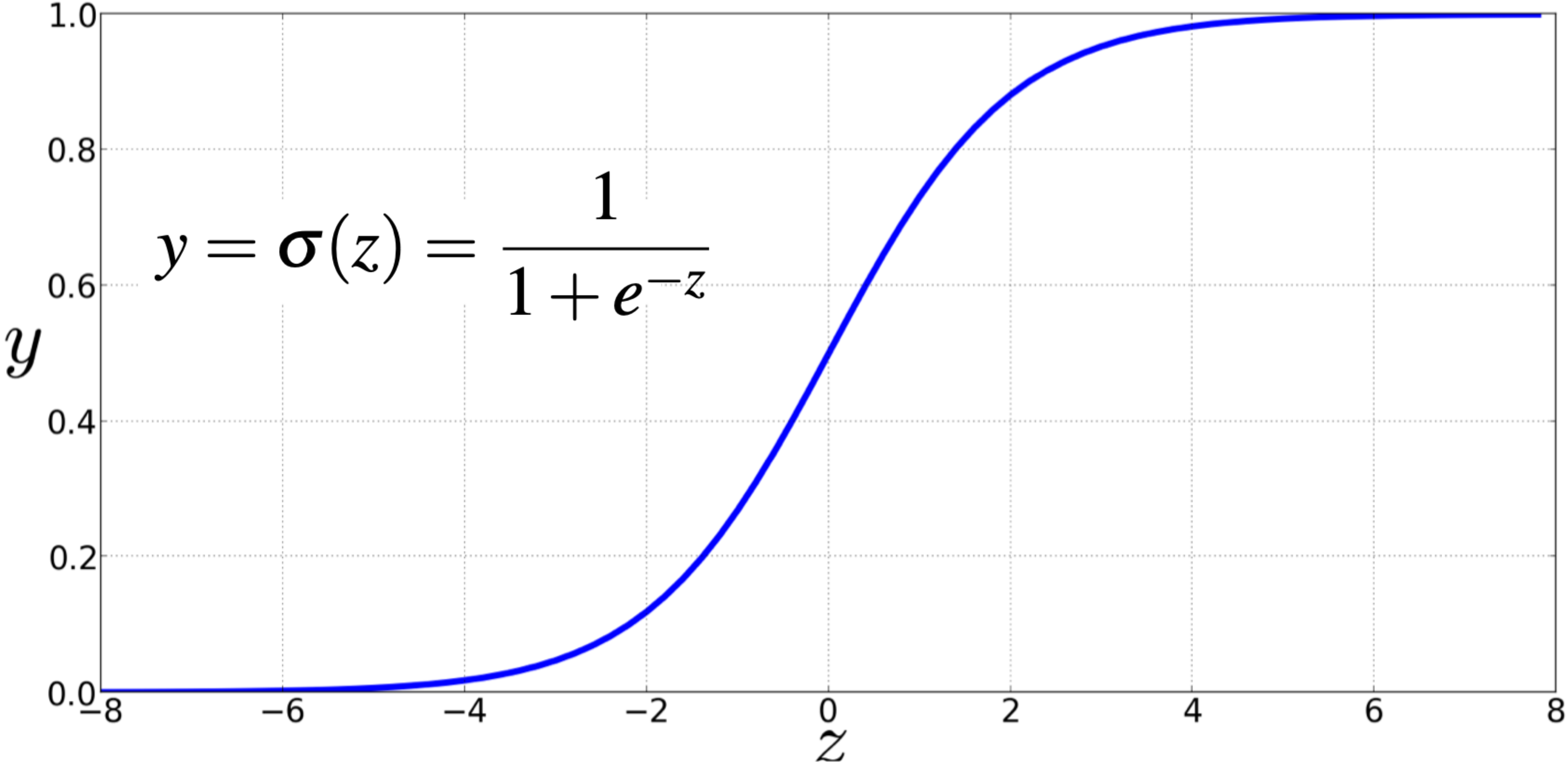
The standard logistic function (a.k.a. sigmoid)

- **The logistic regression model uses a function, called the *logistic* function, to model $P(y = 1)$**

$$y = \sigma(z) = \frac{1}{1+e^{-z}} = \frac{1}{1+\exp(-z)}$$

$$z = w \cdot x + b$$

The standard logistic function (a.k.a. sigmoid)



The two phases of logistic regression

- **<u>Training</u>: we learn weights w using stochastic gradient descent and cross-entropy loss.**

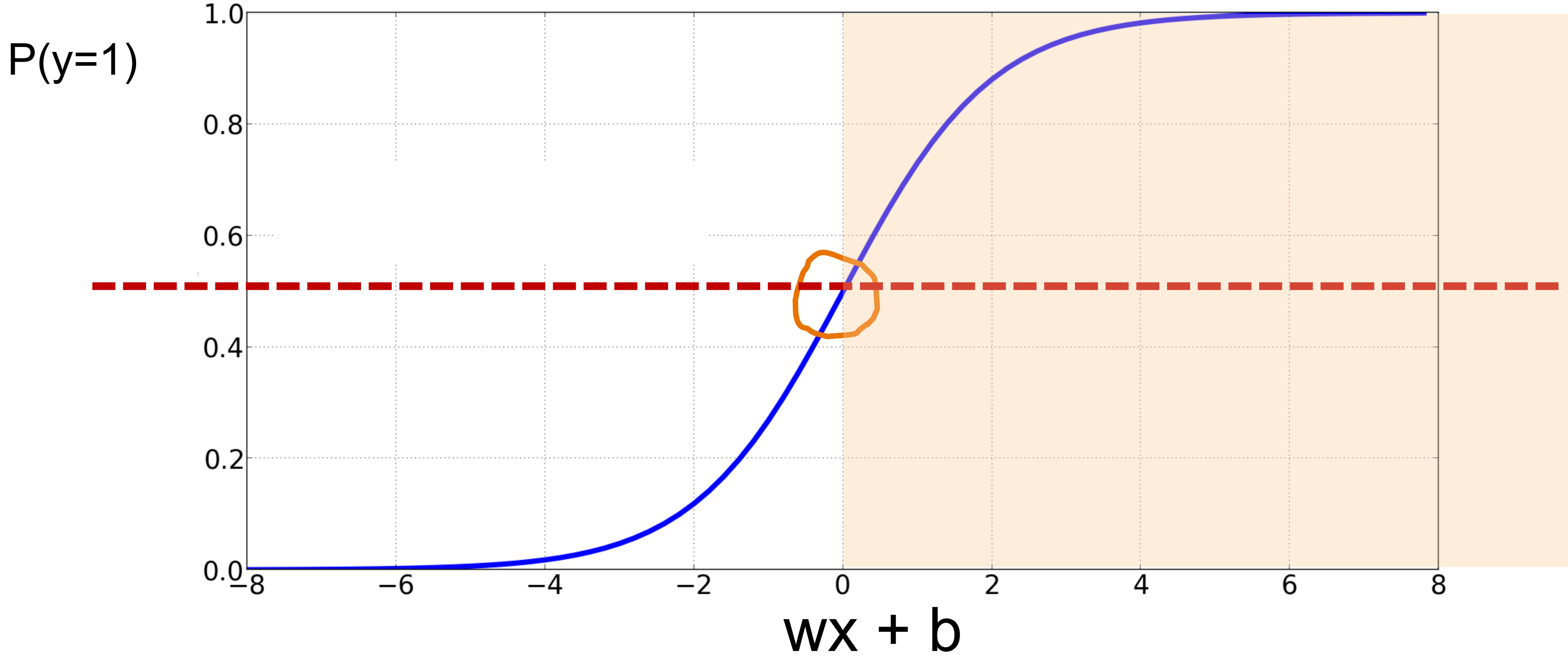
- **<u>Test</u>: Given a test example x we compute $p(y|x)$ using learned weights w, and return whichever label (*y* = 1 or *y* = 0) has higher probability.**

Computing probabilities / doing classification

$$
\begin{aligned}
P(y=1) &= \sigma(w \cdot x + b) \\
&= \frac{1}{1+\exp\left(-(w \cdot x + b)\right)}
\end{aligned}
$$

$$
\begin{aligned}
P(y=0) &= 1-\sigma(w \cdot x + b) \\
&= 1-\frac{1}{1+\exp\left(-(w \cdot x + b)\right)} \\
&= \frac{\exp\left(-(w \cdot x + b)\right)}{1+\exp\left(-(w \cdot x + b)\right)}
\end{aligned}
$$

Using the output of the sigmoid as a classifier



$$\hat{y} = \begin{cases} 1 & \text{if } P(y=1|x) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

if $w \cdot x + b > 0$

if $w \cdot x + b \leq 0$

Feature design

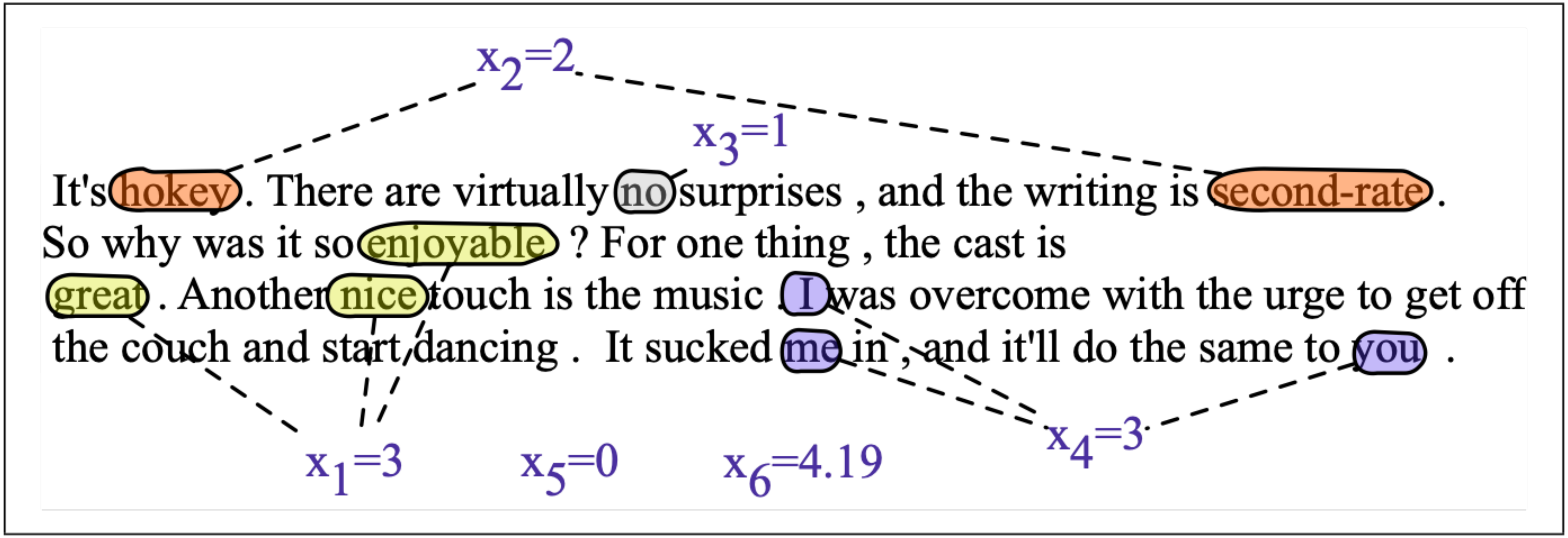
– **The key question is how to come up with good (useful) features**
– **Two approaches:**
  - **Use your intuition (insight, linguistic/domain expertise), and design a small set of good features that you think should work**
  - **Throw in everything you can (the "kitchen sink" approach), and them maybe prune later**
– **You will often want to see which features work and which don't:**
  - **Ablation study – turn off some features, retrain the model and see how the performance changes**
  - **Feature selection – use a method to select the best features. This can also improve the performance (especially in a "kitchen sink" approach)**
– **One of the great advantages of deep learning for NLP is the absence of feature engineering**

Example: sentiment classification with logistic regression

- **Suppose we are doing binary sentiment classification on movie review text, and we would like to know whether to assign the sentiment class 1=positive or 0=negative to the following review:**

It's hokey. There are virtually no surprises, and the writing is second-rate. So why was it so enjoyable? For one thing, the cast is great. Another nice touch is the music. I was overcome with the urge to get off the couch and start dancing. It sucked me in, and it'll do the same to you.

Example: sentiment classification with logistic regression



Var	Definition	Value
x_1	count(positive lexicon) $\in$ doc)	3
x_2	count(negative lexicon) $\in$ doc)	2
x_3	$\begin{cases} 1 & \text{if “no”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	1
x_4	count(1st and 2nd pronouns $\in$ doc)	3
x_5	$\begin{cases} 1 & \text{if “!”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	0
x_6	log(word count of doc)	$\ln(66) = 4.19$

Classifying sentiment for our review as input

Var	Definition	Value
x_1	$\text{count(positive lexicon)} \in \text{doc)}$	3
x_2	$\text{count(negative lexicon)} \in \text{doc)}$	2
x_3	$\begin{cases} 1 & \text{if ``no''} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	1
x_4	$\text{count}(1\text{st and 2nd pronouns} \in \text{doc})$	3
x_5	$\begin{cases} 1 & \text{if ``!''} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	0
x_6	$\log(\text{word count of doc})$	$\ln(66) = 4.19$

Suppose $w = [2.5, -5.0, -1.2, 0.5, 2.0, 0.7]$

$b = 0.1$

Classifying sentiment for our review as input

$$\begin{aligned} p(+|x) = P(Y=1|x) &= \sigma(w \cdot x + b) \\ &= \sigma([2.5, -5.0, -1.2, 0.5, 2.0, 0.7] \cdot [3, 2, 1, 3, 0, 4.19] + 0.1) \\ &= \sigma(.833) \\ &= 0.70 \end{aligned}$$

$$\begin{aligned} p(-|x) = P(Y=0|x) &= 1 - \sigma(w \cdot x + b) \\ &= 0.30 \end{aligned}$$

- **We classify the review as *positive***

Text Classification in logistic regression: summary

- Given:
  - a set of classes: (+ sentiment,- sentiment)
  - a vector x of features [x1, x2, …, xn]. Examples:
    - x1= count("awesome")
    - x2 = log(number of words in review)
  - A vector w of weights [w1, w2, …, wn]
    - w_i for each feature f_i
- Compute the probability of the positive class as:

$$
\begin{aligned}
P(y=1) &= \sigma(w \cdot x + b) \\
&= \frac{1}{1+e^{-(w \cdot x + b)}}
\end{aligned}
$$

Multinomial Logistic Regression

- Often, we have more than two classes (e.g., positive, negative and neutral)
- That is, we need to generalize our binary model to predict more than 2 classes: we call this multinomial logistic regression
- Idea: compute the probability distribution over k classes from the linear combination of (class-specific) weights and input features
- For this, we need first to define a generalization of the sigmoid for multiple classes, where the output (i.e., the total probability mass) over all classes must sum up to 1: i.e., $\sum_i p(y_i) = 1$

The softmax function

- Input: A vector $z = [z_1, z_2, ..., z_k]$ of k arbitrary values
- Output: a probability distribution
  - each value in the range [0,1]
  - all the values summing to 1

$$\text{softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{k} \exp(z_j)} \quad 1 \leq i \leq k$$

$$\text{softmax}(z) = \left[\frac{\exp(z_1)}{\sum_{i=1}^{k} \exp(z_i)}, \frac{\exp(z_2)}{\sum_{i=1}^{k} \exp(z_i)}, \ldots, \frac{\exp(z_k)}{\sum_{i=1}^{k} \exp(z_i)} \right]$$

Softmax in multinomial logistic regression

- **We compute the probability of a class *c* given observation *x* as:**

$$p(y=c|x) = \frac{\exp\left(w_c \cdot x + b_c\right)}{\sum_{j=1}^{k} \exp\left(w_j \cdot x + b_j\right)}$$

- **Input is still the <u>dot product between weight vector *w* and input vector *x*</u> (and a bias term)**
- **But now we have separate weight vectors w_c and bias terms b_c for each of the *k* classes**
- **(For learning weights *w* we can still use stochastic gradient descent and cross-entropy loss)**

Features in binary versus multinomial logistic regression

- **Binary: positive weight → y=1 neg weight → y=0**

$$x_5 = \begin{cases} 1 & \text{if “!”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases} \qquad \mathrm{w}_5 = 3.0$$

- **Multinominal: separate weights for each class:**

Feature	Definition	$w_{5,+}$	$w_{5,-}$	$w_{5,0}$
$f_5(x)$	$\begin{cases} 1 & \text{if “!”} \in \text{doc} \\ 0 & \text{otherwise} \end{cases}$	3.5	3.1	−5.3

Unsupervised Sentiment Classification

- If user ratings are not available, we need manual labelling for supervised machine learning methods
  - Tedious, expensive, time-consuming
- A typical unsupervised approach to sentiment classification:
  1. Extract candidate phrases (e.g., matching predefined POS patterns)
  2. For reach word/phrase, compute some association score (e.g., pointwise mutual information) with sentiment lexicon entries, on a large corpus
     - Association scores (e.g., PMI) with positive seed words
     - Association scores (e.g., PMI) with negative seed words
  3. The sentiment orientation of each phrase is computed as:

$$SO(phr) = \frac{1}{|pos|} \cdot \sum_{p \in pos} PMI(phr, p) - \frac{1}{|neg|} \cdot \sum_{n \in neg} PMI(phr, n)$$

  4. The sentiment of the document is determined by summing or averaging the sentiment orientations of phrases it contains

Unsupervised Sentiment Classification

3. The sentiment orientation of each phrase is computed as:

$$SO(phr) = \frac{1}{|pos|} \cdot \sum_{p \in pos} PMI(phr, p) - \frac{1}{|neg|} \cdot \sum_{n \in neg} PMI(phr, n)$$

4. The sentiment of the document is determined by summing or averaging the sentiment orientations of phrases it contains

- Example:
- pos = { *good*, *beautiful* } and neg = { *bad*, *ugly* }
- PMI scores:

	good	*beautiful*	*bad*	*ugly*
new	0.4	0.7	-0.1	0.2
sneakers	0.2	0.2	0.4	0.3

- SO of ``new sneakers''?

Unsupervised Sentiment Classification

- **Example:**
- **pos = { *good*, *beautiful* } and neg = { *bad*, *ugly* }**
- **PMI scores:**
- **SO of ``new sneakers''?**

	good	*beautiful*	*bad*	*ugly*
new	0.4	0.7	-0.1	0.2
sneakers	0.2	0.2	0.4	0.3

$$SO(\text{new}) = \frac{0.4+0.7}{2} - \frac{-0.1+0.2}{2} = 0.55 - 0.05 = 0.5$$
$$SO(\text{sneakers}) = \frac{0.2+0.2}{2} - \frac{0.4+0.3}{2} = 0.2 - 0.35 = -0.15$$
$$SO(\text{new sneakers}) = 0.5 - 0.15 = 0.35$$

ALTERNATIVE (avg instead of sum): $SO(\text{new sneakers}) = \frac{0.5-0.15}{2} = 0.175$

Outline

1. Introduction to Sentiment Analysis / Opinion Mining
2. Constructing Sentiment Lexicons
3. Sentiment Classification
4. Sarcasm Detection
5. Hate Speech Detection

Sarcasm detection

- **Non-transparent expressions of sentiment cause most errors in sentiment analysis and opinion mining**
  - **Irony and sarcasm being most salient**
- **Sarcasm is a sharp, bitter, or cutting expression or remark; a bitter gibe or taunt**
- **Sarcasm is notoriously difficult to detect in text, even for humans!**

Sarcasm detection

- The variation by which sarcasm is expressed is basically unlimited
- Computational approaches focus merely on specific types of sarcasm
  - Sarcasm as contrast of negative situations and positive sentiment (Riloff et al., 2013)

- Sarcasm as contrast – examples
  - Oh how I love being ignored.
  - Thoroughly enjoyed shoveling the driveway today!
  - Absolutely adore it when my bus is late.
  - I’m so pleased mom woke me up with vacuuming this morning.

Detecting Sarcasm as Contrast

- Detecting sarcasm in tweets as contrast between negative situation and positive sentiment
- Boostrapping rule-based algorithm that automatically learns positive sentiment phrases and negative situation phrases:
  1. Start with (1) single positive sentiment word (*love*) and (2) a set of tweets with hashtag #sarcasm or #sarcastic
  2. Negative situation candidates – n-grams (1-3) that directly follow positive sentiment phrases and fulfill pre-defined POS patterns
  3. Positive sentiment candidates – n-grams (1-3) near the negative situation phrases that satisfy POS patterns
  4. Candidates are scored based on ratio of frequencies in sarcastic (with hashtags) vs. non-sarcastic tweets

Detecting Sarcasm as Contrast

- Some extracted positive sentiment phrases:
  - *missed*, *loves*, *enjoy*, *can't wait*, *excited*, *wanted*, *can't wait*, *appreciate*, *loving*, *really like*, *looooove*, *just keeps*, loveee, ...
- Some extracted negative situation phrases:
  - *being ignored*, *being sick*, *waking up early*, *cleaning*, *crying*, *sitting at home*, *being told*, *not sleeping*, *not talking*, *doing homework*, *being ditched*, *falling*, *walking home*, *getting yelled at*, *taking care*, ...
- Detection performance: 51% F1-score
  - On a very constrained sarcasm detection task
  - Just proves the difficulty of sarcasm detection

Outline

1. Introduction to Sentiment Analysis / Opinion Mining
2. Constructing Sentiment Lexicons
3. Sentiment Classification
4. Aspect-Oriented Sentiment Analysis
5. Sarcasm Detection
6. Hate Speech Detection

Hate Speech

- **Hate speech (HS) is commonly defined as any communication that**
  - **disparages a person or a group**
  - **on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other.**
- **Expressions that:**

  (i) **incite discrimination or violence due to racial hatred, xenophobia, sexual orientation and other types of intolerance;**

  (ii) **foster hostility through prejudice and intolerance.**

J. T. Nockleby (2000). **Hate speech**. Encyclopedia of the American Constitution (2nd ed., edited by Leonard W. Levy, Kenneth L. Karst et al., New York: Macmillan), pp. 1277–1279

Hate Speech and social media

Facebook Admits It Was Used to Incite Violence in Myanmar

Rohingya refugees after crossing the Naf River, which separates Myanmar and Bangladesh, in 2017. A report commissioned by Facebook found the company failed to keep its platform from being used to "foment division and incite offline violence" in Myanmar. Adam Dean for The New York Times

https://www.nytimes.com/2018/11/06/technology/myanmar-facebook.html

Hate Speech: definitions

Source	Hate speech is to incite violence or hate	Hate speech is to attack or diminish	Hate speech has specific targets	Humour has a specific status
EU Code of conduct	Yes	No	Yes	No
ILGA	Yes	No	Yes	No
Scientific paper	No	Yes	Yes	No
Facebook	No	Yes	Yes	Yes
YouTube	Yes	No	Yes	No
Twitter	Yes	Yes	Yes	No

P. Fortuna, S. Nunes (2018). **A survey on automatic detection of hate speech in text.** ACM Computing Surveys (CSUR) 51.4

Hate Speech: more definitions!

Table 1 Glossary of terms relevant to the present survey, with their definitions from the literature

Term and definitions	Source
Hate Speech	Warner and Hirschberg (2012)
Any communication that disparages a person or a group on the basis of some characteristic such as race, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic	
Use of a sexist or racial slur, attack a minority, promotes hate speech or violent crime, blatantly misrepresents truth, shows support of problematic hashtags, defends xenophobia or sexism, or contains a screen name that is offensive	Waseem and Hovy (2016)
Act of offending, insulting or threatening a person or a group of similar people on the basis of religion, race, caste, sexual orientation, gender or belongingness to a specific stereotyped community	Schmidt and Wiegand (2017)
Language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group	Davidson et al. (2017)
Any communication that disparages a target group of people based on some characteristic such as race, colour, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic	Nockleby (2000)

F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, V. Patti. **Resources and benchmark corpora for hate speech detection: a systematic review**. Language Resources and Evaluation, 2020

Example tweets

[Example of indirect insult.]
*@USER Everyone saying **fuck** Russ dont know a damn thing about him or watched the interview* 🤷‍♀️🤷‍♀️🤷‍♀️

[Ex. i): offensive tweet & abusive swearing]
*@USER You are an absolute **dick*** 😡

[Ex. ii): offensive tweet & not abusive swearing]
*@USER I was definitely drunk as **shit***

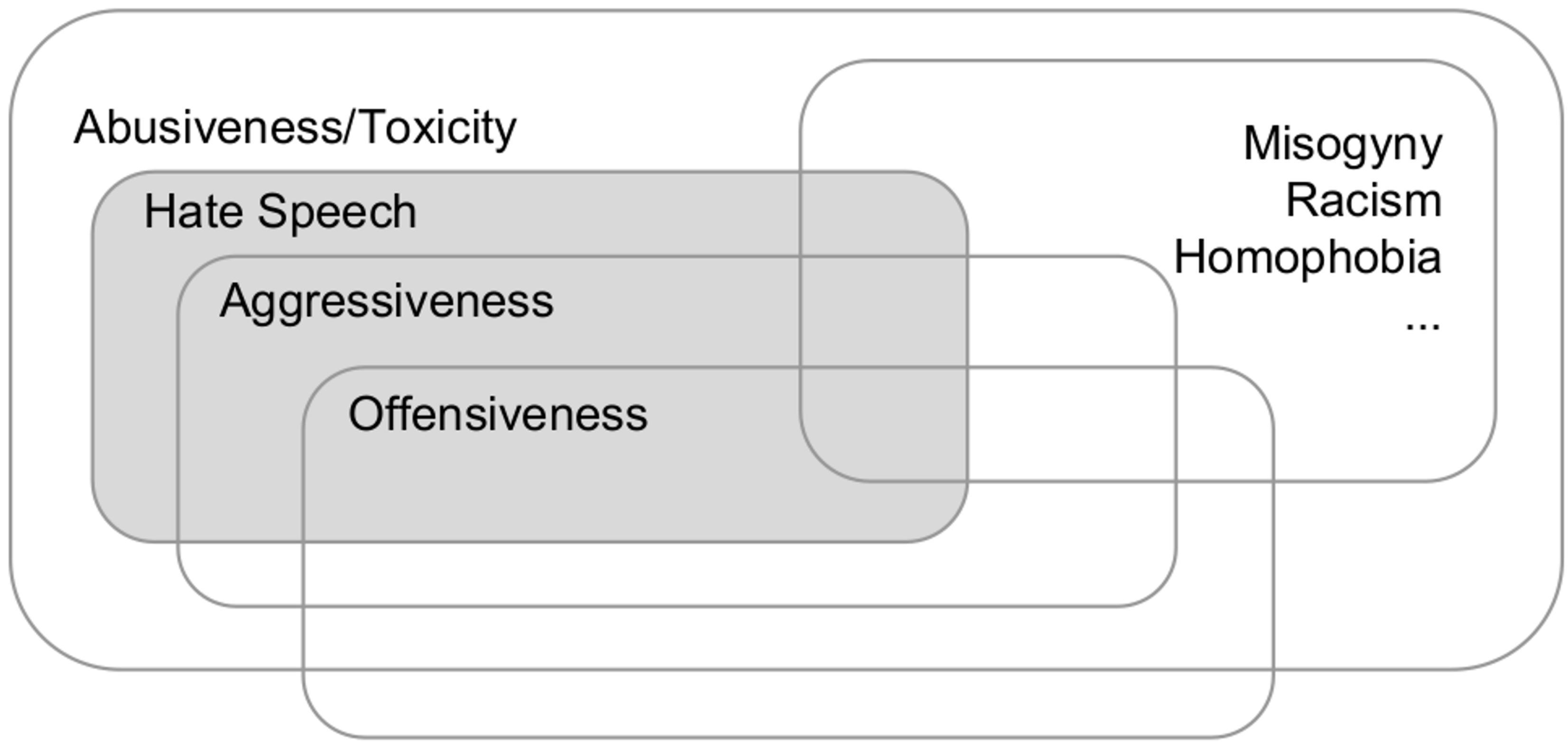
[Ex. iii): not offensive tweet & abusive swearing]
*@USER **bullshit** there's rich liberals too so what are you saying ???*

[Ex. iv): not offensive tweet & not abusive swearing]
*@USER Haley thanx! you know how to brighten up my **shitty** day* 😘

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2020. Do You Really Want to Hurt Me? Predicting Abusive Swearing in Social Media. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6237–6246, Marseille, France. European Language Resources Association.

Hate Speech, offensive language, etc.

- One of the major issues consists in the intrinsic complexity in defining HS and in a widespread vagueness in the use of related terms (such as **abusive, toxic, dangerous, offensive or aggressive language**), that often **overlap** and are prone to strongly **subjective interpretations**



Lexicons for hate speech / offensive language

- Just like there exists sentiment lexicons we have **lexicons for hurtful language**

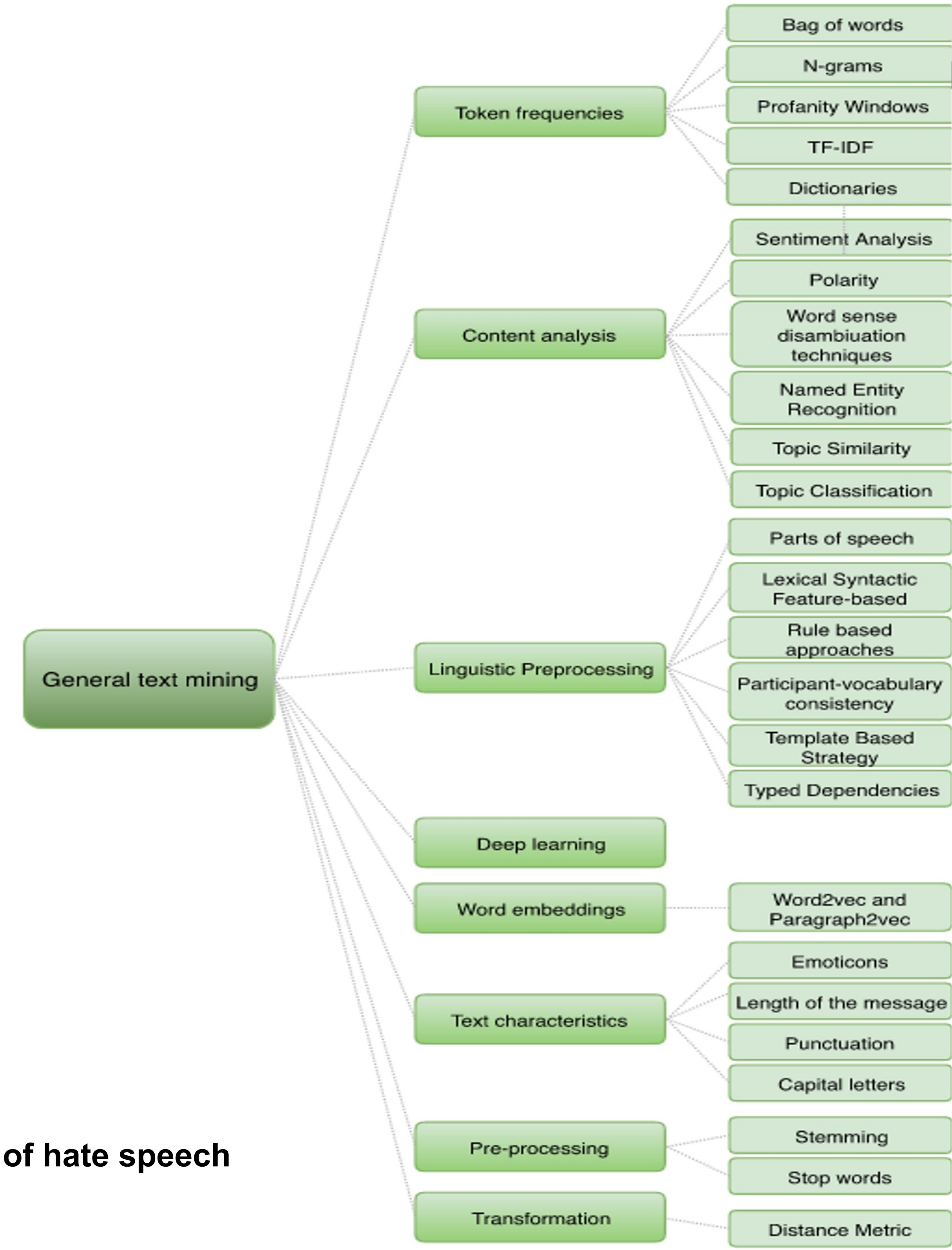
- **HurtLex (Bassignana et al., 2018)**
  - **Multilingual lexicon of "words to hurt"**
  - **53 languages**
  - **17 categories + stereotype**

HurtLex (Bassignana et al., 2018)

Category	# Terms	Examples
negative stereotypes ethnic slurs	371	barbarian, idiotic, dummy, n***oes, infertility
locations and demonyms	24	genoan, savage, barbarian, tike, boor
professions and occupations	192	wooer, politician, peasant, fishwife, academism
physical disabilities and diversity	63	handycapped, midget, worthless, invalidity, impaired
cognitive disabilities and diversity	491	artless, retarded, simple, goof, brute
moral and behavioral defects	715	close-minded, cheater, stinking, forgery, faker
words related to social and economic disadvantage	124	miscreants, miserable, wretch, pitiful, villain
plants	177	finocchio, potato, papaya whip, squash, f**ot
animals	996	b***h, t**t, goose, scoundrel, beastly
male genitalia	426	wanky, c**k, testicles, phallic, prick
female genitalia	144	babe, c**t, t**t, boob, p***y
words related to prostitution	276	s*ut, street walker, crack h*, hooker, w***e
words related to homosexuality	361	drag, crossdressing, shirtlifter, f**, qu**rio
with potential negative connotations	518	bollocks, acolyth, delirious, reject, mooch
derogatory words	2,204	scalawag, boaster, rustler, dunderheaded, pedant
felonies and words related to crime and immoral behavior	619	mafioso, roguery, robber, scalawag, rapscallion
words related to the seven deadly sins of the Christian tradition	527	concupiscience, laziness, vanity, madness, slacker

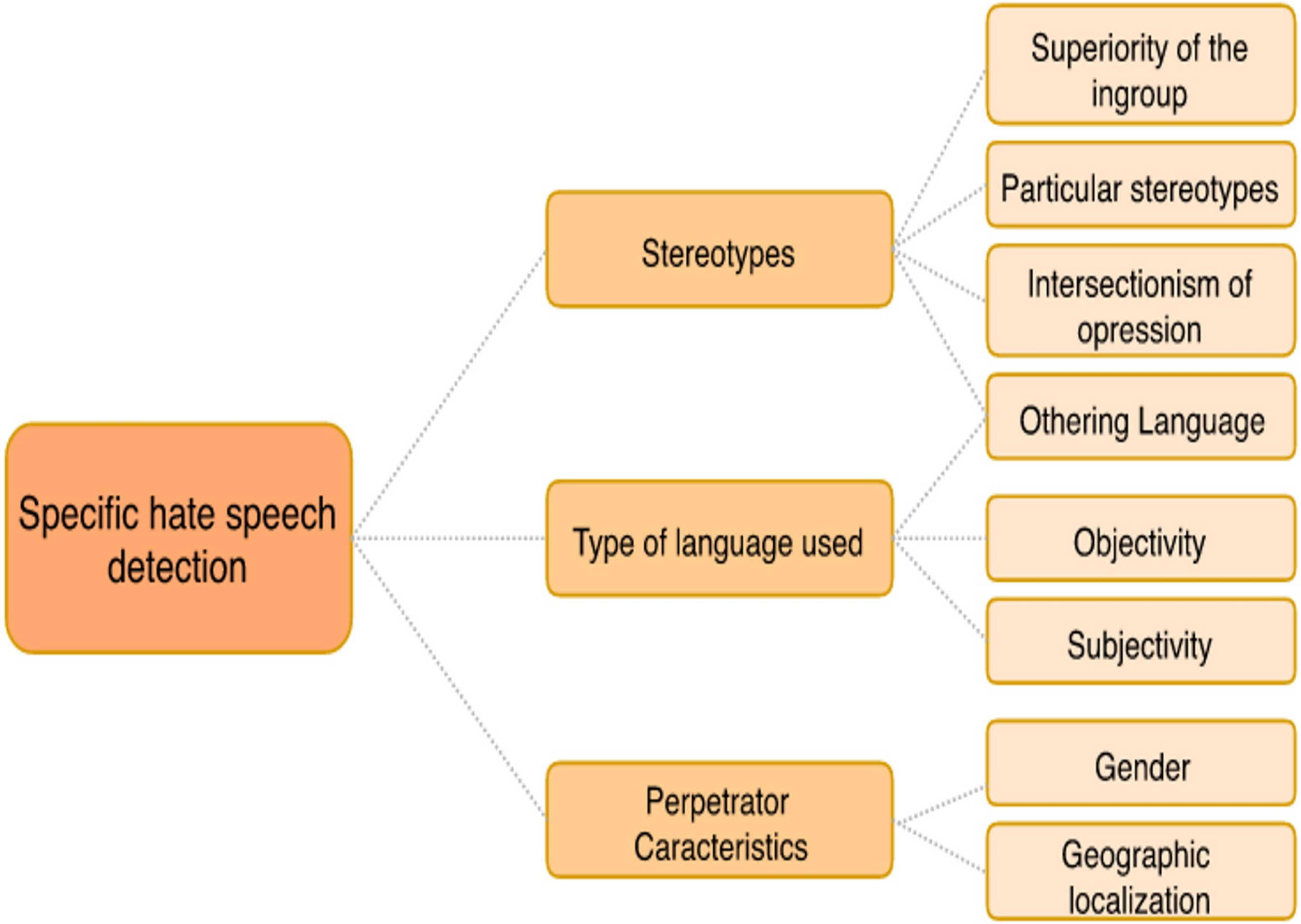
Hate Speech Detection

- Typically addressed as a text classification task
- Binary or multi-label
- Supervised



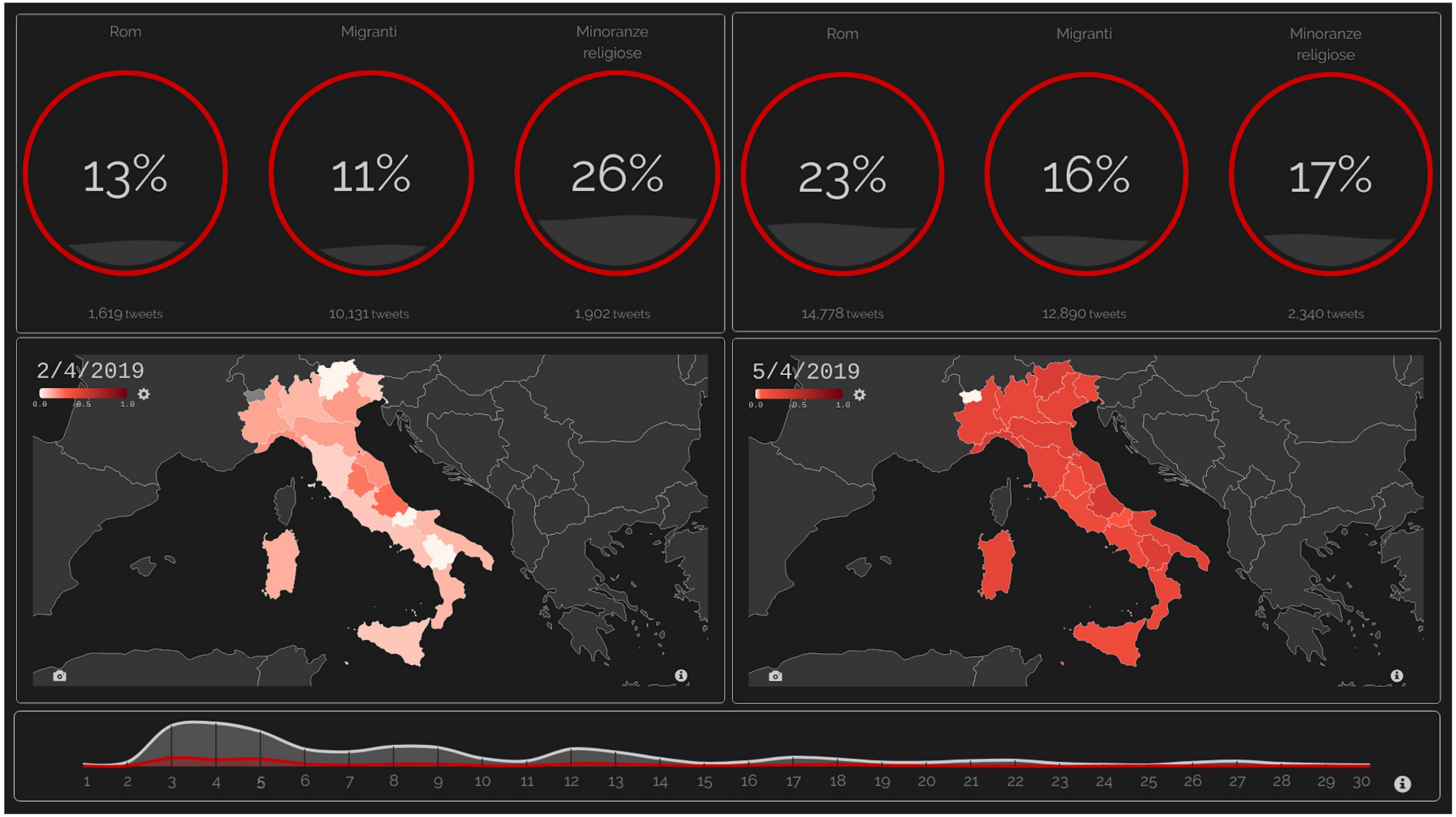
P. Fortuna, S. Nunes (2018). **A survey on automatic detection of hate speech in text.** ACM Computing Surveys (CSUR) 51.4

Specific approaches for HS detection



P. Fortuna, S. Nunes (2018). **A survey on automatic detection of hate speech in text.** ACM Computing Surveys (CSUR) 51.4

Applications: online monitoring of HS



A. T. E. Capozzi et al. (2019). **Computational linguistics against hate: hate speech detection and visualization on social media in the "Contro L'Odio" project**. In Proc. CLiC-it 2019, ceur-ws.org, vol. 2481

Summary

- Web Content Mining
  - Sentiment analysis
  - Sarcasm detection
  - Hate Speech and Offens

Next week: More Content Mining