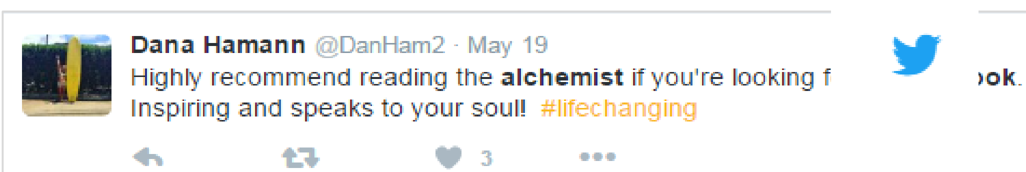
Sentiment Analysis in Textual Streams

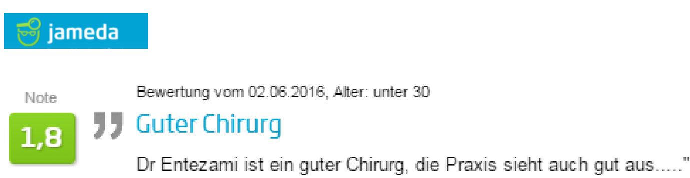
Eirini Ntoutsi

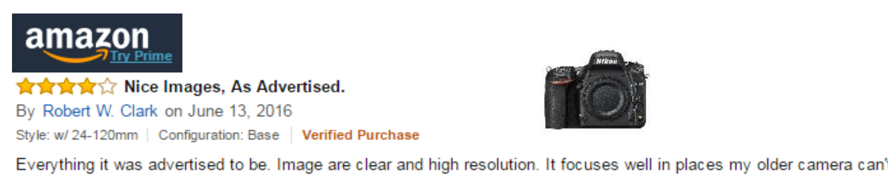
Lebniz Universität Hannover & L3S Research Center

A world of opinions

- Huge amounts of opinions are continuously published and are freely available nowadays
  - Valuable source of information for companies, decision makers, …

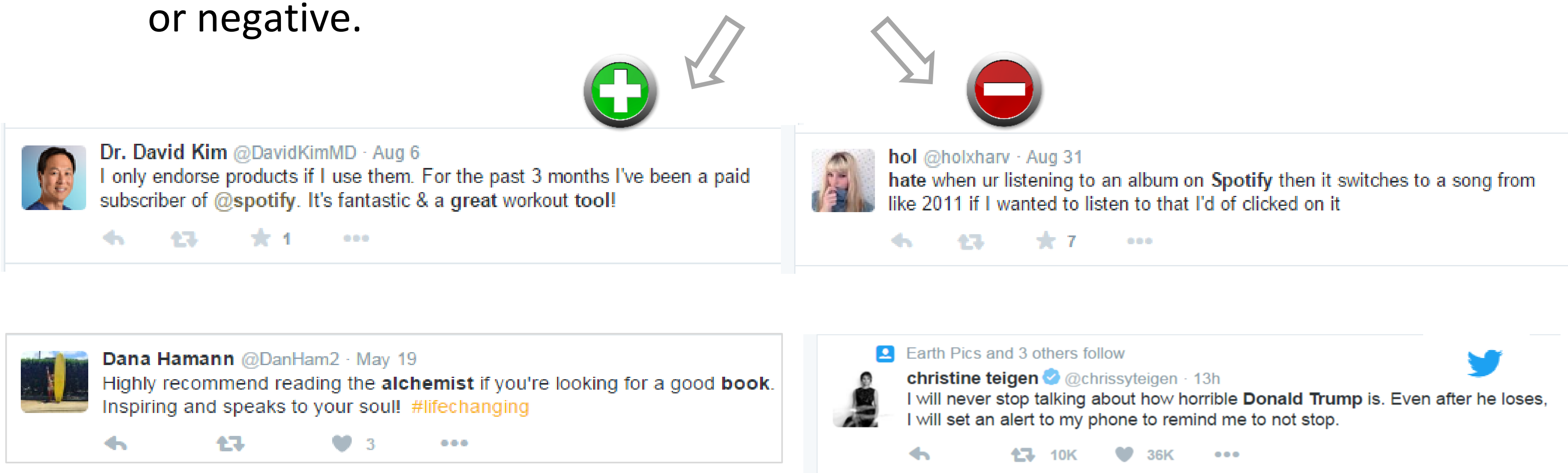






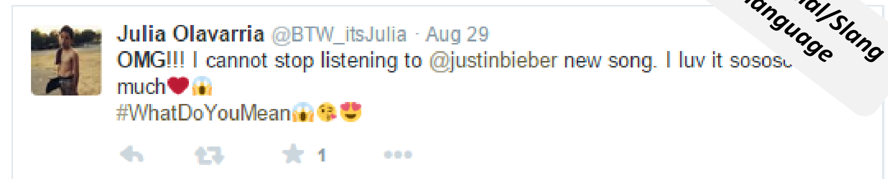
Sentiment analysis

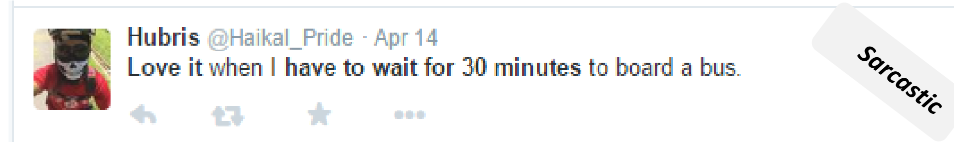
- Sentiment analysis aims at categorizing these opinions as either positive or negative.

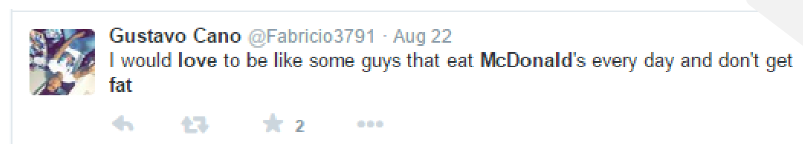


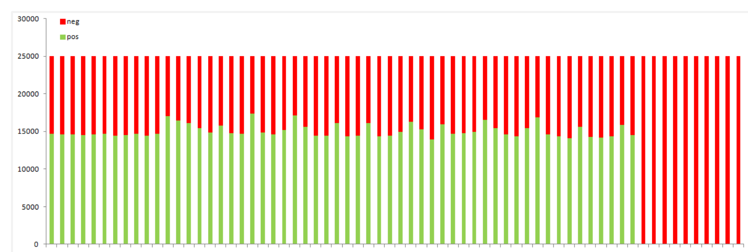
Challenges of sentiment analysis over textual streams

- A variant of text mining, opinion mining and stream mining
- Text mining related challenges
  - preprocessing, POS, high dimensionality
- Sentiment analysis related challenges
  - sentimental words vs facts
  - sarcasm
  - bipolarity
- Stream mining related challenges
  - no random access
  - non-stationary data distributions
  - shortage of class labels



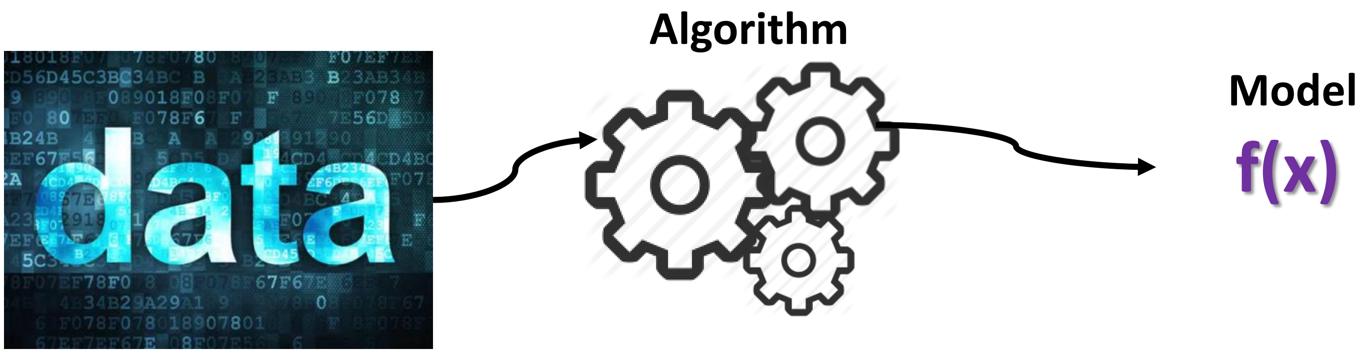






In this talk

- Building a sentiment classifier requires data and algorithms



- In this talk we will focus on
  - Learning: How to build a classifier?
  - Labeling: How to create a (class-labeled) training set?

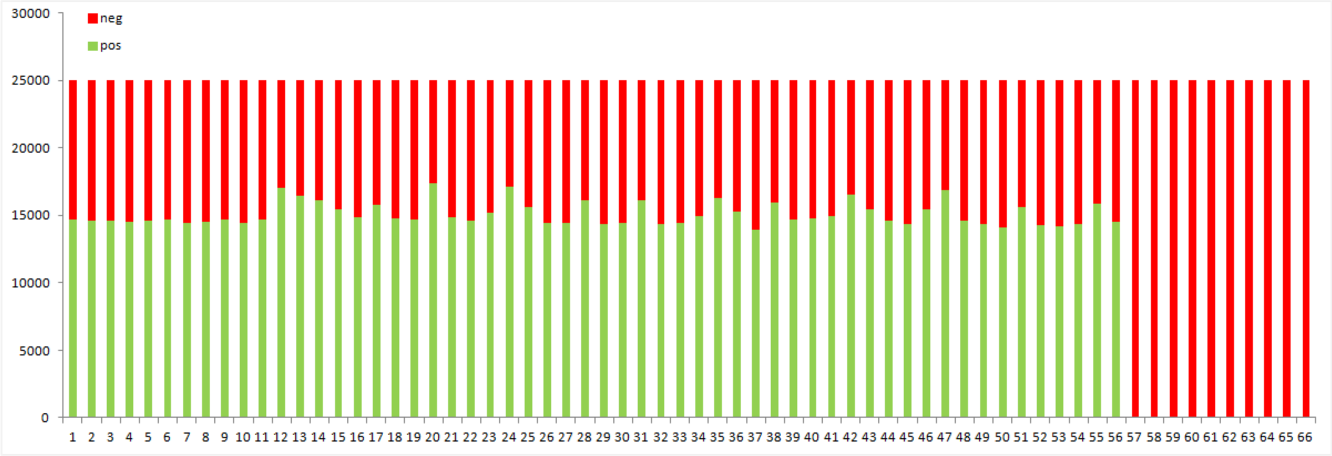
Part 1: Learning

How to build a classifier?

Case study: TwitterSentiment dataset [1]

Source: *http://help.sentiment140.com/for-students/*

- Monitoring period: 1/4/2009 – 1/7/2009 (3 months)
- 1.600.000 English tweets
- Generic stream
- Labeled (based on emoticons + ML)
- (Overall) balanced classes: 800.000 positive, 800.000 negative
  - concept drift towards the end of the stream



Fixed batch (25K) class distribution

How to build a classifier?

Preprocessing part

- Negations
- Colloquial language
- Superfluous words
- Emoticons

Learning part

Blind adaptation

- Multinomial Naïve Bayes
- Adaptive Size Hoeffding Tree
- Ensembles of Hoeffding Trees
- Stochastic Gradient Descend
- Ageing-based MNBs

- Informed adaptation

Preprocessing - Negations (TwitterSentiment)

- Tagging negations with verbs

I do not like → I NOT_like It didn't fit → It NOT_fit

→ 81.348 found negations

- Tagging negations with adjectives
  - 2-part adjective co-occurrences

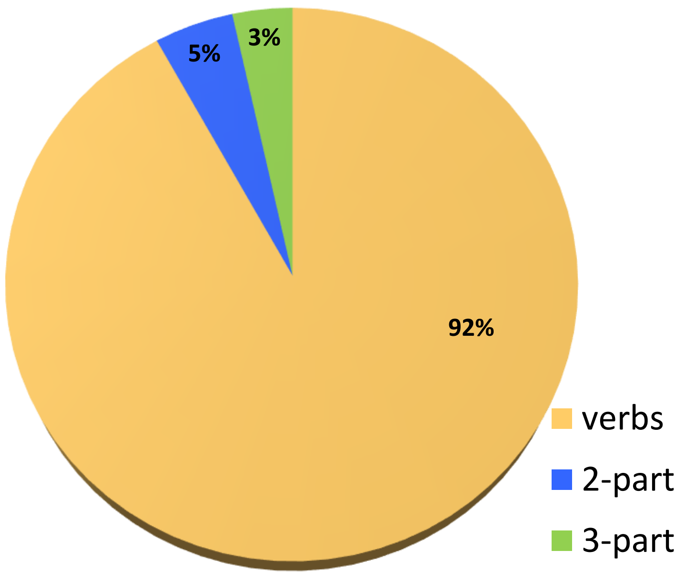
not pretty → ugly not bad → good

→ 4.074 found negations

  - 3-part adjective co-occurrences

not very young → old

→ 3.084 found negations



In total:
→ 88.506 transformations
(affecting 5% of the tweets)
→ Reduction: 0.4% words

Verbs negation list: www.vocabulix.com
Adverbs negation list: www.scribd.com

Preprocessing - Colloquial language (Twitter Sentiment)

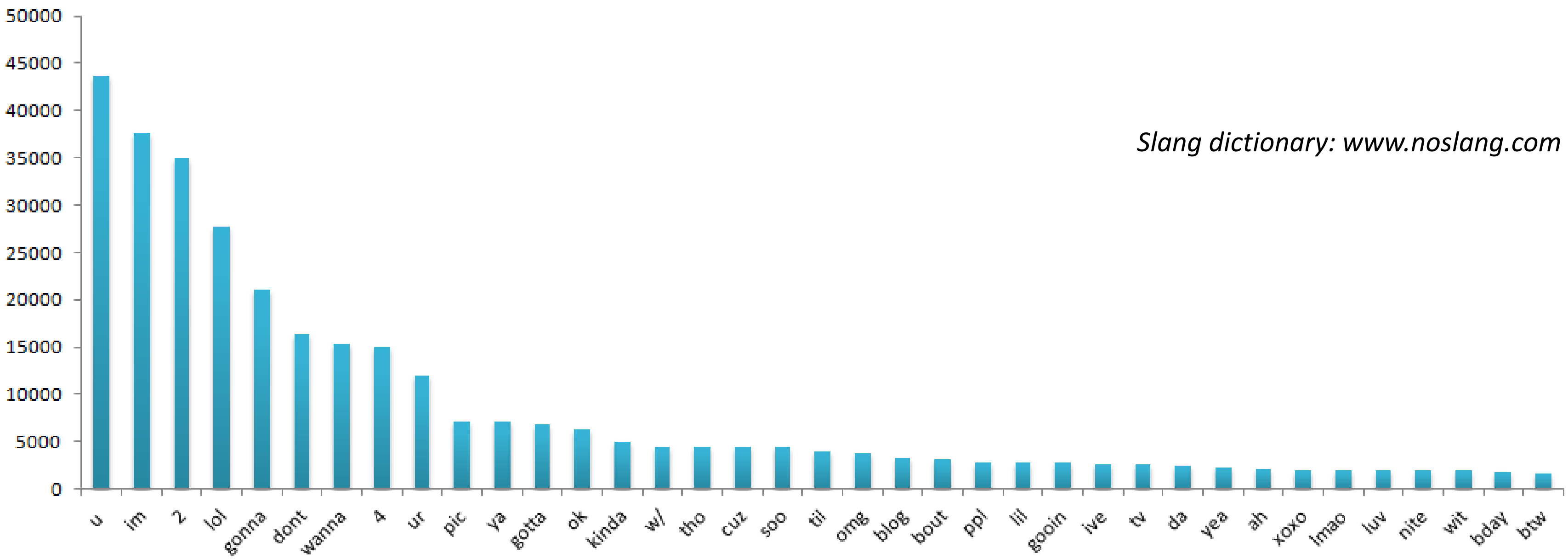
Examples:

lol → laughing out loud

xoxo → kisses and hugs

u → you

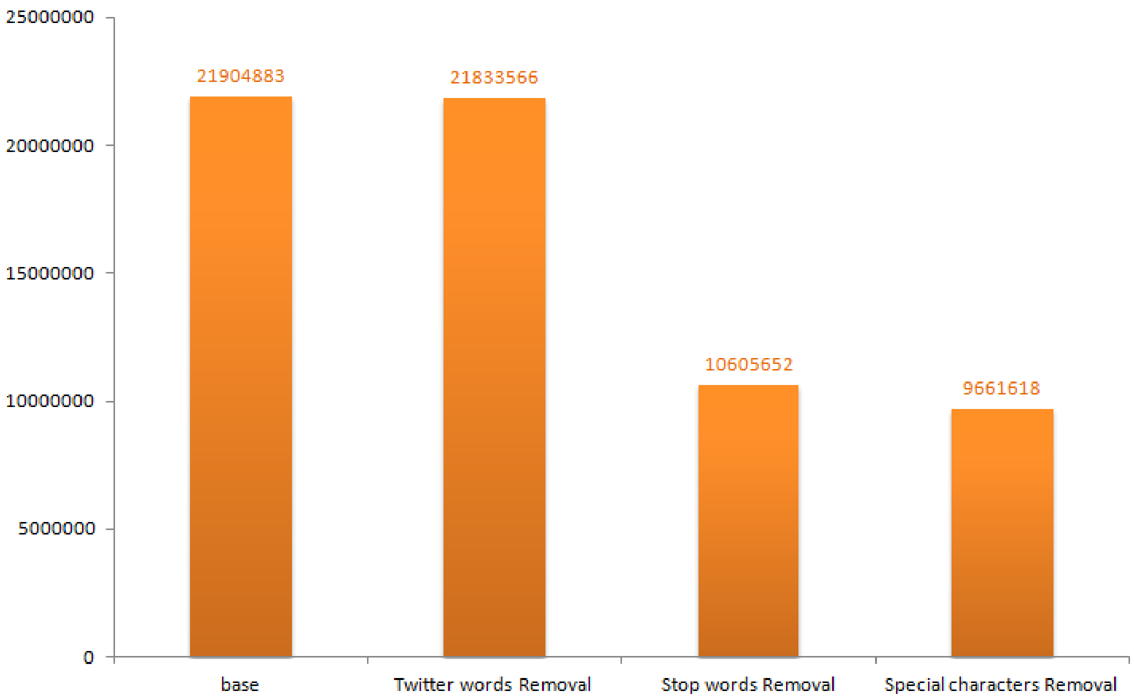
In total:
→ 499.576 transformations
(2,3% words affected)



Slang dictionary: www.noslang.com

Preprocessing - Superfluous words (Twitter Sentiment)

- Removal of Twitter special characters (@, #, RT)
- Removal of stopwords (and, for, with, about, you, me, …)
- Removal of special characters and numbers (?, %,!, 1, 2, 3, …)



In total:
→ 56% word reduction

Stopwords list from Weka:
www.cs.waikato.ac.nz/ml/weka

Preprocessing - Emoticons (Twitter Sentiment)

Examples:

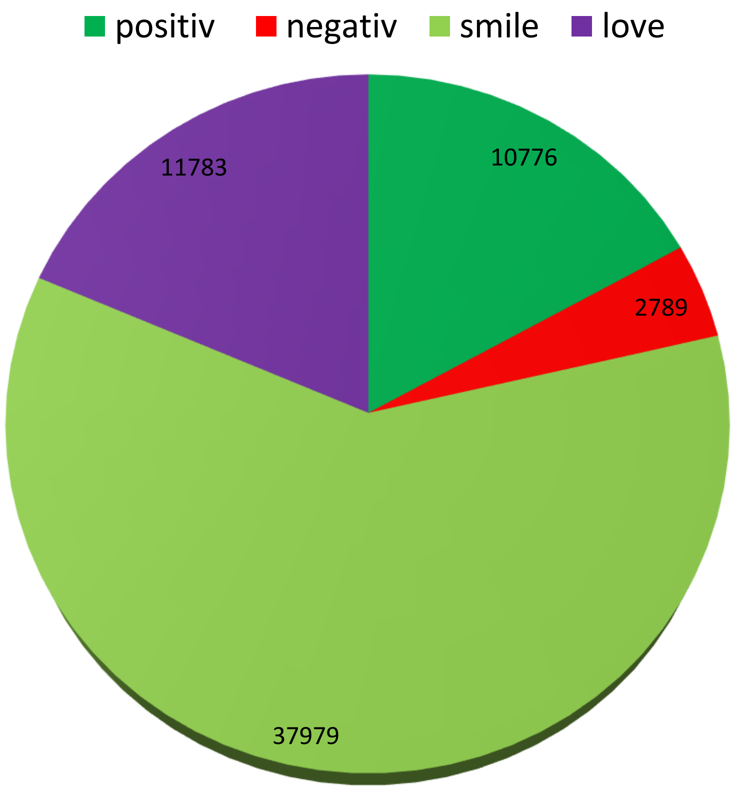
:-) :) :o) =) ;) (: (; (= → "positive"

:(:-(:o(=(;(;-( ):);)= → "negative"

:D :-D :oD =D ;D → "smile"

<3 → "love"

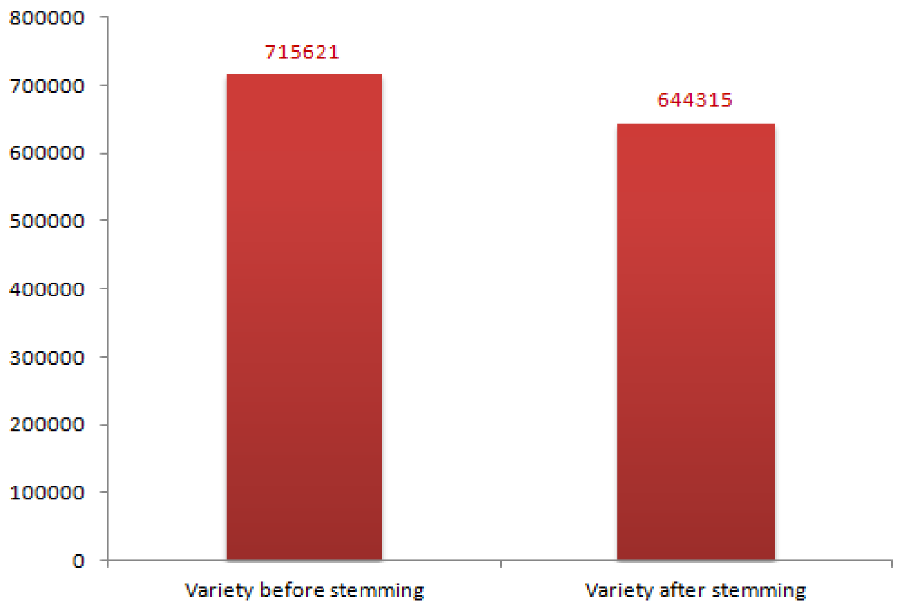
In total:
→ 63.327 emoticons found (0.3%)



Preprocessing – Stemming (TwitterSentiment)

Examples:

monitoring, monitored, monitor → monitor
fishing, fishes, fish → fish



→ 10% variety reduction

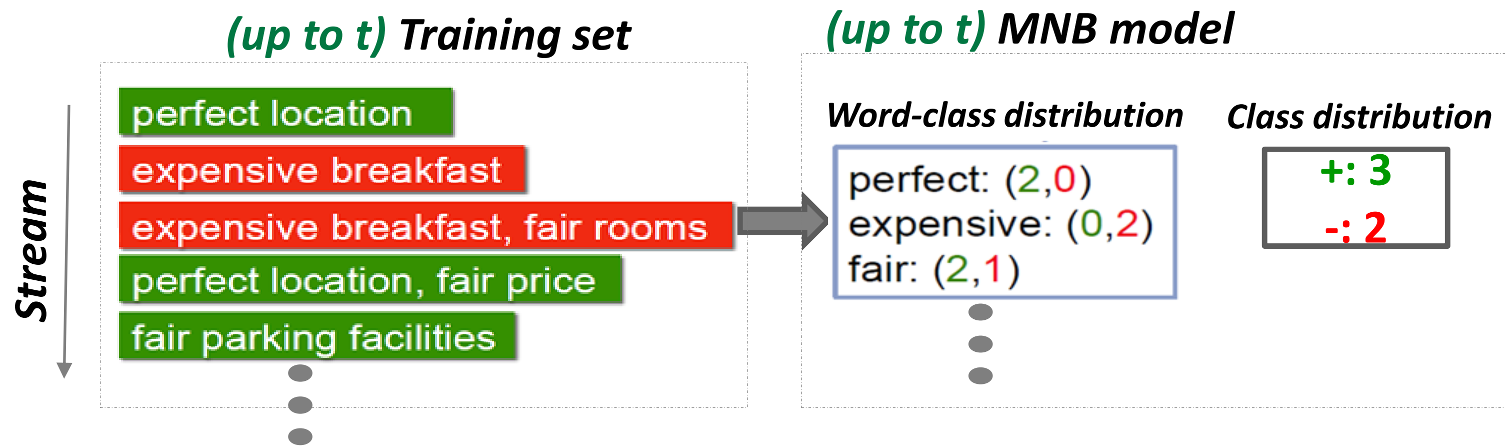
Porter stemmer from WEKA:
www.cs.waikato.ac.nz/ml/weka

Learning algorithms (TwitterSentiment)

- A variety of online learners
  - Multinomial Naive Bayes (MNB)
    - Naïve Bayes classifiers modeling word occurrences
  - Adaptive Size Hoeffding Tree (ASHT)
    - Decision tree with a Hoeffding bound and of limited size
  - Ensemble of Adaptive Size Hoeffding Trees (OzaBag ASHT)
    - Ensemble of different sized ASHT
  - Stochastic Gradient Descent (SGD)
    - A linear classifier optimizing a loss function

Experiments were conducted in MOA
- Extension of WEKA for data streams
- Available at: moa.cs.waikato.ac.nz

Multinomial Naïve Bayes



- Prediction for a new document *d*: based on model counts up to *t*:

$$P^t(c|d) = P^t(c) \prod_{i=1}^{|d|} P^t(w_i|c)^{f_i^d}$$

$$\hat{P}^t(c) = \frac{N_c^t}{|\mathcal{D}|^t} \qquad \hat{P}^t(w_i|c) = \frac{N_{ic}^t}{\sum_{j=1}^{|V|^t} N_{jc}^t}$$

Accumulated counts from the beginning of the stream

- Model update:
  - New observations are accumulated
  - Nothing is forgotten → accumulativeMNB

Adaptive Size Hoeffding Tree (ASHT)

- Hoeffding tree, a decision tree for data streams
  - A small sample could be sufficient to choose an optimal splitting attribute
  - Hoeffding bound: With probability 1-δ, the true mean of variable r is at least r_μ-ε, where

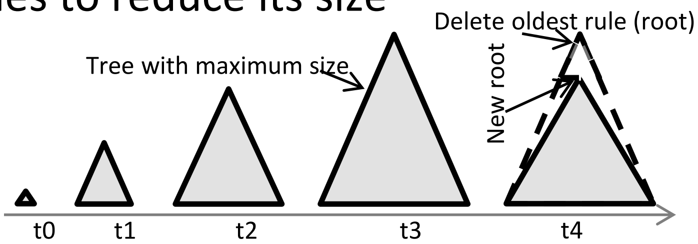
$$\epsilon = \sqrt{\frac{R^2 ln(1/\delta)}{2n}}$$

n: # observations

R: range of the variable

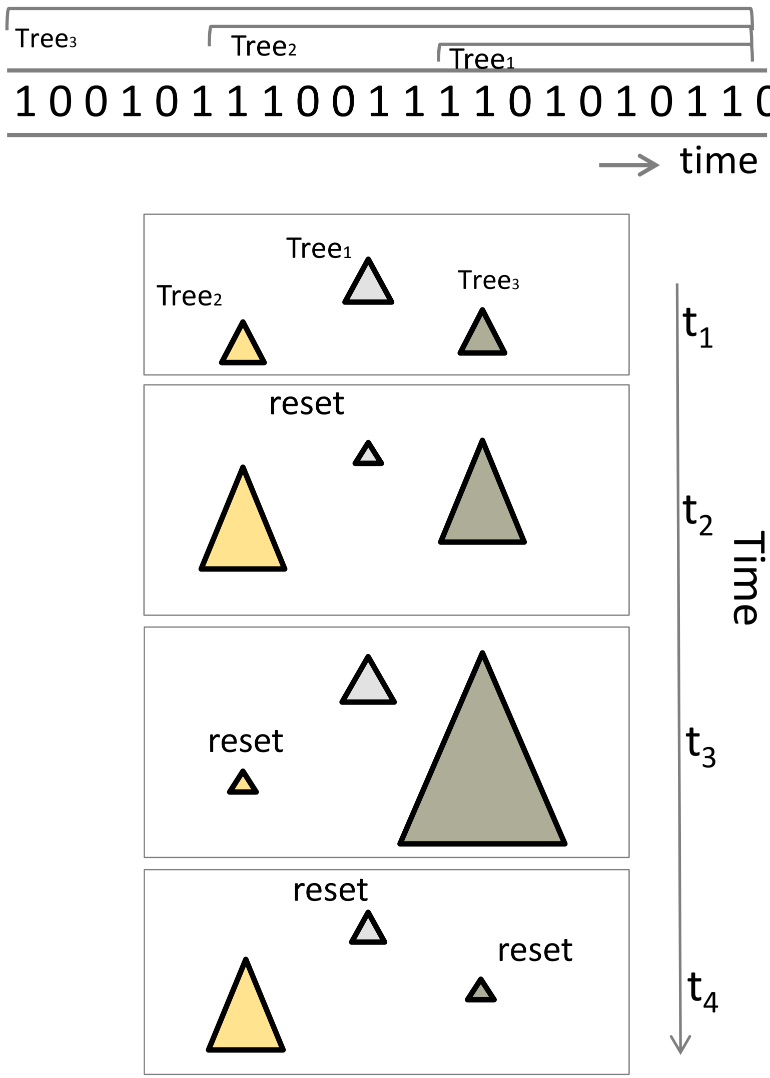
r_μ: computed mean of r

- Adaptive Size Hoeffding Tree (ASHT)
  - The tree has a maximum size (# of splitting nodes)
  - After one node splits, if the number of split nodes of the ASHT is higher than the maximum value, then it deletes some nodes to reduce its size
- Model update:
  - New observations are incorporated
  - (part of the) old model is deleted, due to size limit



Ensemble of Adaptive Size Hoeffding Trees (OzaBagASHT)

- Bagging using ASHTs of different sizes
- The max size of the n^{th} ASHT is twice the max size of the $(n-1)^{th}$ tree.
- Allows building models for different time-frames
  - Smaller trees react faster to change, larger trees slower
  - Larger trees perform better during periods with no or little change
- Model update
  - New observations are incorporated
  - Old ASHTs are deleted, due to size limit



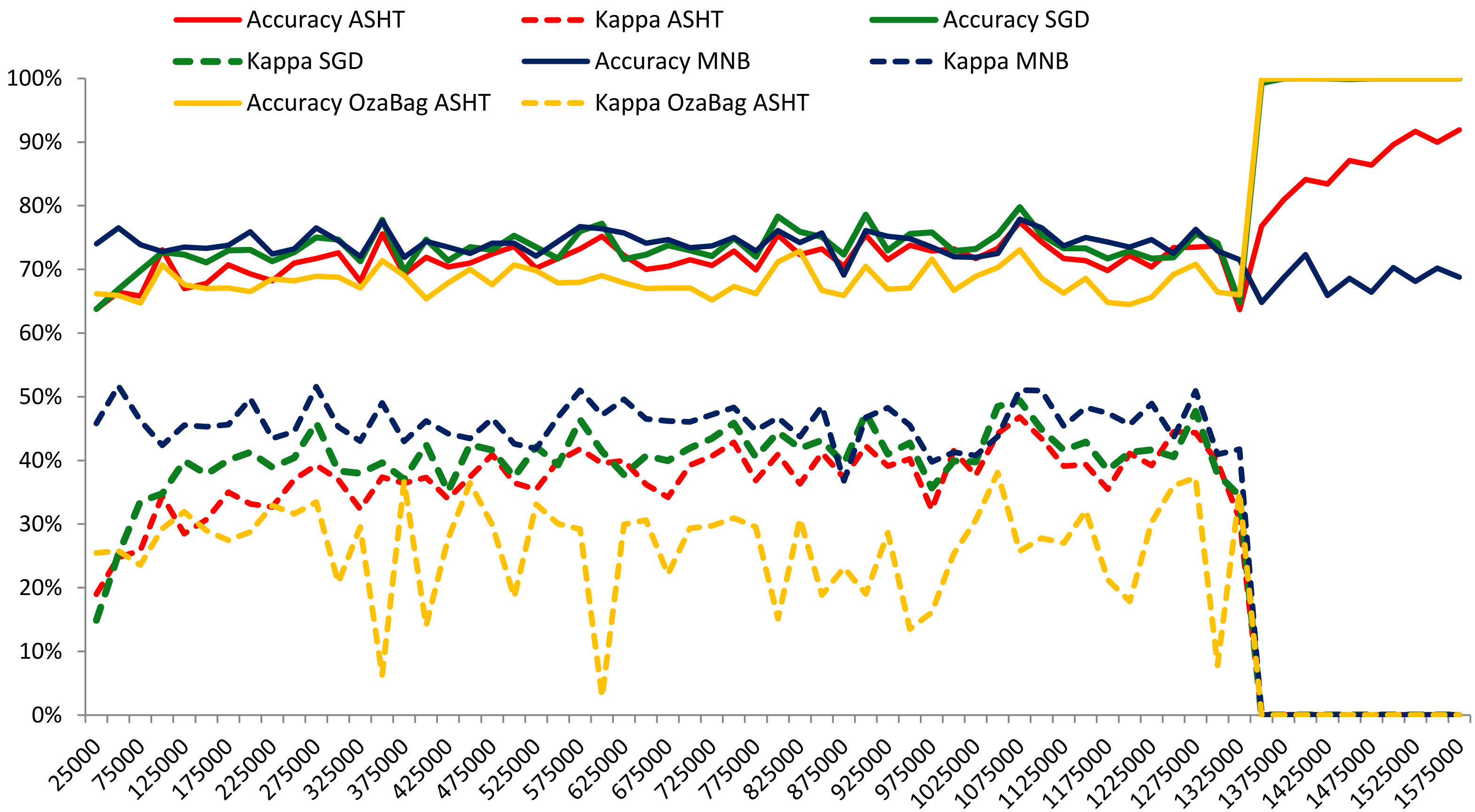
Stochastic Gradient Descent (SGD)

- A gradient descend optimization method for minimizing an objective function
  - In our case we want to minimize the loss, i.e., the cost of predicting $\hat{y}$ when actual answer is *y*. $\ell(\hat{y}, y)$
  - We are looking for a function *f* parameterized by a weight vector *w* that minimizes the loss $Q(z, w) = \ell(f_w(x), y)$ averaged on the examples
- Typically, the gradient of the objective loss function is computed using all training examples, and is used to adjust the parameters.

$$w_{t+1} = w_t - \gamma \frac{1}{n} \sum_{i=1}^{n} \nabla_w Q(z_i, w_t)$$

- Stochastic gradient descent is a simplification, as it estimates the gradient on the basis of single instances
- Model update $\quad w_{t+1} = w_t - \gamma_t \nabla_w Q(z_t, w_t)$
  - New instances are incorporated and parameters are adjusted

Prequential evaluation - “test then train“ results



→ MNB & SGD reach best results when the class distribution is stable

→ OzaBag ASHT & SGD can deal best with distribution changes

→ Single ASHT also has problems with adaptation (still better than MNB)

Ageing-based MNB [2]

- A temporal model that keeps track of the last time that an observation is made in the stream
  - For classes: $(N_c) \rightarrow (N_c, \boxed{t_{lo}^{c}})$ *last class observation time in the stream*
  - For word-class pairs: $(N_{ic}) \rightarrow (N_{ic}, \boxed{t_{lo}^{ic}})$ *last word-class observation time in the stream*
  - Timestamp propagation: from documents → classes, word-class pairs
  - Temporal de-coupling of words from documents
    - Observation updates might come from different documents
  - Allows differentiation of the observations based on their recency

Ageing-based MNB

- Gradual ageing – exponential ageing function

$$age(o,t) = e^{-\lambda(t-t_o)}$$

t: current time
t_o: object's arrival time
λ: the decay rate

  - higher λ, less important the historical data
  - Points are halved every *1/λ* timeunits

- Updated temporal probability estimates

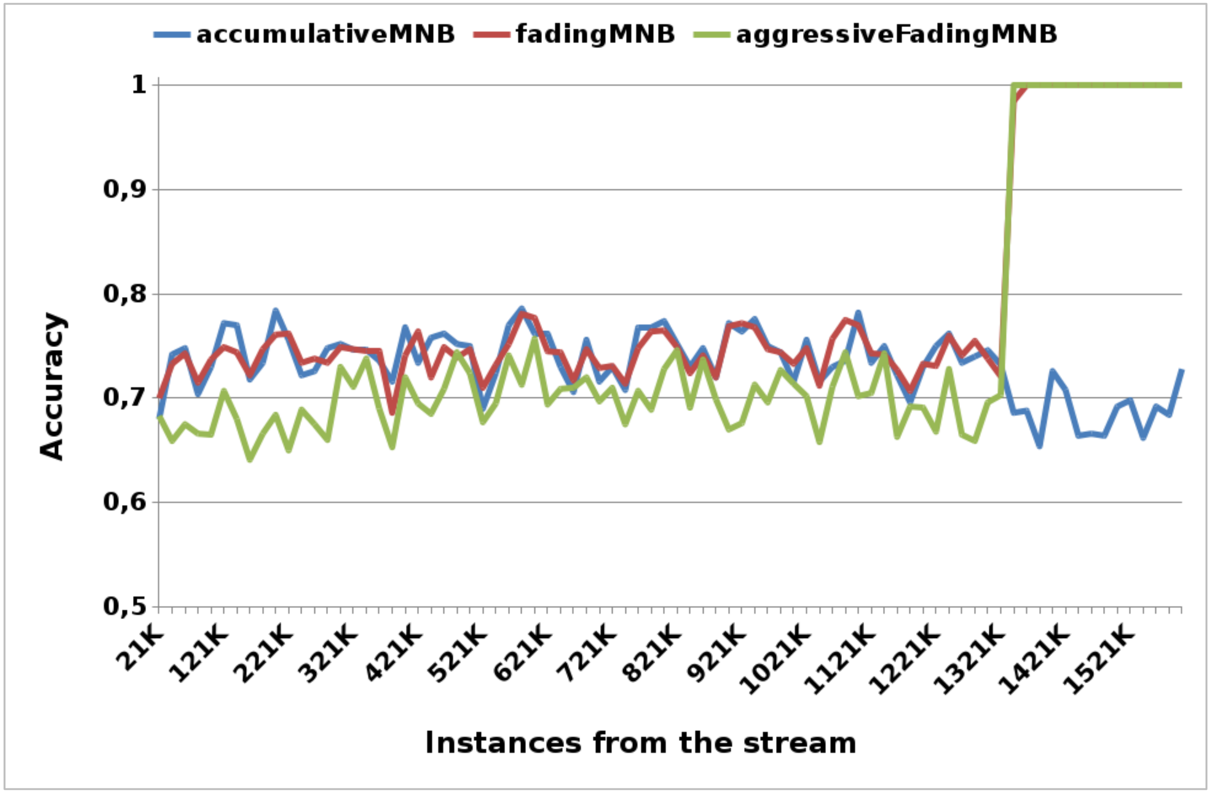
$$\hat{P}^t(c) = \frac{N_c^t * \boxed{e^{-\lambda\cdot(t-t_{lo}^c)}}}{|\mathcal{S}^t|}$$

$$\hat{P}^t(w_i|c) = \frac{N_{ic}^t * \boxed{e^{-\lambda\cdot(t-t_{lo}^{(w_i,c)})}}}{\sum_{j=1}^{|V^t|} N_{jc}^t * \boxed{e^{-\lambda\cdot(t-t_{lo}^{(w_j,c)})}}}$$

ageing effect

Prequential evaluation

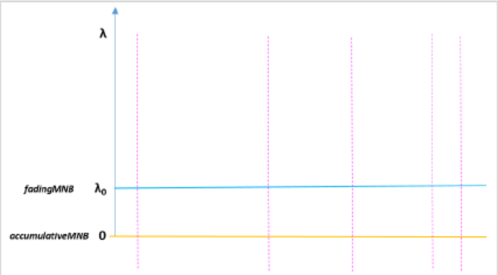
- Hourly-aggregated stream, λ=0.1, evalW=1.000



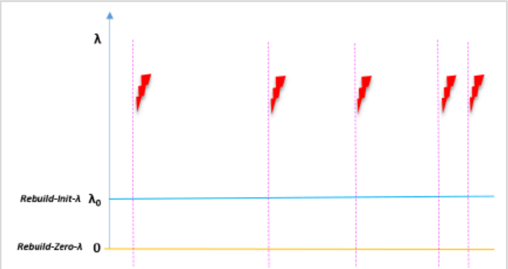
→ Ageing helps model recovery in times of change
→ Gradual fading maintains a good performance in times of stability

Informed adaptation [In progress, with Vasilis Iosifidis]

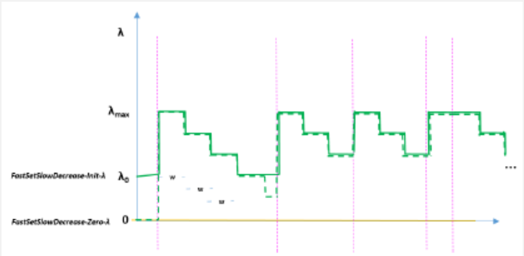
- Adapt the model, when change is detected
- Change detection + adaptation upon change
- Different adaptation strategies
  - Model rebuild
  - Tuning of the ageing factor lambda
    - Abrupt tuning
    - Gradual tuning



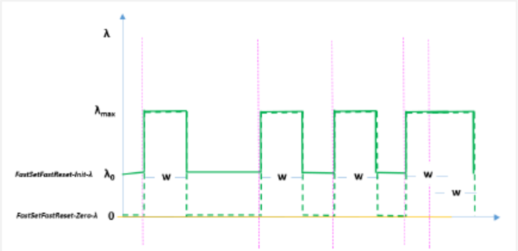
(f) *fadingMNB* and *accumulativeMNB*



(e) *Rebuild*

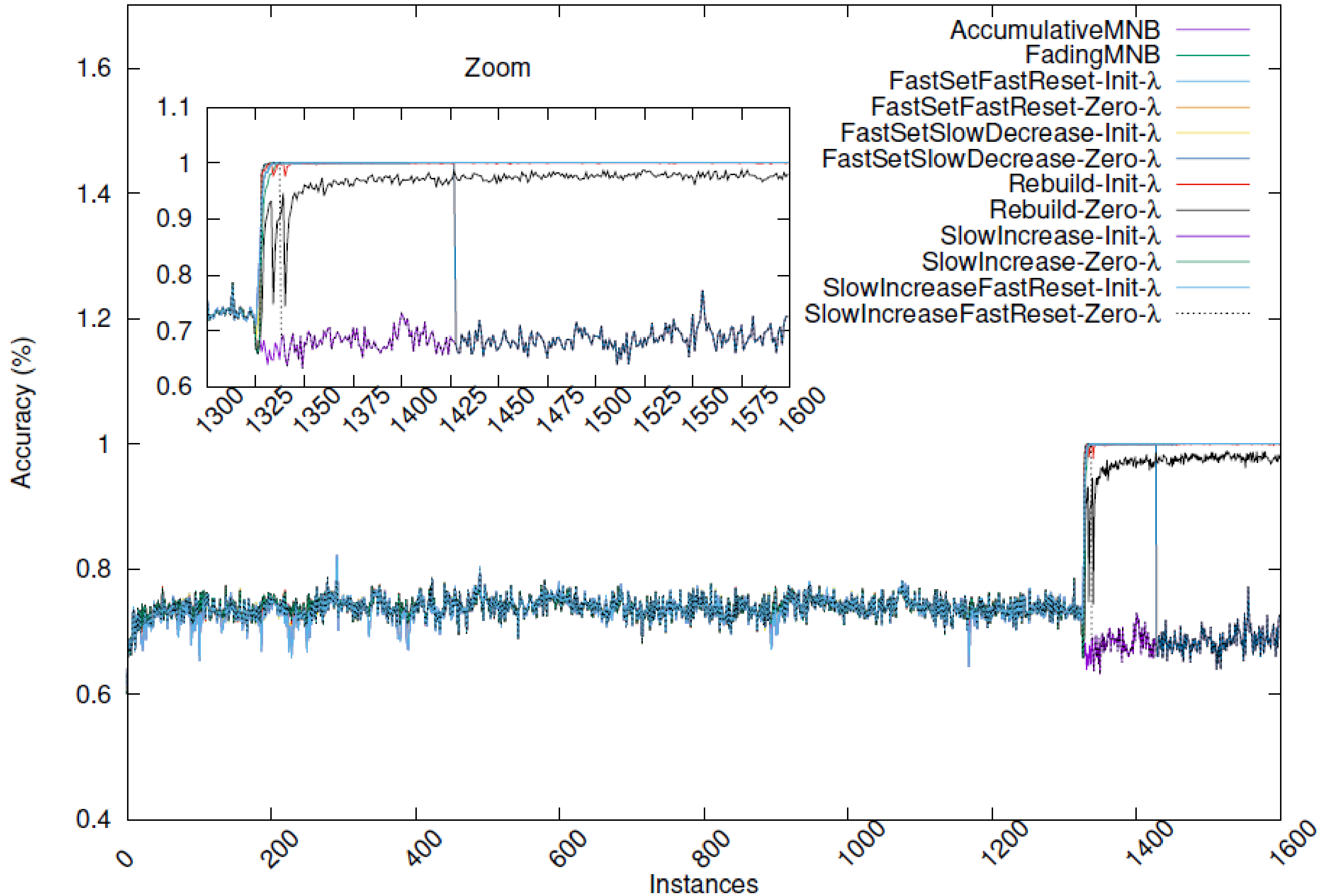


(d) *FastSetSlowDecrease*



(c) *FastSetFastReset*

Prequential evaluation



→ No drastic improvement, but such an approach also informs for change

→ A (small constant) ageing is beneficial even for the "model-rebuild" strategy

Part 2: Labeling

How to create a (class-labeled) training set?

L3S Twitter dataset [In progress, with Vasilis Iosifidis]

- Monitoring period: 1/2/2013 – ...
- 5,405,890,231 tweets (on 29.8.2016)
- Generic stream (1% Twitter sample)
- No labels

- Goal: Sentiment annotation of the collection in order to
  - better understand (specific aspects of) the collection
  - provide datasets for stream mining
- Babysteps: Creating a training set from the 2015 subset
  - 1.882.387.310 tweets in total
  - 486.721.724 tweets in English → 26%
  - 6.052.433.618 words

L3S^{2015} Twitter dataset preprocessing

Preprocessing part

- Negations
- Colloquial language
- Superfluous words
- Emoticons

Labeling part

- Emoticons
- SentiWordNet
- Learning (ML) again

Preprocessing - Negations (L3S^{2015})

- Tagging negations with verbs

I do not like → I NOT_like
It didn't fit → It NOT_fit

→ 27.222.287 found verb negations (0.4%)

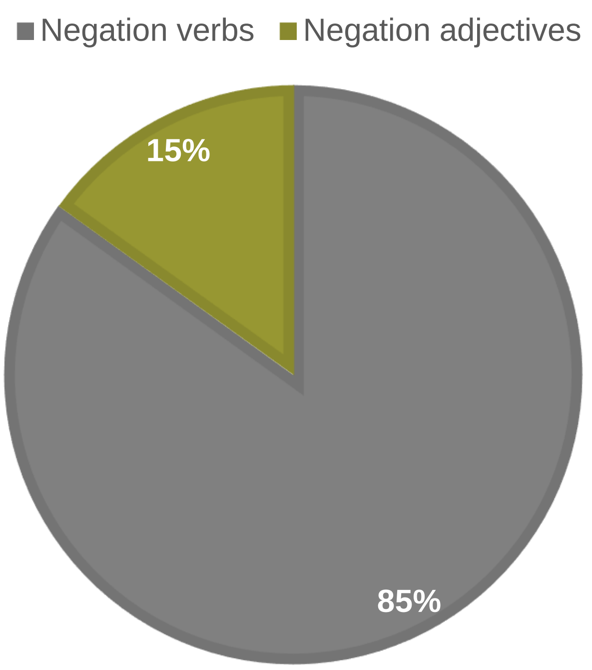
- Tagging negations with adjectives
  - 2-part adjective co-occurrences

not pretty → ugly
not bad → good

  - 3-part adjective co-occurrences

not very young → old

→ 4.832.573 found adjective negations (0.1%)



Verbs negation list: www.vocabulix.com
Adverbs negation list: www.scribd.com

Preprocessing - Colloquial language (L3S^{2015})

Examples:

lol → laughing out loud

xoxo → kisses and hugs

u → you

a.i.m. → aol instant messanger

→1st application (-)83.642.045 transformations (1,4%)

- After removing links, mentions (@user), # ," .!?_ "

→ 2nd application (-)19.421.885 transformations (0,3%)

→ Total (-)103.063.930 transformations (1,7%)

Slang dictionary: www.noslang.com

Preprocessing - Superfluous words & Emoticons (L3S-2015)

- Removal of links
- Removal of mentions (@userX)
- Removal of special characters # ," .!?_ "

→ total 563.334.403 entries removed (9.3%)

- Removal of stopwords

→ Total 1.167.307.795 entries removed (19,3%)

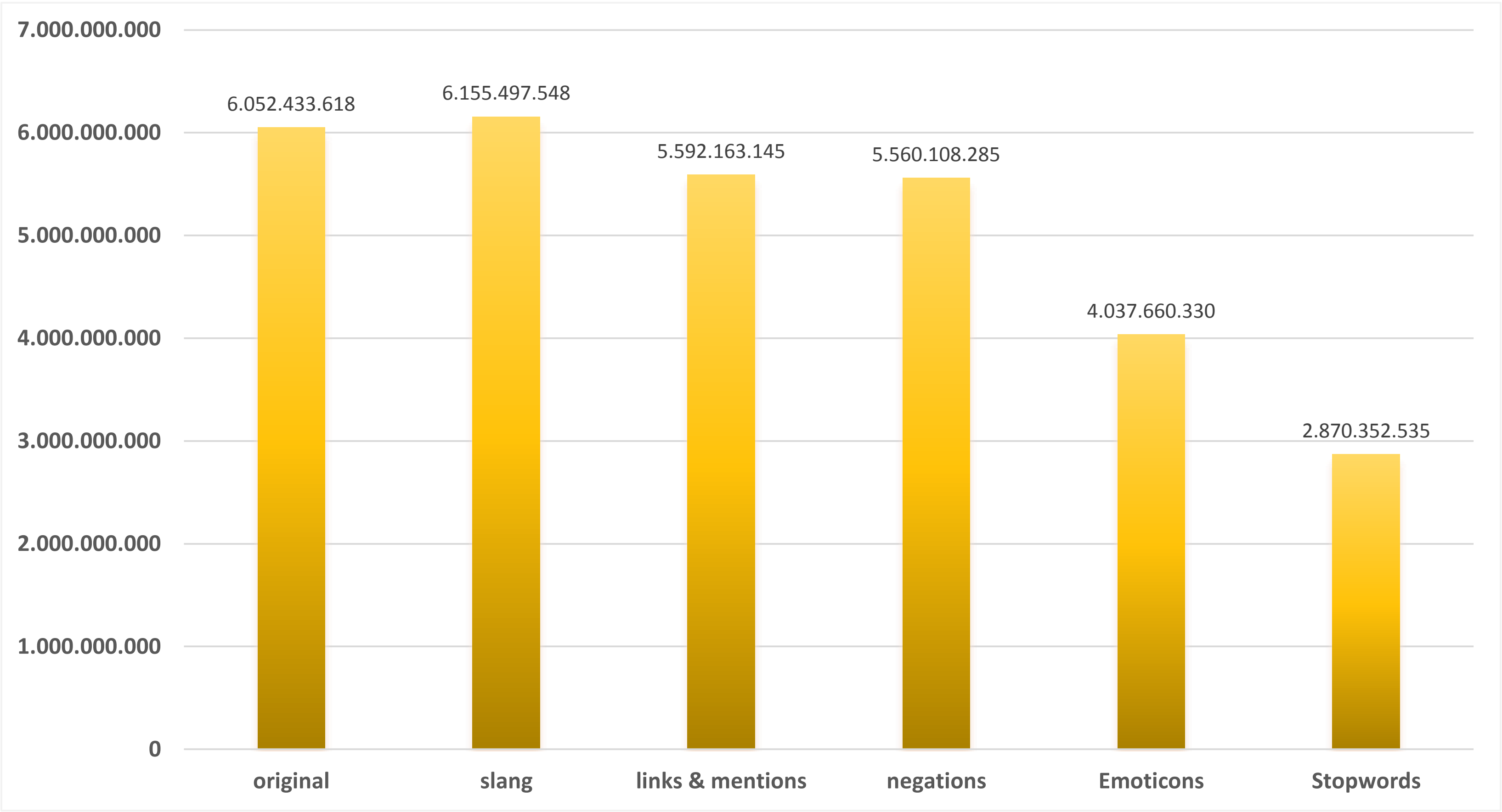
Stopwords list from WEKA
www.cs.waikato.ac.nz/ml/weka

- Removal of emoticons (142 emoticons considered).
- Removal of RT, numbers
- Removal of small words (<2 chars)

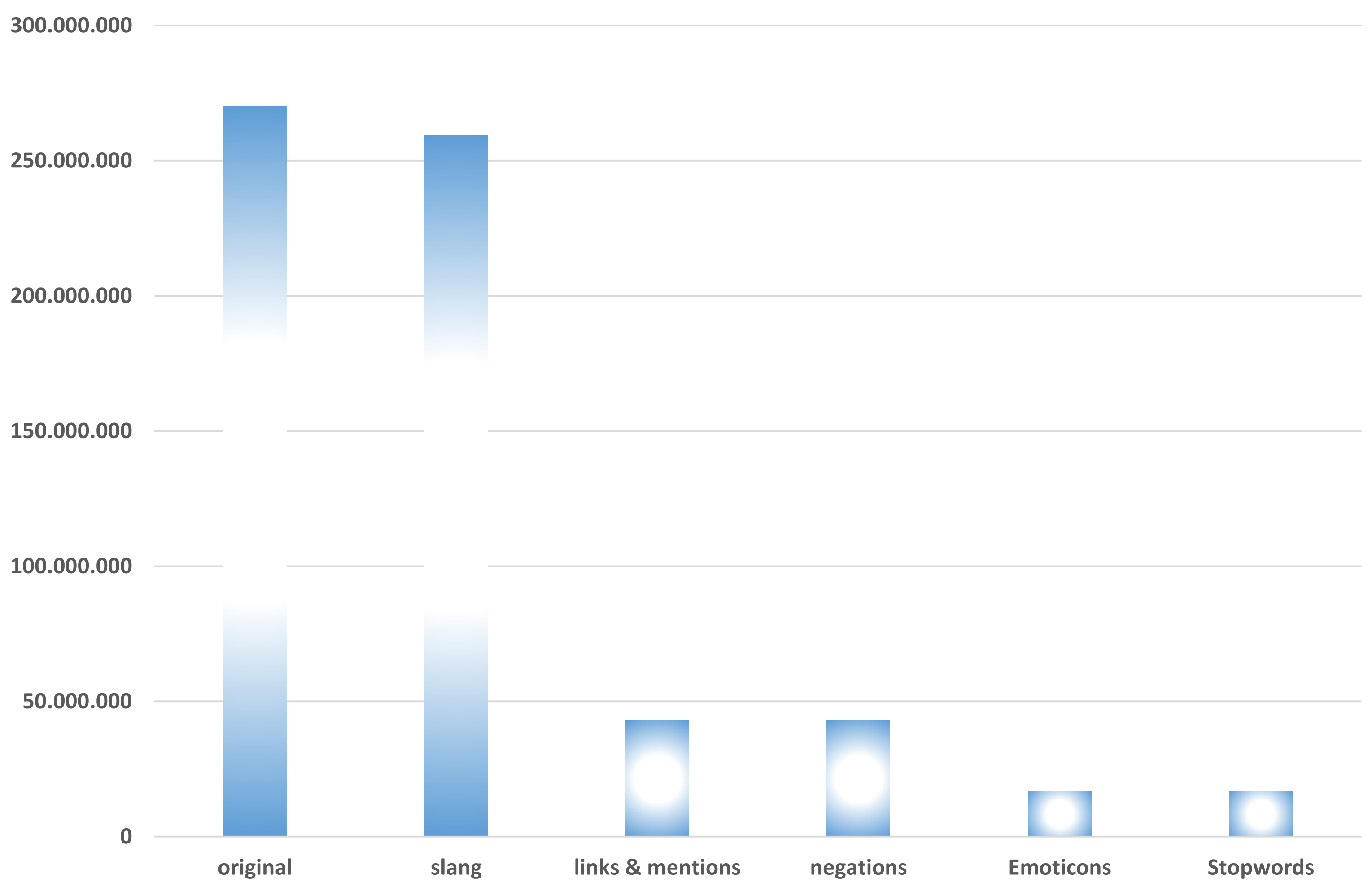
→1.522.447.955 entries removed (25,2%)

Emoticons list:
- *https://en.wikipedia.org/wiki/List_of_emoticons*
- *https://github.com/wooorm/emoji-emotion/blob/master/data/emoji-emotion.json*

Preprocessing effect – Overall view (L3S-2015)



Preprocessing effect – Overall view (distinct words) (L3S-2015)



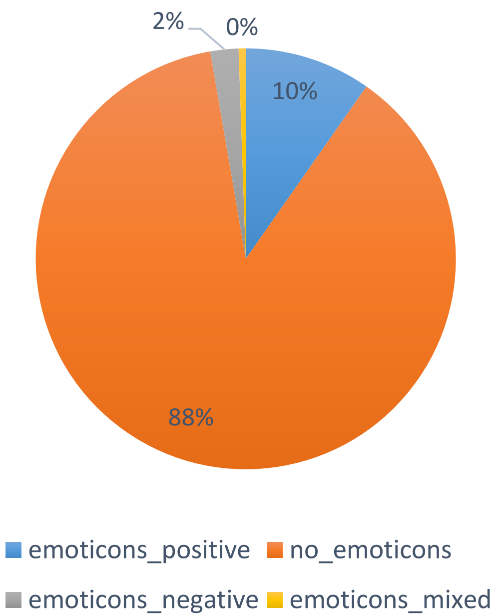
Labeling part

- Human labeling is impossible at this scale → machine-based
- Two approaches thus far
  - Labels through emoticons
  - Labels through sentiment dictionaries (SentiWordNet)

Labels through emoticons

- We assembled a list of positive, negative emoticons
  - #72 positive class emoticons :-) :) :o) =) ;) (: (; (= <3 :D :-D :oD =D ;D
  - #70 negative emoticons :(:-(:o(=(;(;-( ):);)=
- We classified tweets based on their emoticons
  - positive ← only positive emoticons (10%)
  - negative ← only negative emoticons (2%)
  - Mixed ← both positive and negative (1%)
  - No emoticon (88%)

→ In total, 57.340.286 (12%) are pure-labeled.



Emoticons vs SentiWordNet

- SentiWordNet: a lexical resource for supporting sentiment classification
- Sentiment of a tweet as an aggregation of the sentiment of its words
- For the intersection (57.340.286 = 12% tweets with pure sentiment-based labels), we checked agreement in the labels

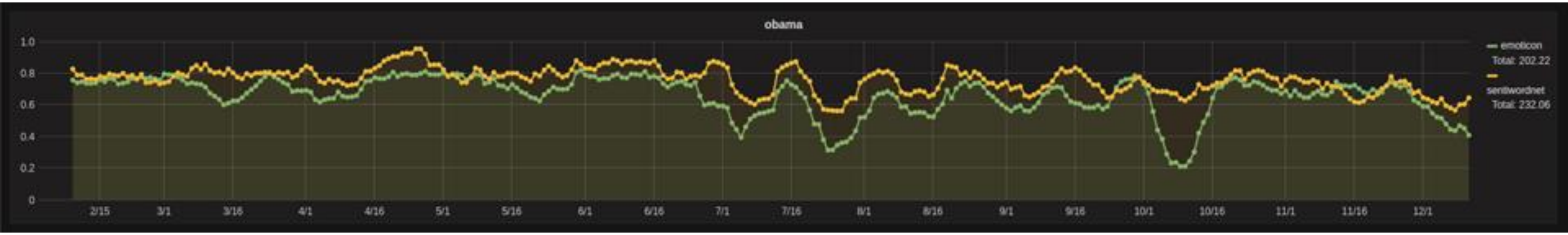
Emoticon-based labeling	SentiWordNet-based labeling				
	Positive	**Negative**	**Neutral**	**Zero sum**	**No-decision**
Positive	28.104.677 *(49%)*	10.756.225 *(19%)*	4.908.237 *(9%)*	23.297 *(0.04%)*	3.140.978 *(5%)*
Negative	4.929.947 *(9%)*	3.885.983 *(7%)*	930.075 *(2%)*	7.527 *(0.01%)*	653.340 *(1%)*

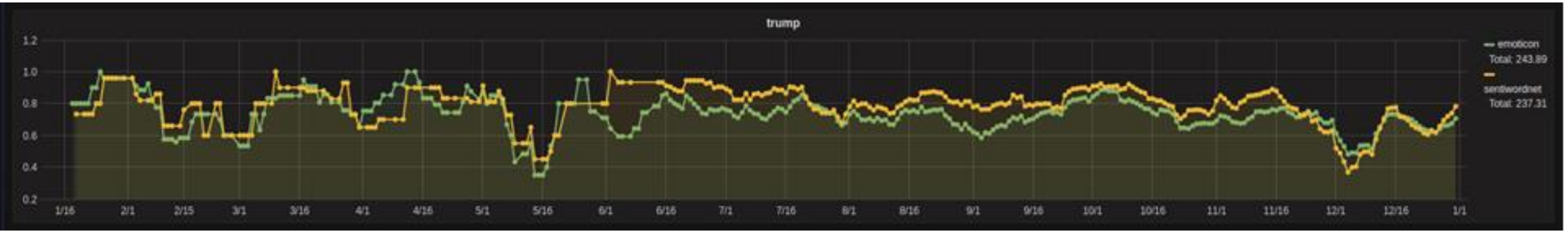
- SentiWordNet labeling results
  - Positive, Negative: overall positive, negative
  - No decision: words do not exist in the lexicon, e.g., #Iloveobama, #refugeecrisis etc
  - Neutral: neutral words (also non-existing).
  - Zero-sum: mix of positive and negative

Causes of disagreement

- Emoticons-based labeling
  - Prone to errors: existence of positive emoticons does not imply positive words
- SentiWordNet-based labeling
  - SentiWordNet is a static dictionary
  - Twitter is very dynamic
    - Words change polarity (also based on context)
    - New words are created (e.g. hashtags) which are not part of the dictionary

SentiWordNet-based vs Emoticon-based labeling examples





How to proceed/ What is the ground truth?

- Trust only one source (emoticons or sentiwordnet)
- Use only tweets for which both emoticon-based and sentiwordnet-based labels agree → smaller set, but probably less noisy in terms of labels
- Next step:
  - Semi-supervised learning of the labels based on an initial labeled seed set
    - Emoticon-based
    - Sentiword-based
    - Intersection

Challenges & Opportunities: Data asquition

- Multilinguality
  - For the L3S-2015 dataset:
    - 486.627.464 (English tweets) out of 1.882.387.310 total tweets → we utilize only 26% of the dataset.

→ Add multilingual content

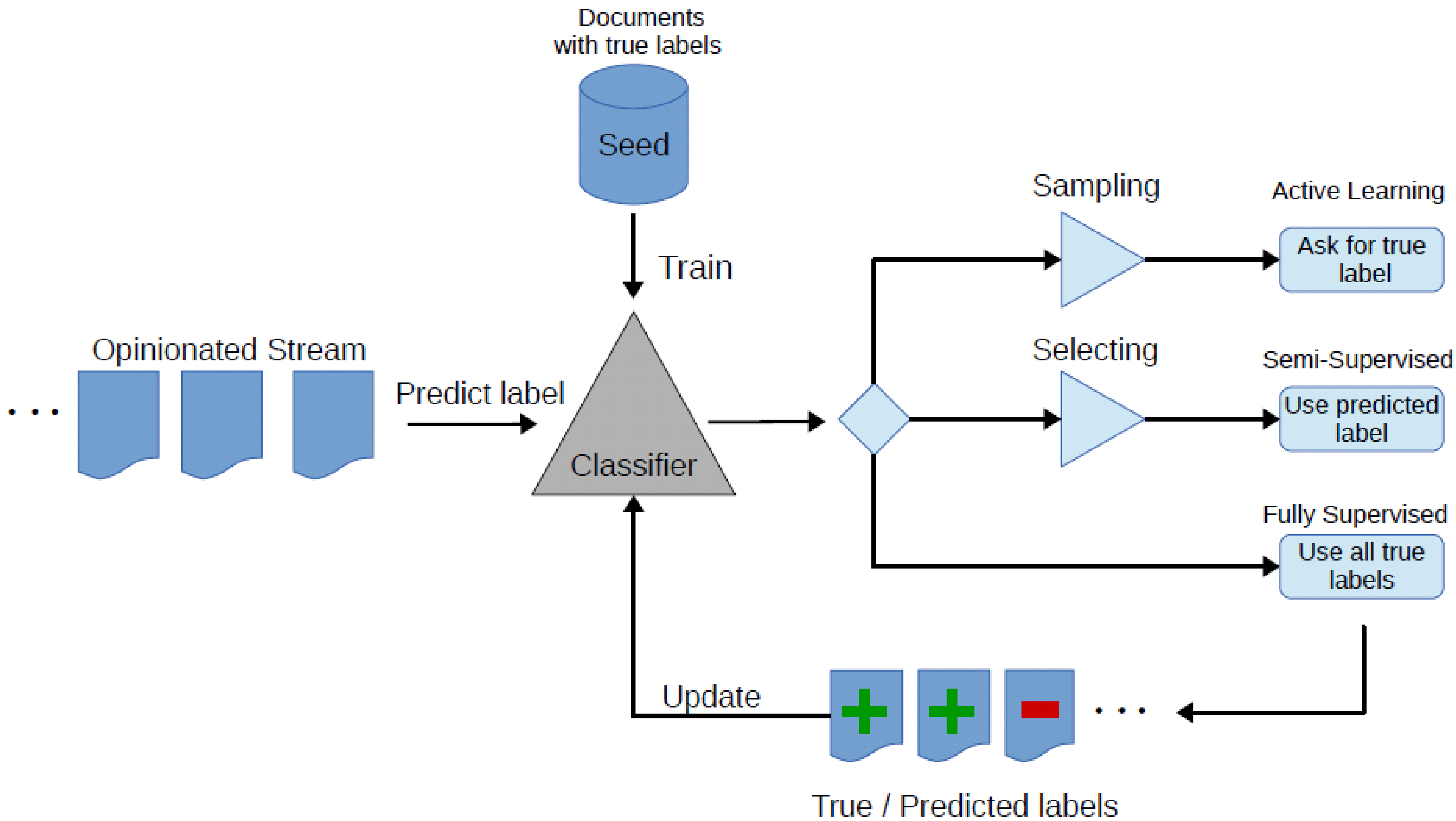
- Exploit the content similarity
  - Not everyone uses emoticons
  - If tweets are similar, “inherit” the sentiment from the “neighboring” tweets

Similar to HSPAM paper

- Exploit the hashtags
  - Start with a seed of positive, negative hashtags

Challenges & Opportunities: Interplay between data and models

- 3 ways of learning: fully-supervised, semi-supervised, active-learning



Challenges & Opportunities: Models

- There are several classification models for batch learning
- Some of the them have been adapted to stream learning
- New methods arise → deep learning
- Results on TwitterSentiment dataset from Kalchbrenner et al, ACL'14

Classifier	Accuracy (%)
SVM	81.6
BINB	82.7
MAXENT	83.0
MAX-TDNN	78.8
NBOW	80.9
DCNN	**87.4**

Table 3: Accuracy on the Twitter sentiment dataset. The three non-neural classifiers are based on unigram and bigram features; the results are reported from (Go et al., 2009).

- Deep learning on streams

Thank you!

Questions?

[1] Sinelnikova et al, Sentiment Analysis in the Twitter stream GfKl'12, based on BA of A. Sinelnikova, LMU 2012.

[2] Wagner et al, Ageing-based Multinomial Naive Bayes Classifiers over Opinionated Data Streams, ECMLPKDD 2015. Based on BA of S. Wagner, LMU 2015.

[3] Spiliopoulou et al, Opinion Stream Mining, Encyclopedia of Machine Learning and Data Mining, Springer 2016.

[4] Informed adaptation, work in progress with V. Iosifidis

[5] Sentiment annotation, , work in progress with V. Iosifidis