Real-Time Associative Classification Algorithms for High-Dimensional Streaming Data

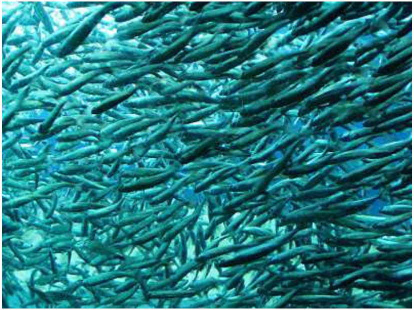
Adriano Veloso

DCC-UFMG

High-Dimensional Streaming Data

We are experiencing a revolution in the capacity to quickly collect and transport large amounts of data.

- Data is produced and collected continuously
- Data complexity and dimensionality is increasing



Learning from High-Dimensional Streaming Data

We want to grab structures, patterns and rules from high-dimensional, rapid data streams.

Learning from High-Dimensional Streaming Data

Challenges for current machine learning algorithms.

- Algorithms must operate with limited resources
- Algorithms must produce models on real-time
- Algorithms must cope with changes in data distribution

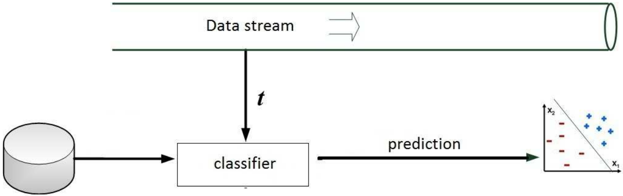
Learning from High-Dimensional Streaming Data

Challenges for current machine learning algorithms.

- Algorithms must operate with limited resources
- Algorithms must produce models on real-time
- Algorithms must cope with changes in data distribution

Alternate approach: Demand-Driven Associative Classification

Classification in High-Dimensional Streaming Data



Associative Classification

Classifier is composed of a set of rules $\mathcal{X} \rightarrow c$.

- $\mathcal{X}$ is a feature-set
- c is the class variable

Rules are extracted from training (labeled) examples.

- Classifier is used in the test set
- It outputs $\hat{p}(c|t)\ \forall\ t$: the likelihood of c being the label for instance t

Advantage: Fast and smart algorithms for rule extraction.

- Effectiveness is competitive

Associative Classification

Classifier is composed of a set of rules $\mathcal{X} \rightarrow c$.

- $\mathcal{X}$ is a feature-set
- c is the class variable

Rules are extracted from training (labeled) examples.

- Classifier is used in the test set
- It outputs $\hat{p}(c|t)\ \forall\ t$: the likelihood of c being the label for instance t

Advantage: Fast and smart algorithms for rule extraction.

- Effectiveness is competitive

Big problem: Exponential dependence on data dimension.

Demand-Driven Associative Classification

Wait for a test instance (t) to come.

- Extract only rules $\mathcal{X} \rightarrow c$ matching t (i.e., $\mathcal{X} \subseteq t$)

Number of rules grows polynomially with data dimension (n).

- But it still grows exponentially with the size of t
  - Fortunatelly, $O(2^{|t|}) \ll O(n^{|t|})$

Contiguous Matching

Test instance t may be large (i.e., text).

- A pattern $\mathcal{X} = \{x_1, x_2, \ldots, x_k\}$ is contiguous if $\forall$ pair (x_i, x_{i+1}), x_i is adjacent to x_{i+1} in t.

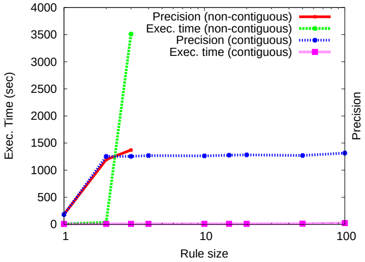
The number of rules grows quadratically with the size of t (i.e., enumeration via sliding window).

- It grows linearly if we bound the cardinality of the rules

Protein Folding (PDB)

Protein structure is related to its function.

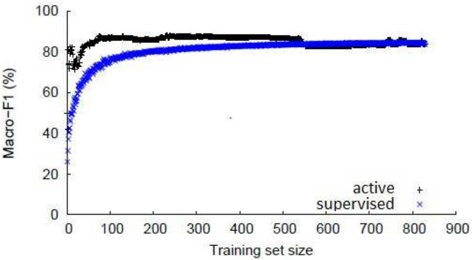
- Predicts structure based on the amino acid sequence



Detecting Polluted Web Content (Youtube)

Hard to obtain examples of spammers and promoters: Active Learning.

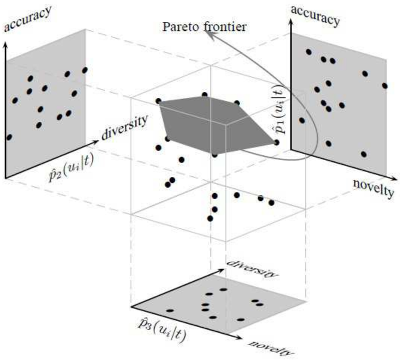
- Select examples that demand the least number of rules
- Select most hard examples



Multi-Objective Recommender Systems (Movielens)

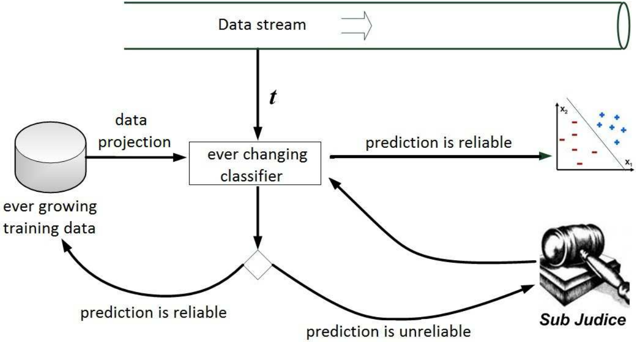
Selective Sampling and Aggregation.

- Select examples with most divesified, accurate or novel items.
- Aggregate results based on Pareto optimality



Sentiment Analysis

Self-Training and Concept Drift



- The only and all the rules that must be updated due to the inclusion of a labeled example $< t, c >$ are those matching t.

Sentiment Analysis

Keeping the classifier always up-to-date as data is updated is challenging.

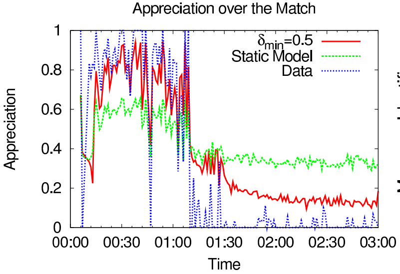
- The only and all the rules that must be updated due to the inclusion of a labeled example $< t, c >$ are those matching t.

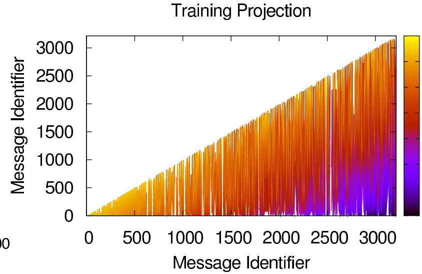
The classifier is totally incremental.

- Highly practical

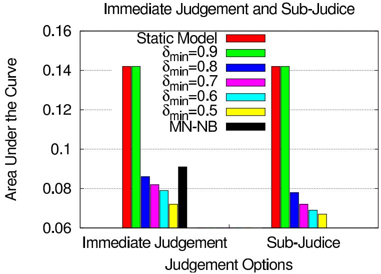
Sentiment Analysis (Twitter) - SIGIR 11

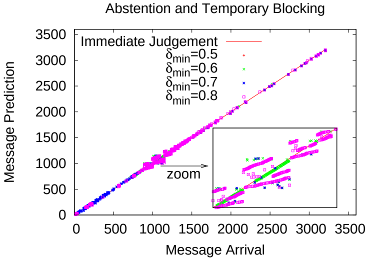
Sentiment Analysis (Twitter) - SIGIR 11



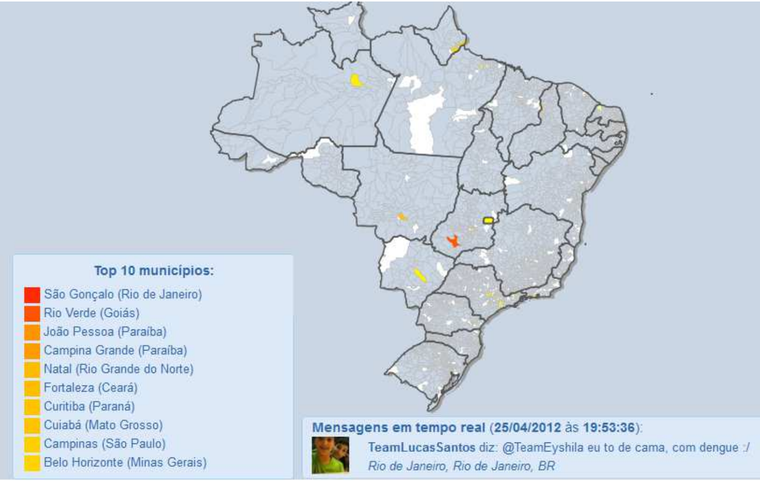


Sentiment Analysis (Twitter) - SIGIR 11





Sentiment Analysis (Twitter) - WEBSCI 11



Named Entity Disambiguation (Twitter) - ACL 12

Given a stream of messages and a list of names $n_1, n_2, \ldots, n_k$ used for mentioning a specific entity e.

- We must monitor the stream and predict whether an incoming message containing n_i indeed refers to e (positive example) or not (negative example).

Named Entity Disambiguation (Twitter) - ACL 12

Expectation-Maximization: positive examples plus unlabeled data.

- Initially, unlabeled examples are treated as negative ones. The process iterates changing labels (i.e., $x^{\ominus \to \oplus}$) until convergence.
  - Label changing operation for instance t is triggered if $\hat{p}(\oplus, t)$ is greater than a threshold. Each instance may have a different threshold.

Operation $x^{\ominus \to \oplus}$ changes the data.

- All rules that must be updated due to operation $x^{\ominus \to \oplus}$ are those matching instance x

The classifier is totally incremental.

Named Entity Disambiguation (Twitter) - ACL 12

	assoc.	SVM	B-SVM
ST_1	$\mathbf{0.67} \pm 0.02$	0.59 ± 0.03	0.61 ± 0.03
ST_2	$\mathbf{0.59} \pm 0.01$	0.54 ± 0.01	0.57 ± 0.01
ST_3	$\mathbf{0.69} \pm 0.01$	0.61 ± 0.03	0.64 ± 0.03
ST_4	$\mathbf{0.59} \pm 0.01$	0.50 ± 0.04	0.55 ± 0.02
ST_5	$\mathbf{0.77} \pm 0.01$	0.67 ± 0.02	0.72 ± 0.03
ST_6	$\mathbf{0.72} \pm 0.01$	0.63 ± 0.01	0.68 ± 0.02

Thank You!

adrianov@dcc.ufmg.br