SpamStop

Intelligent Neural Networks against Spam

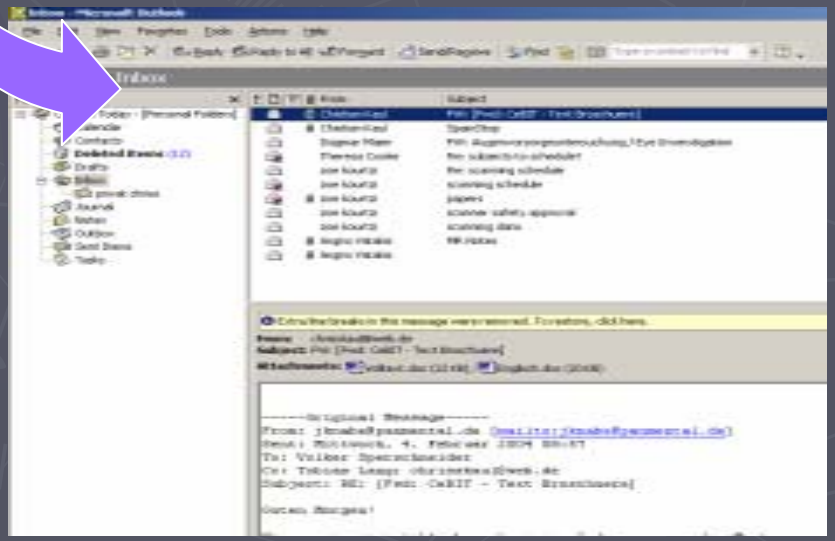
What do we want ?

Sort e-mails in two groups

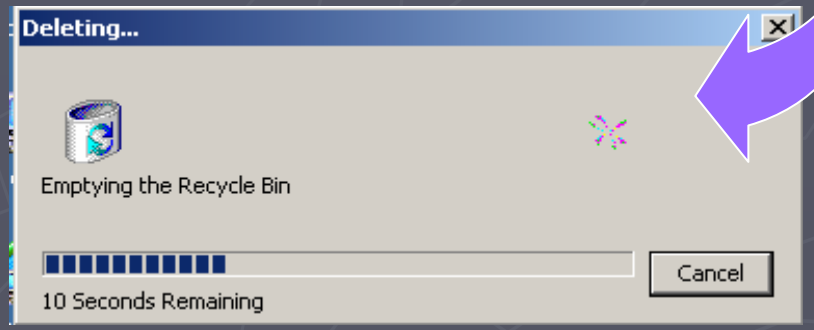
Good, wanted e-mails

Bad e-mails, ads SPAM

to your e-mail program:



deletion / to specified folder



How do YOU classify ?

- Individual classification:
  - Someone might think of ads as interesting e-mails while somebody else might classify them as spams.
- Spam filtering has do be an individual process.

Individual Classification

- Our goal is to train a Support Vector Machine with your personal e-mails.

- SVM = mathematical classification technique
  - Learning from examples
  - SVM-advantages:
    - Efficient learning algorithm
    - Handles arbitrary complex classifications
    - Flexible, no overfitting

SVM: Mathematical Background

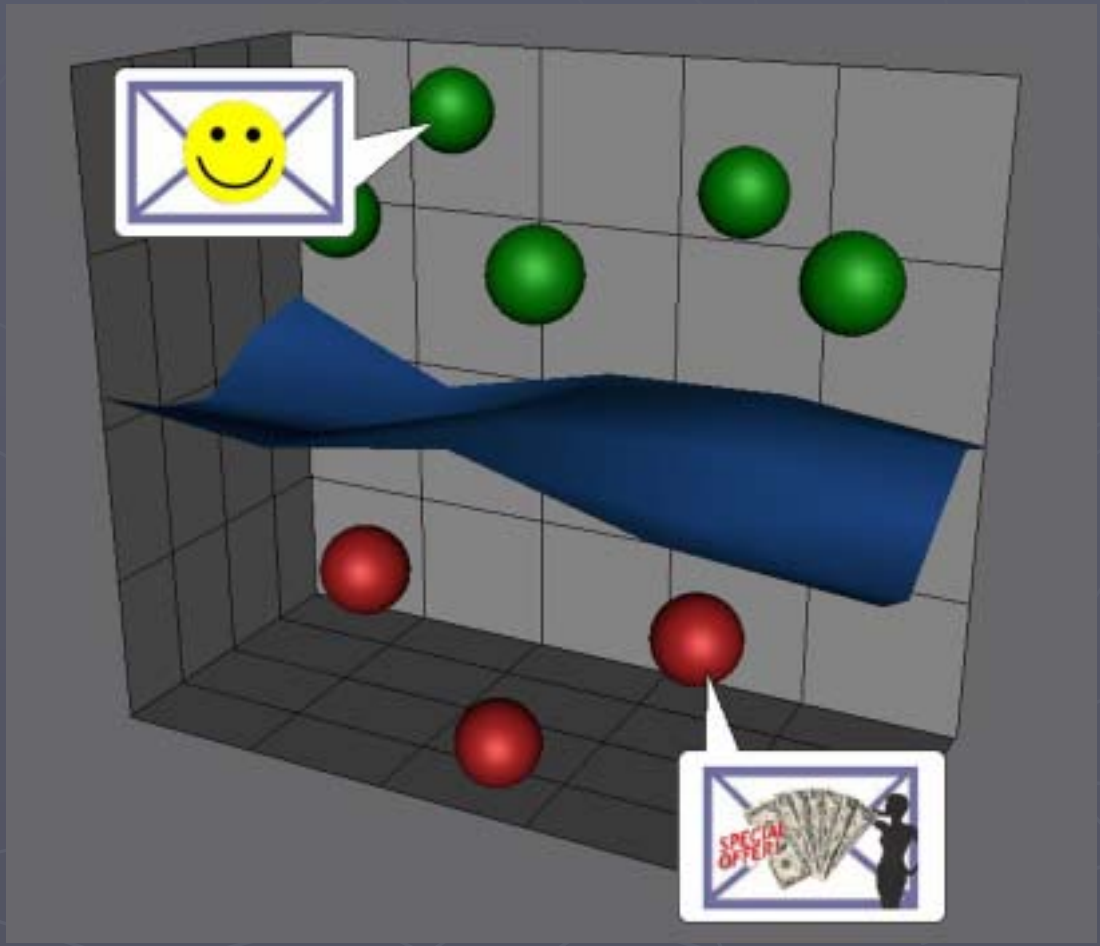
- SVM = Kernel machine
  - Maps data into high-dimensional space
  - There, examples become linearly separable
- Optimization techniques construct optimal separator
  - As far away from the example points as possible
  - No overfitting

… Mathematical Background (II)

- Support vectors = closest points to separator, “hold up” the separating “plane”
- Kernel functions imitate calculating in the high-dimensional space
  - Classifying becomes straight-forward and efficient

SVM for SpamStop

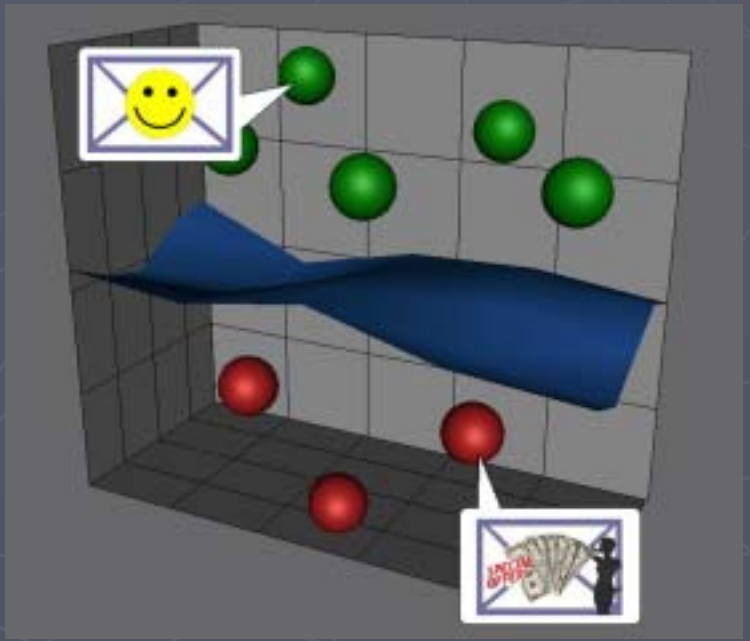
- Creates a separator between wanted e-mail and spams



Optimal performance

- We compare different Kernel functions to ensure **optimal performance for each user**.

- We use methods of an open source Java-package for Kernel machines.
  - Chih-Chung Chang and Chih-Jen Lin, National Taiwan University

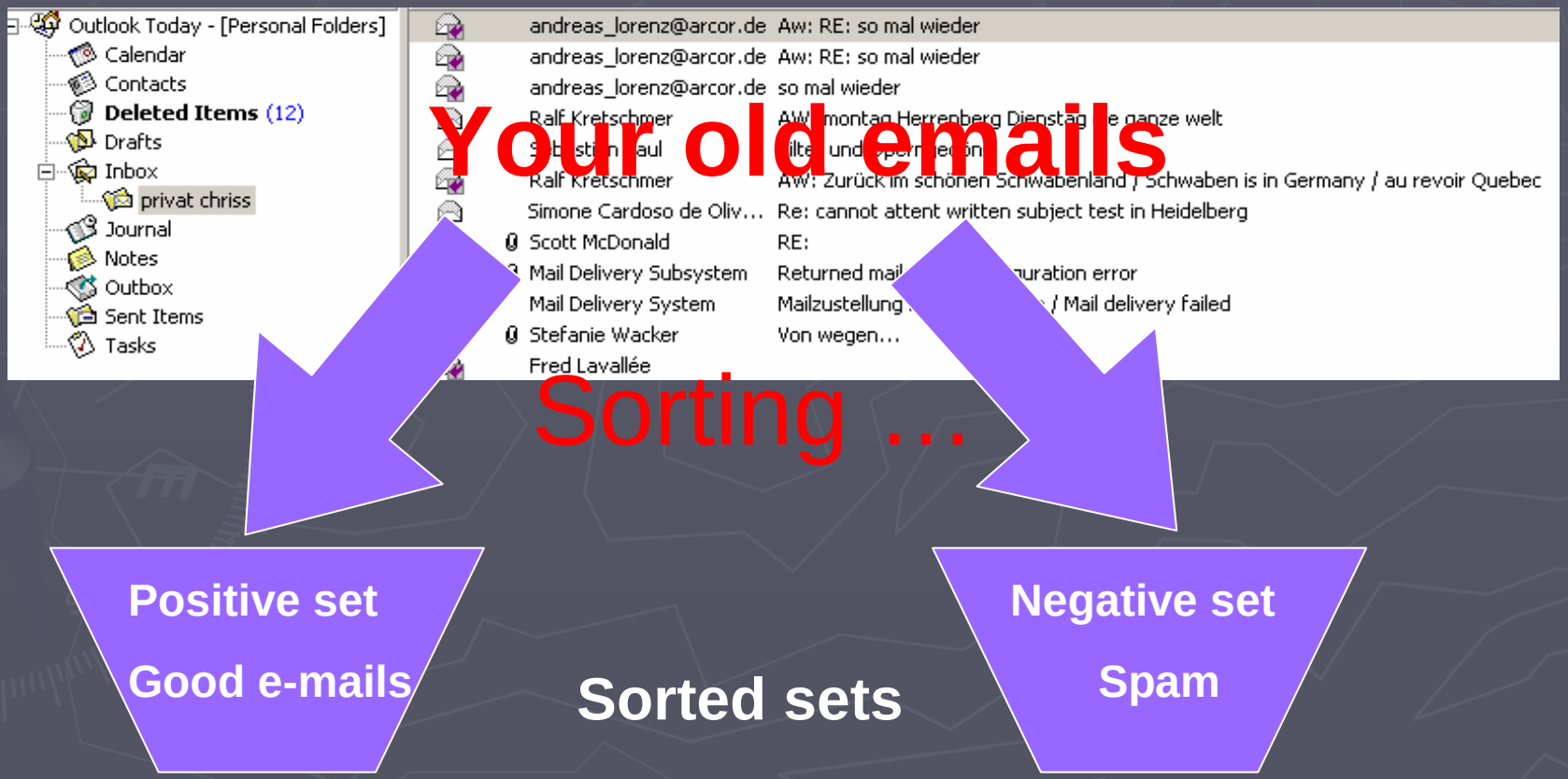


Training **your** SpamStop

- SpamStop uses an SVM to sort your e-mail.
- Therefore, it needs sets of example e-mails to learn your individual classification.
  - Which e-mails you want
  - Which e-mails you consider as spam

Individual Classification

- These sets have to be formed out of your old e-mails



From Email to Vector

- Once sorted, the e-mails need to be transformed into a vector readable for our support vector machine.

From E-Mail to Vector

- Transformation is based on:
  - Words in the e-mail sets
  - Selected header information
  - Sender identity
- Resulting vector represents key information about the transformed e-mail.

...keep in mind that these **key infos** are different for every new user.

...but why is this better ?

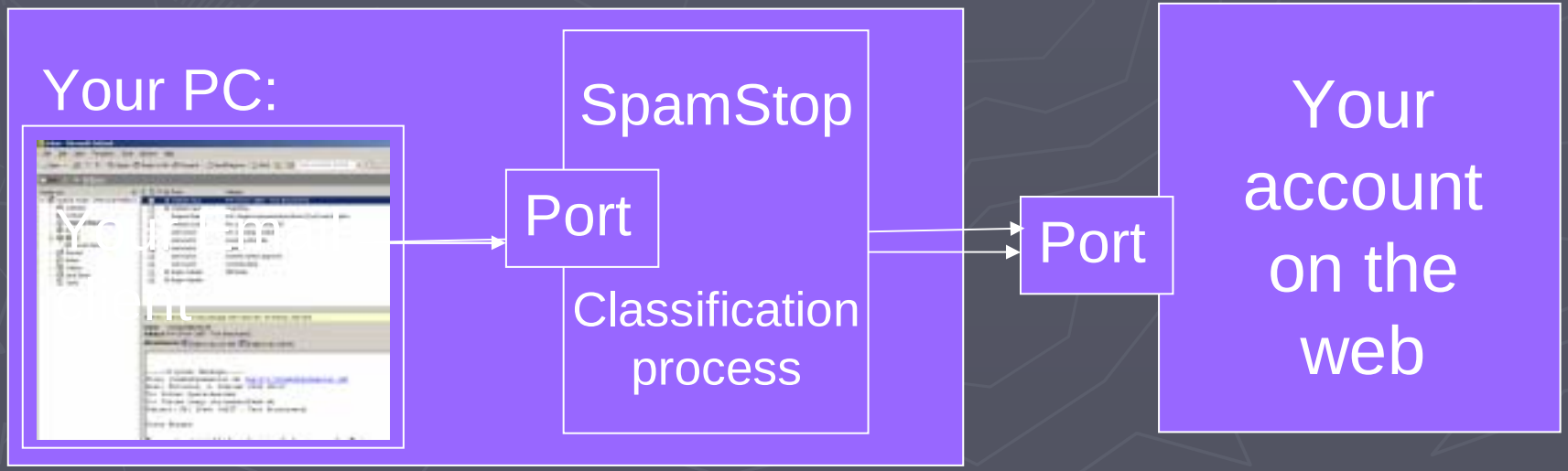
- Unlike the current standard software, we are able to extract key information from your own e-mail.
- Example:
  - A common trick to undergo current Spamfilter is to use transformations of a word:
    - Sex, S_E_X, sE-x, XXX, hardcore, HaRdCoRe, ...
- ...but: semantically these e-mails are still close together and very much different from your normal, good e-mails.
- Therefore: detected by Spamstop

... and further advantages

- Maximal flexibility
  - SpamStop can easily adapt to new ways of classification.
  - Only new examples need to be provided.
- No maintenance
  - No single line of code needs to be changed for new classifications.

How does Spamstop work ?

- By simulating a POP3-server we collect and sort the e-mail before reaching your program:
- Normal design: Using a Proxy:



Demo

1) Training SpamStop:

- Sort your e-mail into two sets
  - Negative set: spams
  - Positive set: your e-mail
- Train the program with your data

2) Let SpamStop collect your correspondence

Contact Info

Program developers:
Christian Kaul Johannes Knabe Tobias Lang

Supervisor:
Prof. Dr. Volker Sperschneider

Internet:
www.spamstop.panmental.de