technology
from seed

The L2F Strategy for Sentiment Analysis and Topic Classification

Fernando Batista and Ricardo Ribeiro

Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento em Lisboa

L² F - Spoken Language Systems Laboratory

Outline

- Data
- Approach
- Experiments
    - Features
    - Submitted runs
- Conclusions and future work
- TASS discussion

Data

Labeled data	Training	Development
7200 tweets	80% (5755 tweets)	20% (1444 tweets)

- We have used a XML file with information about the users that authored at least one of the tweets in the data. Includes information concerning the *user type, which assumes* three possible values:
  - periodista (journalist), famoso (famous person), and politico (politician)
- Sentiment Lexicons in Spanish (Perez-Rosas et al., 2012).
  - Only the most robust part was used: *fullStrengthLexicon,* containing 1346 words, automatically labelled with sentiment polarity

Approach

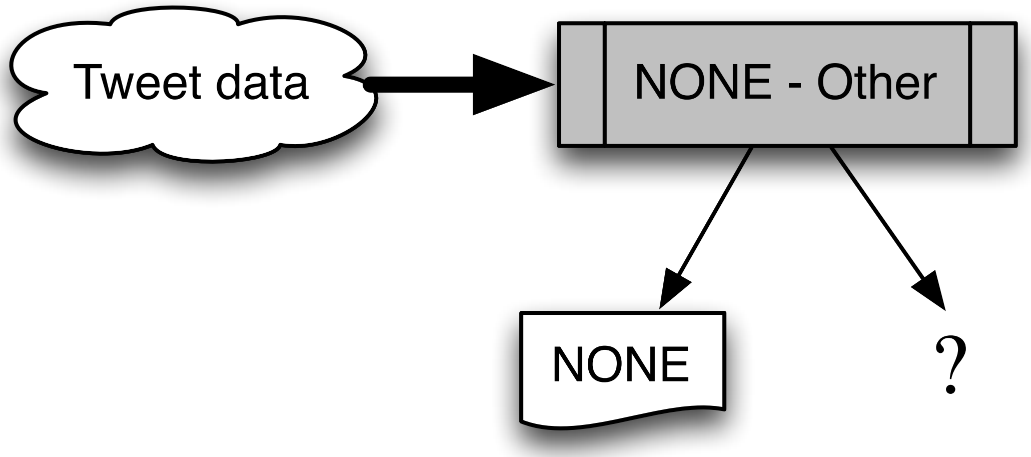
- **Both tasks considered as classification tasks**
  - share the same method: Maximum Entropy Models
- **Most successful recent experiments**
  - binary classification problems (discriminate between two classes)
- **Maximum Entropy models**
  - clean way of expressing and combining different information properties
  - probabilistic classifications, a generalisation of Boolean classification, which provides probability distributions over the classes
  - The ME models used in this study were trained using the MegaM tool (Daume, 2004), which uses an efficient implementation of conjugate gradient (for binary problems).

Sentiment Analysis

- **6 possible classes:**
  - N, N+ negative polarity;
  - P, P+ positive polarity;
  - NEU contains both positive and negative sentiments;
  - NONE without polarity information.
  - The plus sign (+) signals the sentiment intensity.

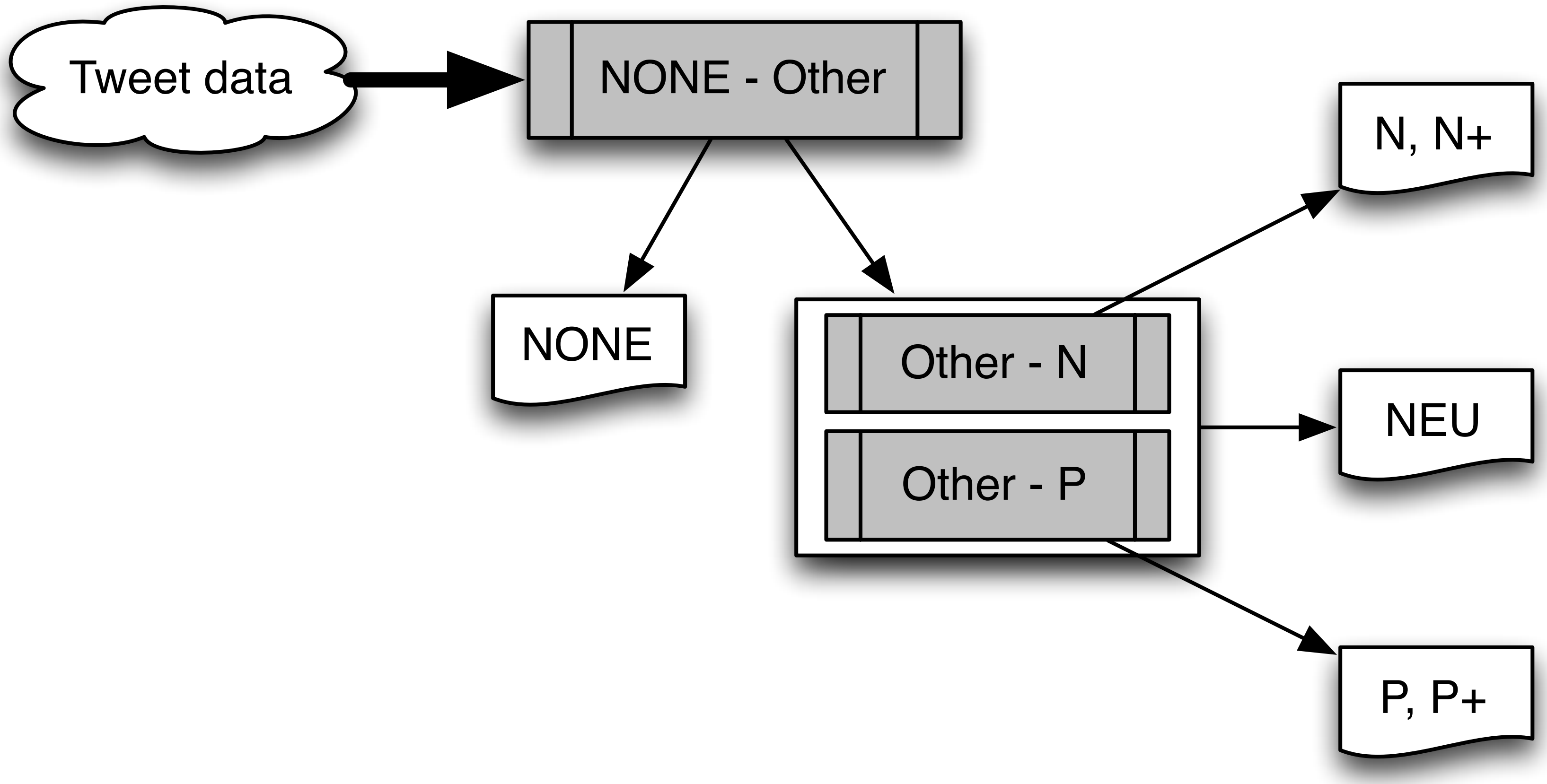
Sentiment Analysis Strategy

- **The first interesting results were achieved by combining 5 different binary classifiers, one for each class.**
  - <NONE, other> was used to discriminate between NONE and any other class.
  - <other, N>, and <other, P> allow to detect negative and positive sentiments, respectively.
  - <N, N+> and <P, P+>, allow perceiving the sentiment intensity



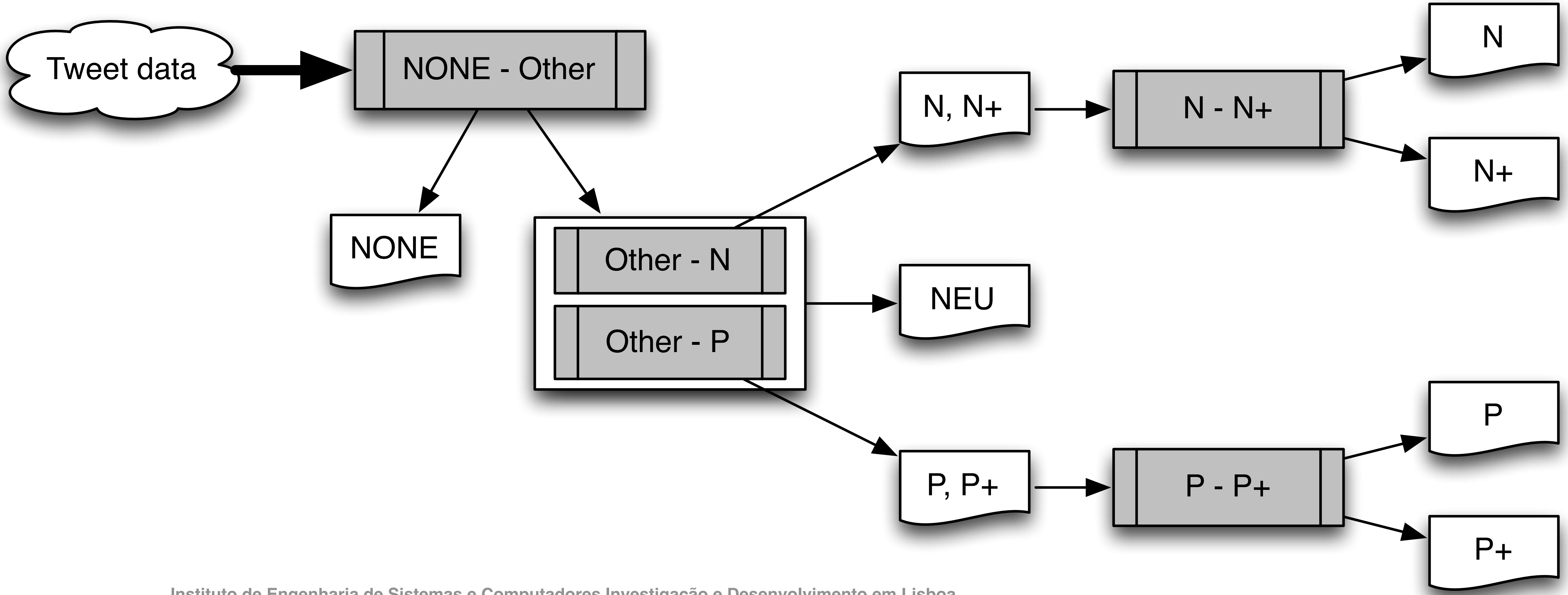
Sentiment Analysis Strategy

- **The first interesting results were achieved by combining 5 different binary classifiers, one for each class.**
  - <NONE, other> was used to discriminate between NONE and any other class.
  - <other, N>, and <other, P> allow to detect negative and positive sentiments, respectively.
  - <N, N+> and <P, P+>, allow perceiving the sentiment intensity

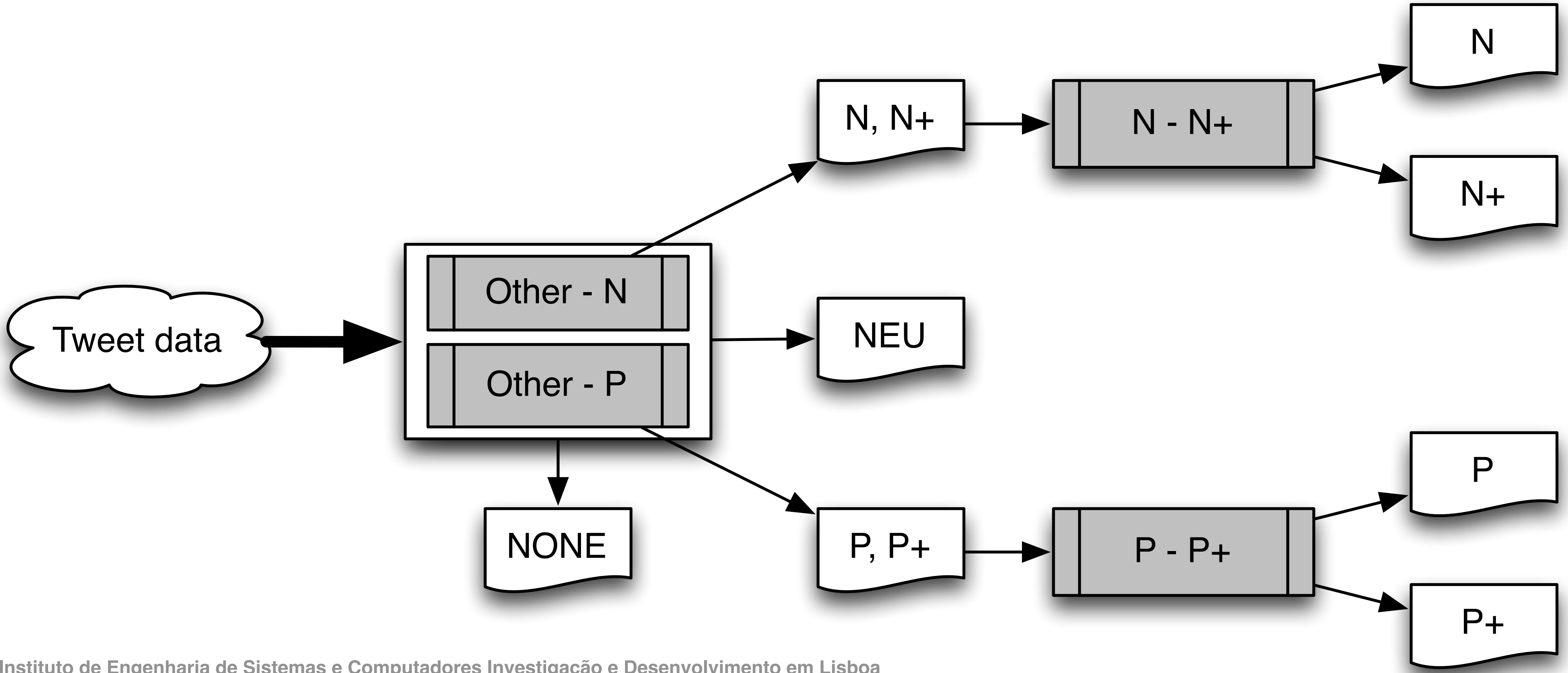


Sentiment Analysis Strategy

- **The first interesting results were achieved by combining 5 different binary classifiers, one for each class.**
  - <NONE, other> was used to discriminate between NONE and any other class.
  - <other, N>, and <other, P> allow to detect negative and positive sentiments, respectively.
  - <N, N+> and <P, P+>, allow perceiving the sentiment intensity

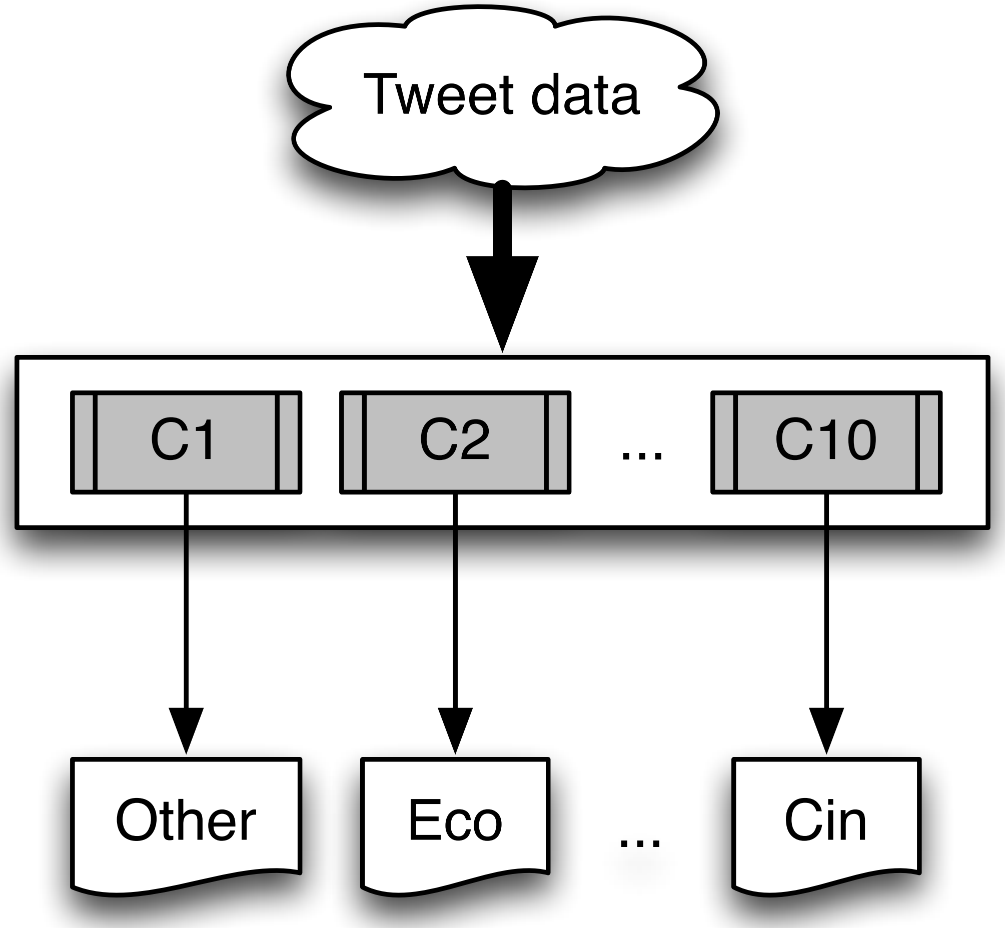


Sentiment Analysis
Adopted Strategy



Topic Detection

- **10 distinct binary classifiers, each one for a different topic.**
  - Each classifier selects its corresponding topic, and in the case no topic was selected, the most probable topic is then selected based on the available classification probabilities



Experiments

- **Tweet content pre-processing**
  - The content of each tweet was firstly tokenized using twokenize (https://bitbucket.org/jasonbaldridge/twokenize/src), a tokenization tool for English tweets, with some minor modifications for dealing with Spanish instead of English
- **Features**
  - Most of the features were used both for sentiment analysis and for topic detection, with small differences, specially concerning the use of punctuation marks

Features

- **The following features, concerning the tweet text, were used for each tweet:**
  - Punctuation marks: used as feature for the sentiment task, but not for topic detection.
  - All words after the words "nunca" (never) or "no" (no) prefixed by "NO_" until reaching some punctuation mark or until reaching the end of the tweet.
  - Each token starting with "http:" was converted into the token "HTTP".
  - All tokens starting with "#" were expanded into two tokens, one with and the other without the "#"
    - A lesser weight was given to the stripped version of the token.
  - All tokens starting with "@" were used, but the token "@USER" was introduced as well, with a smaller weight.
  - All words with more than 3 repeating letters were also used. However, whenever they occur, two more features are produced: "LONG_WORD" with a lower weight, and the corresponding word without repetitions with a high weight (3.0).
  - All cased words were used, but the corresponding lowercase words were used as well. Uppercase words were assigned also to a higher weight, since they are often used for emphasis.
- **Apart from the features extracted from the text, two more features were used:**
  - Username of the author of the tweet.
  - Usertype, corresponding to the user classification, according to the file users-info.xml.
- **Some experiments use feature bigrams involving the following tokens**
  - HTTP, words starting with # without the diacritic #, @USER, LONG_WORD, all other words converted to lowercase.

Experiments

- Sentiment analysis
  - Baseline: 52.5 Accuracy (Acc) in the development set
  - plus tweet's author name: 53.6 Acc (+1.1)
  - plus user type: 54.2 (+0.6)
  - Best results achieved by using punctuation marks: 55.1 Acc (+0.9)
- Topic detection
  - Differences across experiments were subtle, because improvements in one classifier may worsen results in another classifier
  - Adding the author's name produced slightly better results
  - Contrarily to what was expected, providing the user type as a feature did not improve results
  - Adding punctuation marks decreased the overall performance

	Sentiment analysis	Topic detection
Unigrams only	63.4 (55.2)	64.9 (43.2)
Unigrams, Bigrams	62.2 (53.8)	65.4 (42.5)
Sentiment lexicon	63.2 (54.8)	

Future

- **Improve tokenization and normalisation**
  - Named entity detection
  - Specific writing styles
- **Use the remainder information available**
  - For example, the use of the sentiment polarity type (AGREEMENT, DISAGREEMENT), together with other information about the user (e.g. number of tweets, number of followers, number of following), would probably have an impact on the results.

Thank you

Obrigado

L^2 F - Spoken Language Systems Laboratory