ELAC: Ensemble Learning for Anaphora- and Coreference-Resolution-Systems

http://dakhma.net/elac

Thomas Bögel, Lukas Funk, Andreas Kull
Softwareprojekt SS10

Motivation & Architecture

Motivation

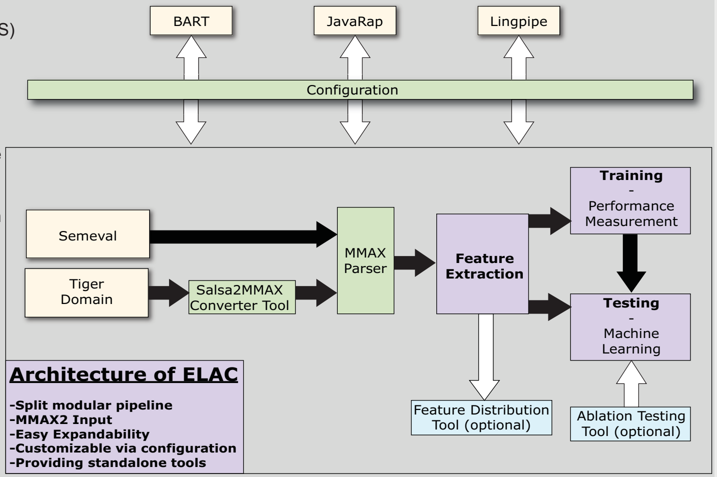
Anaphora- and Coreference-Resolution-Systems (ACRS) are typically very specialized on a single phenomenon and failing on others, especially when it comes to domain adaptation.

Therefore we created a system, built around the machine learning tool WEKA, to combine several single ACR-Systems.

The goal was to test the ensemble learning approach in order to obtain improvements in comparison to a single system and on domain adaptation.

Our ressources:
– Training & Testing: Semeval2010 Task2
– Domain Adaptation: Sherlock Holmes
– MUC6 evaluation standard
– MMAX2 format

Our guidelines:
– Modularity
– Expandability
– Usability


BART
JavaRap
Lingpipe
Configuration
Semeval
Tiger Domain
Salsa2MMAX Converter Tool
MMAX Parser
Feature Extraction
Training - Performance Measurement
Testing - Machine Learning
Feature Distribution Tool (optional)
Ablation Testing Tool (optional)

Architecture of ELAC

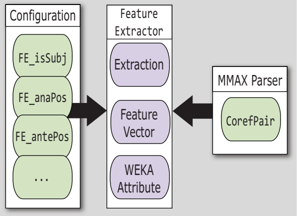
-Split modular pipeline
-MMAX2 Input
-Easy Expandability
-Customizable via configuration
-Providing standalone tools

Training: Extraction & Measurement

Feature Extractor

Extraction: The Coreference pairs and their features are extracted from the corpus.

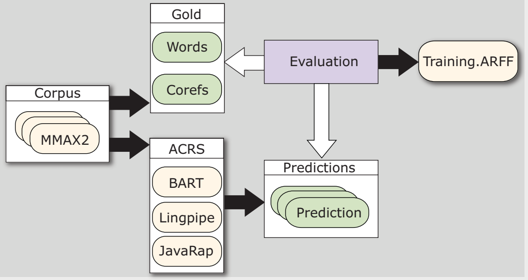
Modularity: New features can be added easily.



Feature Dependent Performance Measurement

Predictions of coreferences, created by the ACR-Systems, are checked against the gold standard.

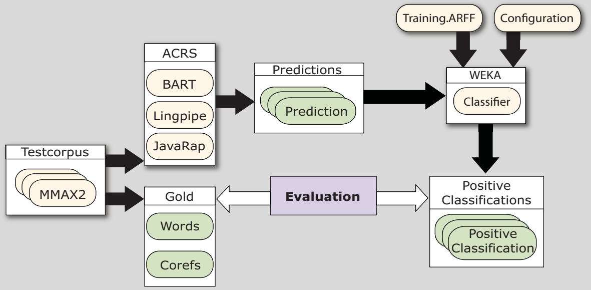
An **ARFF file for WEKA**, containing the positive and negative results of the evaluation, is created.



Testing: Learning & Evaluation

Machine Learning

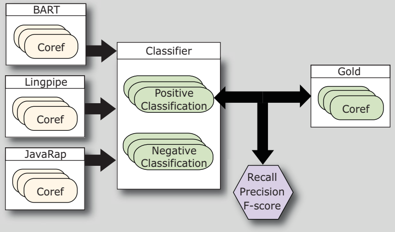
WEKA: Classifier(s) are trained with an ARFF file
ACR-Systems: Predict anaphora and coreferences in test data
Classification: Classifier classifies predictions as trustworthy or untrustworthy



Evaluation

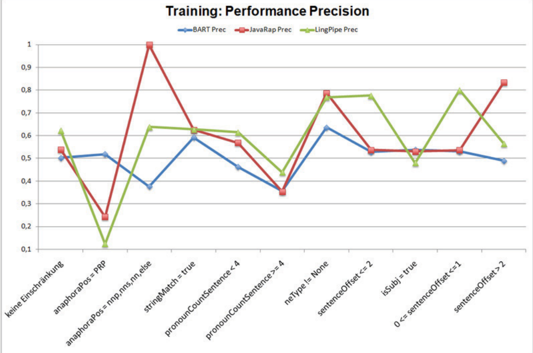
A **Positive list** with trustworthy classifications is created by the classifier.

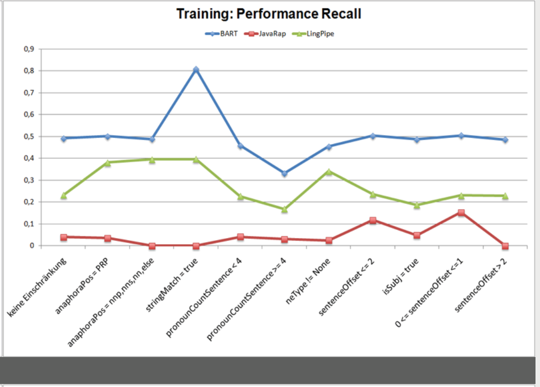
It **is compared to the gold standard** which is extracted from the test data.

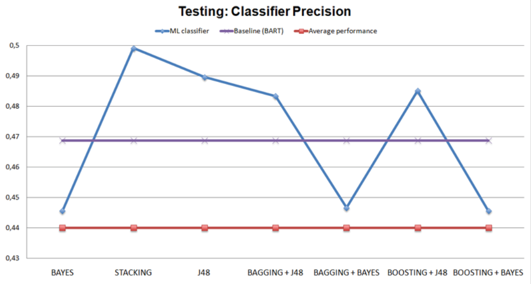


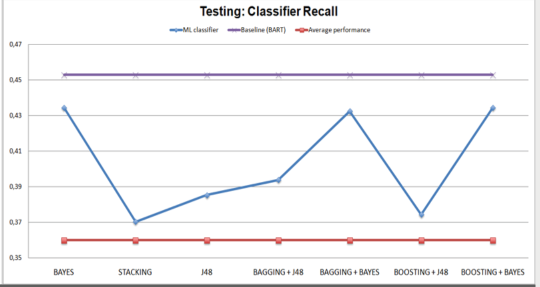
Recall, precision and F-score are computed to evalute the Classifier(s)

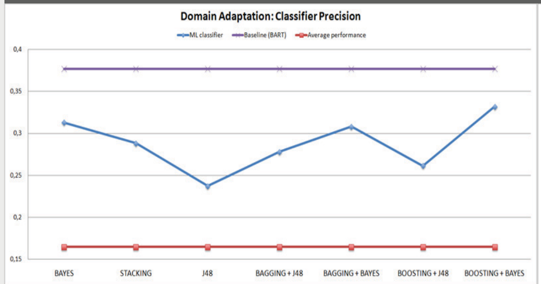
Findings: Data

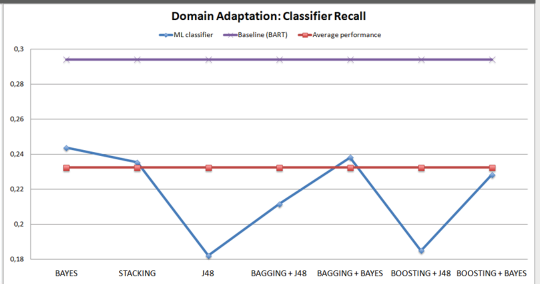












Findings: Conclusion

Findings

Training:
– Performance of individual ACR-Systems with various features
– Based on 50% of Semeval2010 Task2 corpus
– MUC6 evaluation which causes problems for JavaRap
– BART is clearly dominating

Testing:
– Precision improved with an ensemble of classifiers
– Baseline's F-score not reached
– JavaRap made too many bad decisions

Domain Adaptation:
– Sherlock Holmes novel from Tiger/Salsa corpus
– BART has achieved the best F-score
– Baseline not reached
– ACR-Systems provided too much wrong data
– Better precision than the average of all systems

Conclusion

– Ensemble Learning better than the average of all ACR-Systems
– BART is an overall robust system (only if trained properly)
– Best results probably with completely complementary systems

Department of Computational Linguistics
http://www.cl.uni-heidelberg.de
University of Heidelberg
boegel@cl.uni-heidelberg.de funk@cl.uni-heidelberg.de kull@cl.uni-heidelberg.de