Automatic Tagging of Stack Overflow Questions

Authors: Glatzhofer Michael, Schiffer Verena

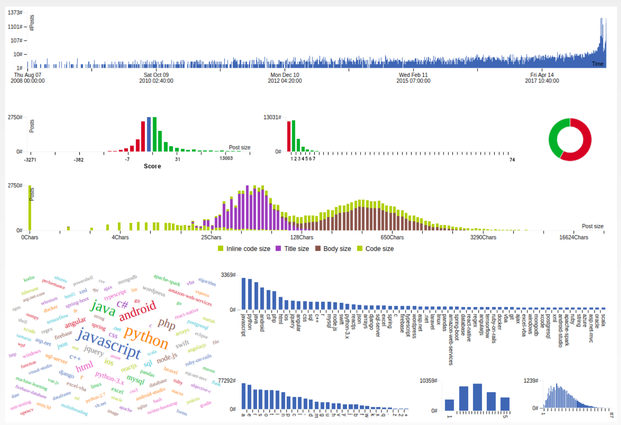
Overview

Stack Overflow is a Q&A site for programmers. Questions can have up to five tags representing technologies or programming languages. The goal of this project is to compare classification and clustering methods applied for automatic question tagging. The following analysis only handles single-label classification to achieve a proper comparision with clustering. Natural language processing is based on a tf-idf vector in various configurations, which form a large search space with multiple decomposition, feature selection, clustering and classification algorithms. In this search space supervised classification algorithms achieve higher scores.

Data Set

Stack Overflow provides a public API to access questions and answers, but for this project only questions are considered. Questions are provided by Stack Overflow as HTML and therefore meta-information about the content-type can be extracted. Title, body, code and inline-code are considered in various compositions. To avoid problems with dynamic HTML and supply a convenient parsing module, JSDOM has proven to be effective. Furthermore the tags of each question are extracted.

Data Type	Description	Comment
$tf = raw \vee bin \vee log \vee max$	Term frequency variant	
$df = raw \vee max \vee log$	Document frequency variant	
$vi = nltk \vee sklearn$	Vecimpl	nltk, sklearn
$c = T^3 B^1 C^1 I^2$	Question features	Title, Body, Code, Inlinecode
$tf_{min}, df_{min} \in \mathbb{R}$	filter	nltk, sklearn
$cfg = tf + df + vi + c + tf_{min} + df_{min}$	Configuration	
$s_{cfg} \in \mathbb{N}^+$	Number of Samples	−unused
$f_{cfg} \in \mathbb{N}^+$	Number of terms	$-(tf < 25 \vee df < 5)$
$q_{cfg} \in \mathbb{N}^+$	Number of classes	− unused
$C \in S^c_{cfg}$	Class name (top 200)	s wrt. cfg
$S \in S^s_{cfg}$	Sample Id. (Question Id)	s wrt. cfg
$F \in S^f_{cfg}$	Feature names	term or ngram
$Y \in \mathbb{B}^{cq}_{cfg}$	sample class vector with same mask as X	s and q wrt. cfg
$R \in \mathbb{N}^{sf}_{cfg}$	tf-raw	s and f wrt. cfg
$X \in \mathbb{R}^{sf}_{cfg}$	tf-idf	s and f wrt. cfg



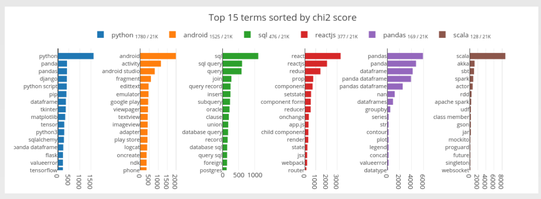
Basic Statistics

As seen on left figure, question timestamps show an exponential distribution. Content-type size differs strongly but are equally weighted in the tf-idf matrix. The tag occurence distribution is exponential. Only the top 200 tags are considered, since rare tags achieve low scores, but consume valuable computational resources.

Classification

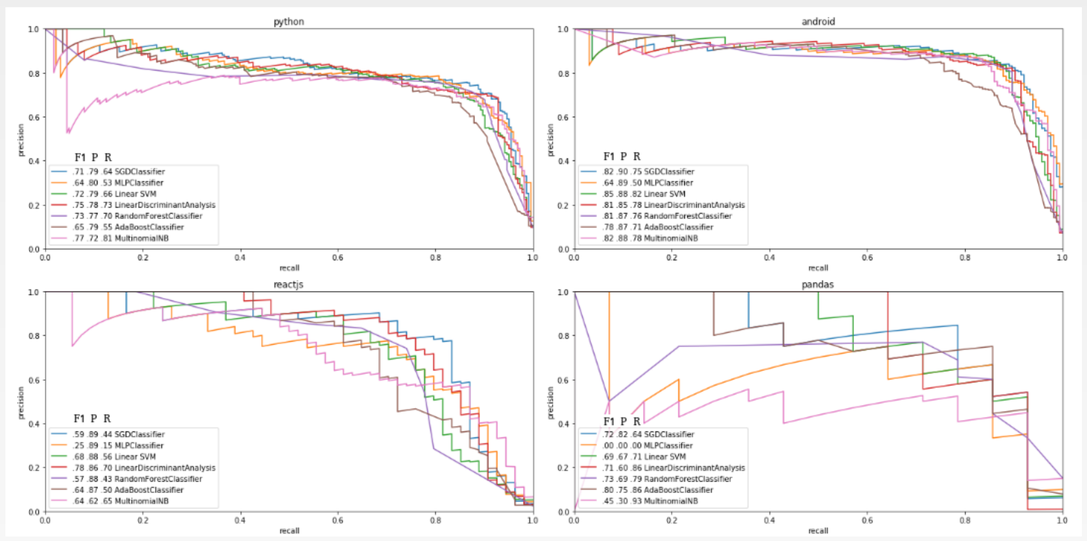
Feature Selection

Chi², ANOVA and mutual information scores are pre-computed to decrease computation time. The right figure shows shows chi² scores for a selected subset of tags. For the computation of mutual information a subset of 5k samples are randomly selected due to performance issues.



Classification Algorithms

Classifiers are trained for top 200 tags, whereby the following alogrihtms show proper results: Stochastic Gradient Descent, Multi Layer Perceptron, Linear Support Vector Machine, Linear Discriminant Analysis, Random Forest Classifier, Adaptive Boosting and Multinomial Naive Bayes.

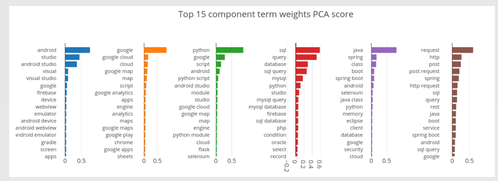


The results shown in the diagram are independent of the feature selection algorithm, as well as the tf-idf configuration, but strongly depend on k, in terms of k-best features, stemming/lemming algorithm and the combination of different content-types. For this dataset it is easy to achieve a high precision, but harder to get good recall scores. Tags occuring in less than 1% of all questions have a significantly lower F1 score.

Clustering

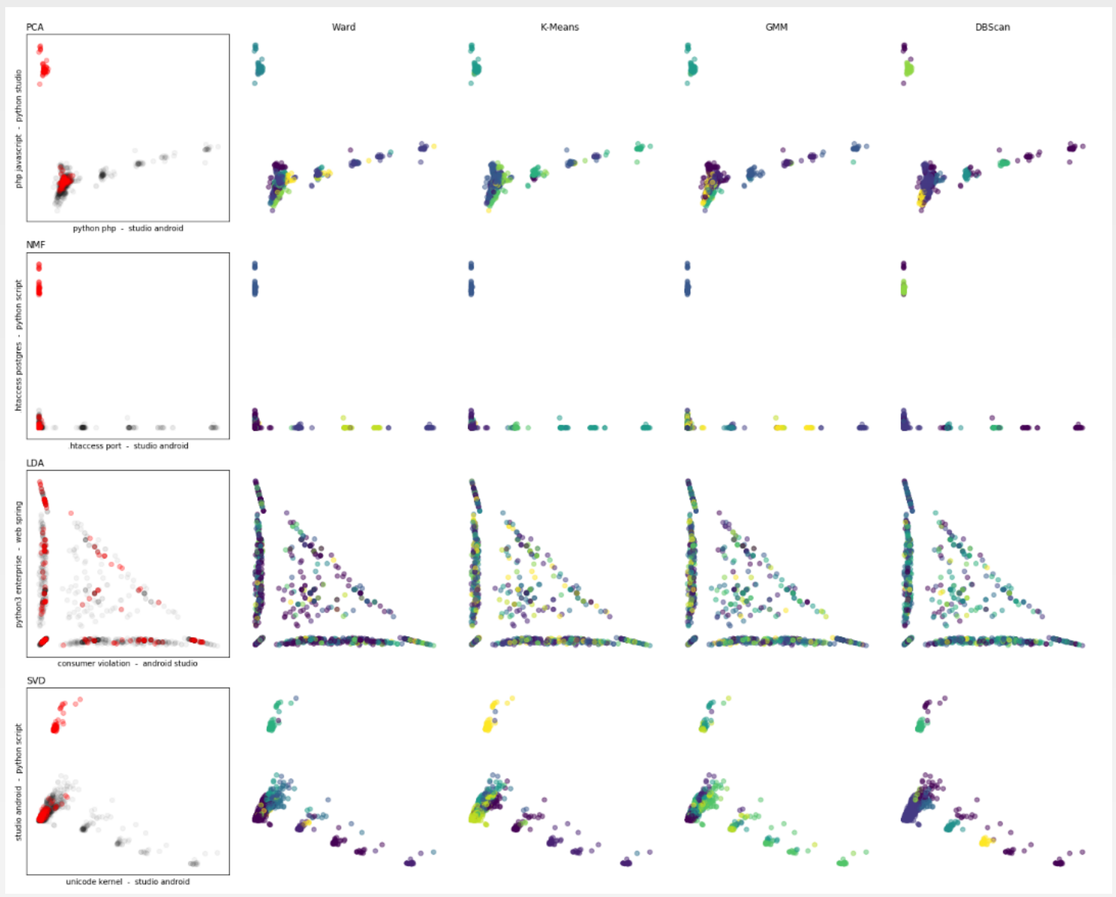
Decompositions

The following decompositions are precomputed for all tf-idf vector configurations with a decompositon dimension range d=[5, 100]: Singular Value Decomposition, Latent Dirichlet Allocation, Principal Component Analsysis, Non-negative Matrix Factorization. Additionally the decopositions with dimension 2 are computed for visualization puproses only.



Clustering Algorithms

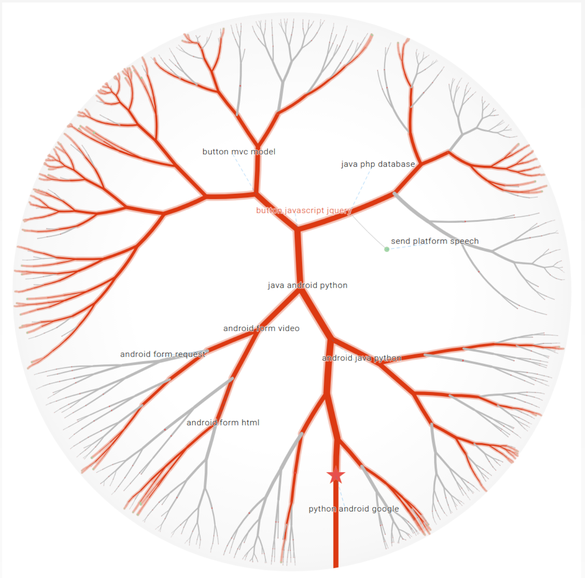
Various clustering algorithms, namely k-means, gaussian mixture model, density-based spatial clustering of applications with noise, agglomerative clustering with ward linkage function are applied to each decomposed matrix with a clustercount range k=[5, 100].
The figure below shows clustering algorithm results with clustercount k=15, decomposition dimension d=75 projected in a 2D space. First column shows ground truth in red. The following columns show clustering results for different algorithms. Each row positions the samples by a two dimensional decomposition, but colors are determined by higher dimension clustering results.



Single clusters may achieve good precision, but rarely an optimal recall. By combining multiple clusters with high precision, F1 score of 0.8 are achievable for some selected tags. The following hierarchical cluster visualisation of the javascript tag, shows the basics of this approach in detail.

The 10 o'clock area, shows the main cluster of javascript tags in context of HTML, but javascript is also used in combination with other technologies, visible in the 1 o'clock area. This project implements a simple algorithm which collects all clusters with precision > p. The union of this clusters must have a precision > p, and the recall must be higher than the recall of its components.

This approach may be further improved. Agglomerative cluster trees can be used to search a better result more efficiently, by following the root-path of high precision nodes until precision drops. For such an algorithm the number of clusters may be unknown, and cluster size distribution is not necessarily uniform in comparison to traditional agglomerative clustering with ward distance function.



References

[1] Scikit-learn: Machine Learning in Python; 2011; Pedregosa et al., JMLR 12, pp. 2825-2830
[2] NLTK: the Natural Language Toolkit; 2002; ETMTNLP '02 Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics - Volume 1

TU Graz – Institute of Interactive Systems and Data Science
8010 Graz, Inffeldgasse 16c, Austria, Tel.: +43 (316) 873-5624
office.isds@tugraz.at, https://www.tugraz.at/institute/isds/home/