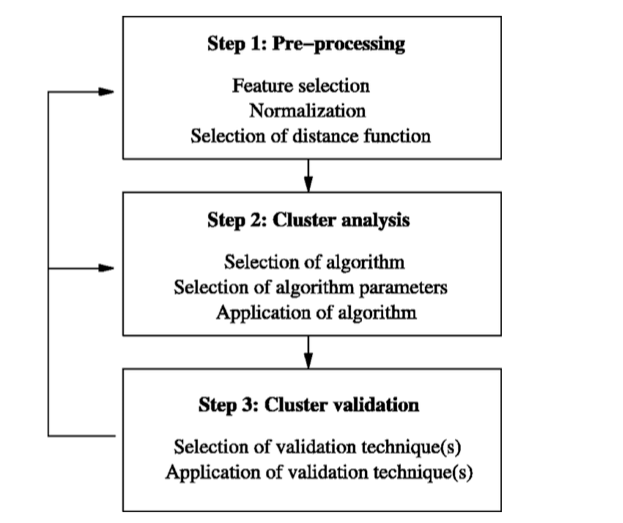
WHAT IS CLUSTERING?

Maybe chapter with overview

A clustering process consists of three steps: pre-processing of the data, cluster analysis, and cluster validation (Handl 2005).



PREPROCESSING

Tokenization with NLTK.

Choosing features

TFIDF

No PCA etc cause to obscure

CLUSTERING

Distance metric: no Euclidean e.g. kmeans

Research in machine learning suggests that Euclidean distances don’t work well with high-dimensional data such as this (e.g. Aggarwal et al who advise against using Euclidean on data with more than 20 features). Instead, the Manhattan distance, the Cosine distance or fractional Ls (Aggarwal) are used.

This already limits our choice of clustering algorithms as for instance the popular k-means algorithm works with Euclidean distances.

Choice of clustering algorithm, settings

CLUSTER VALIDATION

Compare different algos

Clustering vs cluster

The last two steps were automated by a Python module written for this study called Clustertools which is available for inspection here: github.

The Clustertools package, following the outline of a General Purpose Computer-Assisted Clustering Methodology in Grimm and King (2011). It gives several approaches to cluster evaluation.

For each clustering, it establishes basic characteristics and then compares clusterings to each other. More specifically it outputs the size of each cluster in a clustering; if supplied with external labels, it shows the distribution of labels over clusters. For instance, if we input the category labels of the Craigslist data, it shows how many items of each category are contained in the cluster. Likewise, it shows how each category is split up over clusters by computing the percentage of this category’s total contained in a cluster.

Sample output [edited to exclude cluster validation section]

CLUSTERING CALLED <class 'sklearn.cluster.k\_means\_.KMeans'> HAS 4 CLUSTERS

[…]

Cluster 0 contains 15840 items, 25.0 % of the total

1962.0 items of category w4w make up 12.0 % of this cluster

3978.0 items of category w4m make up 25.0 % of this cluster

7030.0 items of category m4w make up 44.0 % of this cluster

2867.0 items of category m4m make up 18.0 % of this cluster

1.0 items of category m4t make up 0.0 % of this cluster

1.0 items of category m4mw make up 0.0 % of this cluster

1.0 items of category t4m make up 0.0 % of this cluster

Cluster 1 contains 21065 items, 33.0 % of the total

3298.0 items of category w4w make up 16.0 % of this cluster

[…]

-----------

Statistics per category

Category w4w has 11851 items

1962 items or 17.0 percent in cluster 0

3298 items or 28.0 percent in cluster 1

3348 items or 28.0 percent in cluster 2

3243 items or 27.0 percent in cluster 3

Category w4m has 14237 items

3978 items or 28.0 percent in cluster 0

4222 items or 30.0 percent in cluster 1

The script then outputs distinctive features: which variables differ the most between two clusters? In the example below, the word *and* is positively correlated with members of cluster 0, the word *sports* with cluster 1. Computed from centroids???

-----------

Strongly predictive features are

Raw Scores

Cluster 0 and cluster 1 are differentiated by

and : 0.0526853074344, the : 0.0526851086915, sports : -0.0482161635876, […]

Cluster 0 and cluster 2 are differentiated by

[…]

The respective centroids are also used to present some “prototypical data” (Grimm) to the user. This represents the most central documents of each cluster. Output edited for brevity.

-----------

Here is a typical document for each cluster

We set the distance metric to euclidean

CLUSTER 0

<file> <title=workout buddy - w4w> <plat=stp> <city=miami> <date=2015-05-05> <time=2:56pm> <clid=5011207259> <link=http://miami.craigslist.org/pbc/stp/5011207259.html> […]

Looking for a workout buddy to go to la fitness with me. Im on the bigger side and if you are too thats even better so we can give each other motivation. </text> </file>

CLUSTER 1

[..]

The cluster validation section of Clustertools computes a cluster-internal (i.e. it does not need external labels of the “correct” cluster) quality metric, the silhouette score (Rousseeuw 1987). It also computes several metrics (v-score, adjusted rand, etc ) that can be used to compare different clusterings. Sample output, edited for readability.

CLUSTERING CALLED <class 'sklearn.cluster.k\_means\_.KMeans'> HAS 4 CLUSTERS

Its silhouette score is 0.25221911471

[…]

-----------

Comparing clusterings

---

Metric: adjustedrand\_sim

KMeans

MiniBatchKMeans 0.504176061542

---

Metric: adjustedmutualinfo\_sim

[…]