Chapter 4

Clustering Algorithms and Evaluations

There is a huge number of clustering algorithms and also numerous possibilities for evaluating a clustering against a gold standard. The choice of a suitable clustering algorithm and of a suitable measure for the evaluation depends on the clustering objects and the clustering task. The clustering objects within this thesis are verbs, and the clustering task is a semantic classification of the verbs. Further cluster parameters are to be explored within the cluster analysis of the verbs.

This chapter provides an overview of clustering algorithms and evaluation methods which are relevant for the natural language clustering task of clustering verbs into semantic classes. Section 4.1 introduces clustering theory and relates the theoretical assumptions to the induction of verb classes. Section 4.2 describes a range of possible evaluation methods and determines relevant measures for a verb classification. The theoretical assumptions in this chapter are the basis for the clustering experiments in the following Chapter 5.

4.1 Clustering Theory

The section starts with an introduction into clustering theory in Section 4.1.1. Section 4.1.2 relates the theoretical definitions of data objects, clustering purpose and object features to verbs as the clustering target within this thesis, and Section 4.1.3 concentrates on the notion of similarity within the clustering of verbs. Finally, Section 4.1.4 defines the clustering algorithms as used in the clustering experiments and refers to related clustering approaches. For more details on clustering theory and other clustering applications than the verb classification, the interested reader is referred to the relevant clustering literature, such as Anderberg (1973); Duda and Hart (1973); Steinhausen and Langer (1977); Jain and Dubes (1988); Kaufman and Rousseeuw (1990); Jain *et al.* (1999); Duda *et al.* (2000).

4.1.1 Introduction

Clustering is a standard procedure in multivariate data analysis. It is designed to explore an inherent natural structure of the data objects, where objects in the same cluster are as similar as possible and objects in different clusters are as dissimilar as possible. The equivalence classes induced by the clusters provide a means for generalising over the data objects and their features. Clustering methods are applied in many domains, such as medical research, psychology, economics and pattern recognition.

Human beings often perform the task of clustering unconsciously; for example when looking at a two-dimensional map one automatically recognises different areas according to how close to each other the places are located, whether places are separated by rivers, lakes or a sea, etc. However, if the description of objects by their features reaches higher dimensions, intuitive judgements are less easy to obtain and justify.

The term *clustering* is often confused with a *classification* or a *discriminant analysis*. But the three kinds of data analyses refer to different ideas and are distinguished as follows: Clustering is (a) different from a classification, because classification assigns objects to already defined classes, whereas for clustering no a priori knowledge about the object classes and their members is provided. And a cluster analysis is (b) different from a discriminant analysis, since discriminant analysis aims to improve an already provided classification by strengthening the class demarcations, whereas the cluster analysis needs to establish the class structure first.

Clustering is an exploratory data analysis. Therefore, the explorer might have no or little information about the parameters of the resulting cluster analysis. In typical uses of clustering the goal is to determine all of the following:

- The number of clusters,
- The absolute and relative positions of the clusters,
- The size of the clusters,
- The shape of the clusters,
- The density of the clusters.

The cluster properties are explored in the process of the cluster analysis, which can be split into the following steps.

1. Definition of objects: Which are the objects for the cluster analysis?
2. Definition of clustering purpose: What is the interest in clustering the objects?
3. Definition of features: Which are the features that describe the objects?
4. Definition of similarity measure: How can the objects be compared?
5. Definition of clustering algorithm: Which algorithm is suitable for clustering the data?
6. Definition of cluster quality: How good is the clustering result? What is the interpretation?

Depending on the research task, some of the steps might be naturally given by the task, others are not known in advance. Typically, the understanding of the analysis develops iteratively with the experiments. The following sections define a cluster analysis with respect to the task of clustering verbs into semantic classes.

4.1.2 Data Objects, Clustering Purpose and Object Features

This work is concerned with inducing a classification of German verbs, i.e. the data objects in the clustering experiments are **German verbs**, and the clustering purpose is to investigate the automatic acquisition of a linguistically appropriate **semantic classification** of the verbs. The degree of appropriateness is defined with respect to the ideas of a verb classification at the syntax-semantic interface in Chapter 2.

Once the clustering target has been selected, the objects need an attribute description as basis for comparison. The properties are grasped by the data features, which describe the objects in as many dimensions as necessary for the object clustering. The choice of features is of extreme importance, since different features might lead to different clustering results. Kaufman and Rousseeuw (1990, page 14) emphasise the importance by stating that ‘a variable not containing any relevant information is worse than useless, because it will make the clustering less apparent by hiding the useful information provided by the other variables’.

Possible features to describe German verbs might include any kind of information which helps classify the verbs in a semantically appropriate way. These features include the alternation behaviour of the verbs, their morphological properties, their auxiliary selection, adverbial combinations, etc. Within this thesis, I concentrate on defining the verb features with respect to the alternation behaviour, because I consider the **alternation behaviour** a key component for verb classes as defined in Chapter 2. So I rely on the meaning-behaviour relationship for verbs and use empirical verb properties at the **syntax-semantic interface** to describe the German verbs.

The verbs are described on three levels at the syntax-semantic interface, each of them refining the previous level by additional information. The first level encodes a purely syntactic definition of verb subcategorisation, the second level encodes a syntactico-semantic definition of subcategorisation with prepositional preferences, and the third level encodes a syntactico-semantic definition of subcategorisation with prepositional and selectional preferences. So the refinement of verb features starts with a purely syntactic definition and step-wise adds semantic information. The most elaborated description comes close to a definition of the verb alternation behaviour. I have decided on this three step proceeding of verb descriptions, because the resulting clusters and even more the changes in clustering results which come with a change of features should provide insight into the meaning-behaviour relationship at the syntax-semantic interface. The exact choice of the features is presented and discussed in detail in the experiment setup in Chapter 5.

The representation of the verbs is realised by vectors which describe the verbs by distributions over their features. As explained in Chapter 1, the distributional representation of features for natural language objects is widely used and has been justified by Harris (1968). The feature values for the distributions are provided by the German grammar, as described in Chapter 3. The distributions refer to (i) real values f representing frequencies of the features with $0 \leq f$, (ii) real values p representing probabilities of the features with $0 \leq p \leq 1$, and (iii) binary values b with $b \in \{0, 1\}$. Generally speaking, a standardisation of measurement units which converts the original measurements (such as frequencies) to unitless variables (such as probabilities) on the one hand may be helpful by avoiding the preference of a specific unit, but on the other hand might dampen the clustering structure by eliminating the absolute value of the feature.

4.1.3 Data Similarity Measures

With the data objects and their features specified, a means for comparing the objects is needed. The German verbs are described by features at the syntax-semantic interface, and the features are represented by a distributional feature vector. A range of measures calculates either the distance d or the similarity sim between two objects x and y. The notions of ‘distance’ and ‘similarity’ are related, since the smaller the distance between two objects, the more similar they are to each other. All measures refer to the feature values in some way, but they consider different properties of the feature vector. There is no optimal similarity measure, since the usage depends on the task. Following, I present a range of measures which are commonly used for calculating the similarity of distributional objects. I will use all of the measures in the clustering experiments.

Minkowski Metric The *Minkowski metric* or L_q *norm* calculates the distance d between the two objects x and y by comparing the values of their n features, cf. Equation (4.1). The Minkowski metric can be applied to frequency, probability and binary values.

$$d(x, y) = L_q(x, y) = \sqrt[q]{\sum_{i=1}^{n} (x_i - y_i)^q} \tag{4.1}$$

Two important special cases of the Minkowski metric are $q = 1$ and $q = 2$, cf. Equations (4.2) and (4.3).

- *Manhattan distance* or *City block distance* or L_1 *norm*:

$$d(x, y) = L_1 = \sum_{i=1}^{n} |x_i - y_i| \tag{4.2}$$

- *Euclidean distance* or L_2 *norm*:

$$d(x, y) = L_2 = \sqrt{\sum_{i=2}^{n} (x_i - y_i)^2} \tag{4.3}$$

Kullback-Leibler Divergence The *Kullback-Leibler divergence (KL)* or *relative entropy* is defined in Equation (4.4). KL is a measure from information theory which determines the inefficiency of assuming a model distribution given the true distribution (Cover and Thomas, 1991). It is generally used for x and y representing probability mass functions, but I will also apply the measure to probability distributions with $\sum_i x_i > 1$ and $\sum_i y_i > 1$.

$$d(x,y) = D(x||y) = \sum_{i=1}^{n} x_i \ * \ log \ \frac{x_i}{y_i} \tag{4.4}$$

The Kullback-Leibler divergence is not defined in case $y_i = 0$, so the probability distributions need to be smoothed. Two variants of KL, *information radius* in Equation (4.5) and *skew divergence* in Equation (4.6), perform a default smoothing. Both variants can tolerate zero values in the distribution, because they work with a weighted average of the two distributions compared. Lee (2001) has recently shown that the skew divergence is an effective measure for distributional similarity in NLP. Related to Lee, I set the weight w for the skew divergence to 0.9.

$$d(x,y) = IRad(x,y) = D(x||\frac{x+y}{2}) \ + \ D(y||\frac{x+y}{2}) \tag{4.5}$$

$$d(x,y) = Skew(x,y) = D(x||w * y \ + \ (1-w) * x) \tag{4.6}$$

τ coefficient Kendall's τ *coefficient* (Kendall, 1993) compares all feature pairs of the two objects x and y in order to calculate their distance. If $\langle x_i, y_i \rangle$ and $\langle x_j, y_j \rangle$ are two pairs of the features i and j for the objects x and y, the pairs are concordant if $x_i > x_j$ and $y_i > y_j$ or if $x_i < x_j$ and $y_i < y_j$, and the pairs are discordant if $x_i > x_j$ and $y_i < y_j$ or if $x_i < x_j$ and $y_i > y_j$. If the distributions of the two objects are similar, a large number of concordances f_c is expected, otherwise a large number of discordances f_d is expected. τ is defined in Equation (4.7), with p_c the probability of concordances and p_d the probability of discordances; τ ranges from -1 to 1. The τ coefficient can be applied to frequency and probability values. Hatzivassiloglou and McKeown (1993) use τ to measure the similarity between adjectives.

$$sim(x,y) \ = \ \tau(x,y) \ = \ \frac{f_c}{f_c \ + \ f_d} \ - \ \frac{f_d}{f_c \ + \ f_d} \ = \ p_c \ - \ p_d \tag{4.7}$$

Cosine $cos(x,y)$ measures the similarity of the two objects x and y by calculating the *cosine of the angle* between their feature vectors. The degrees of similarity range from -1 (highest degree of dissimilarity with vector angle = $180°$) over 0 (angle = $90°$) to 1 (highest degree of similarity with vector angle = $0°$). For positive feature values, the cosine lies between 0 and 1. The cosine measure can be applied to frequency, probability and binary values.

$$sim(x,y) = cos(x,y) = \frac{\sum_{i=1}^{n} x_i * y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} * \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{4.8}$$

Binary Distance Measures In addition, there are specific measures for binary distributions. The following list is taken from Manning and Schütze (1999). The measures are defined on basis of the feature sets X and Y for the objects x and y, respectively. Referring to the notion of set intersection and set union, the agreement and disagreement of the feature values is measured.

- The *matching coefficient* counts the dimensions on which both vectors are non-zero.

$$sim(x,y) = match(x,y) = |X \cap Y| = \sum_{i=1}^{n} |x_i = y_i = 1| \tag{4.9}$$

- The *Dice coefficient* normalises the matching coefficient for length by dividing by the total number of non-zero entries.

$$sim(x,y) = dice(x,y) = \frac{2 * |X \cap Y|}{|X| + |Y|} = \frac{\sum_{i=1}^{n} |x_i = y_i = 1|}{\sum_{i=1}^{n} |x_i = 1| \ + \ \sum_{i=1}^{n} |y_i = 1|} \tag{4.10}$$

- The *Jaccard coefficient* or *Tanimoto coefficient* penalises a small number of shared entries (as a proportion of all non-zero entries) more than the Dice coefficient does.

$$sim(x,y) = jaccard(x,y) = \frac{|X \cap Y|}{|X \cup Y|} = \frac{\sum_{i=1}^{n} |x_i = y_i = 1|}{\sum_{i=1}^{n} |(x_i = 1) \vee (y_i = 1)|} \tag{4.11}$$

- The *overlap coefficient (ol)* has a value of 1 if every feature with a non-zero value for the first object is also non-zero for the second object or vice versa, i.e. $X \subseteq Y$ or $Y \subseteq X$.

$$sim(x,y) = ol(x,y) = \frac{|X \cap Y|}{min(|X|, |Y|)} = \frac{\sum_{i=1}^{n} |x_i = y_i = 1|}{min(\sum_{i=1}^{n} |x_i = 1|, \sum_{i=1}^{n} |y_i = 1|)} \tag{4.12}$$

4.1.4 Clustering Algorithms

Clustering is a task for which many algorithms have been proposed. No clustering technique is universally applicable, and different techniques are in favour for different clustering purposes. So an understanding of both the clustering problem and the clustering technique is required to apply a suitable method to a given problem. In the following, I describe general parameters of a clustering technique which are relevant to the task of inducing a verb classification.

- Parametric design:

  Assumptions may (but need not) be made about the form of the distribution used to model the data by the cluster analysis. The parametric design should be chosen with respect to the nature of the data. It is often convenient to assume, for example, that the data can be modelled by a multivariate Gaussian.

- Position, size, shape and density of the clusters:

  The experimenter might have an idea about the desired clustering results with respect to the position, size, shape and density of the clusters. Different clustering algorithms have different impact on these parameters, as the description of the algorithms will show. Therefore, varying the clustering algorithm influences the design parameters.

- Number of clusters:

  The number of clusters can be fixed if the desired number is known beforehand (e.g. because of a reference to a gold standard), or can be varied to find the optimal cluster analysis. As Duda *et al.* (2000) state, ‘In theory, the clustering problem can be solved by exhaustive enumeration, since the sample set is finite, so there are only a finite number of possible partitions; in practice, such an approach is unthinkable for all but the simplest problems, since there are at the order of $\frac{k^n}{k!}$ ways of partitioning a set of n elements into k subsets’.

- Ambiguity:

  Verbs can have multiple senses, requiring them being assigned to multiple classes. This is only possible by using a soft clustering algorithm, which defines cluster membership probabilities for the clustering objects. A hard clustering algorithm performs a *yes/no* decision on object membership and cannot model verb ambiguity, but it is easier to use and interpret.

The choice of a clustering algorithm determines the setting of the parameters. In the following paragraphs, I describe a range of clustering algorithms and their parameters. The algorithms are divided into (A) hierarchical clustering algorithms and (B) partitioning clustering algorithms. For each type, I concentrate on the algorithms used in this thesis and refer to further possibilities.

A) Hierarchical Clustering

Hierarchical clustering methods impose a hierarchical structure on the data objects and their step-wise clusters, i.e. one extreme of the clustering structure is only one cluster containing all objects, the other extreme is a number of clusters which equals the number of objects. To obtain a certain number k of clusters, the hierarchy is cut at the relevant depth. Hierarchical clustering is a rigid procedure, since it is not possible to re-organise clusters established in a previous step. The original concept of a hierarchy of clusters creates hard clusters, but as e.g. Lee (1997) shows that the concept may be transferred to soft clusters.

Depending on whether the clustering is performed top-down, i.e. from a single cluster to the maximum number of clusters, or bottom-up, i.e. from the maximum number of clusters to a single cluster, we distinguish divisive and agglomerative clustering. Divisive clustering is computationally more problematic than agglomerative clustering, because it needs to consider all possible divisions into subsets. Therefore, only agglomerative clustering methods are applied in this thesis. The algorithm is described in Figure 4.1.

```
1   Given: a set of objects $O = \{o_1, ..., o_n\} \subseteq \mathbb{R}^m$;
            a function for distance measure $d : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$
2   for all objects $o_i \in O$ do
3       establish cluster $C_i = \{o_i\}$
4   let $C = \{C_1, ..., C_n\}$
5   while $|C| \neq 1$ do
6       for all pairs of clusters $\langle C_i, C_{j \neq i} \rangle \in C \times C$ do
7           calculate $d(C_i, C_j)$
8       let $best(C_i, C_j) = \forall \langle C_{k \neq i}, C_{l \neq k,j} \rangle \in C \times C : [d(C_i, C_j) \leq d(C_k, C_l)]$
9       for $best(C_i, C_j)$ do
10          let $C_{ij} = C_i \cup C_j$
11          let $C^{new} = C \setminus \{C_i, C_j\}$
12          let $C = C^{new} \cup C_{ij}$
13  end
```

Figure 4.1: Algorithm for agglomerative hierarchical clustering

The algorithm includes a notion of measuring and comparing distances between clusters (step 7). So far, I have introduced measures for object distance and similarity in Section 4.1.3, but I have not introduced measures for cluster distance. The concept of cluster distance is based on the concept of object distance, but refers to different ideas of cluster amalgamation. Below, five well-known measures for cluster amalgamation are introduced. All of them are used in the clustering experiments.

Nearest Neighbour Cluster Distance The distance d between two clusters C_i and C_j is defined as the minimum distance between the cluster objects, cf. Equation (4.13). The cluster distance measure is also referred to as *single-linkage*. Typically, it causes a chaining effect concerning the shape of the clusters, i.e. whenever two clusters come close too each other, they stick together even though some members might be far from each other.

$$d(C_i, C_j) = d_{min}(C_i, C_j) = min_{x \in C_i, y \in C_j} d(x, y) \tag{4.13}$$

Furthest Neighbour Cluster Distance The distance d between two clusters C_i and C_j is defined as the maximum distance between the cluster objects, cf. Equation (4.14). The cluster distance measure is also referred to as *complete-linkage*. Typically, it produces compact clusters with small diameters, since every object within a cluster is supposed to be close to every other object within the cluster, and outlying objects are not incorporated.

$$d(C_i, C_j) = d_{max}(C_i, C_j) = max_{x \in C_i, y \in C_j} d(x, y) \tag{4.14}$$

Distance between Cluster Centroids The distance d between two clusters C_i and C_j is defined as the distance between the cluster centroids cen_i and cen_j, cf. Equation (4.15). The centroid of a cluster is determined as the average of objects in the cluster, i.e. each feature of the centroid vector is calculated as the average feature value of the vectors of all objects in the cluster. The cluster distance measure is a natural compromise between the nearest and the furthest neighbour cluster distance approaches. Different to the above approaches, it does not impose a structure on the clustering effect.

$$d(C_i, C_j) = d_{mean}(C_i, C_j) = d(cen_i, cen_j) \tag{4.15}$$

Average Distance between Clusters The distance d between two clusters c_i and c_j is defined as the average distance between the cluster objects, cf. Equation (4.16). Like d_{mean}, the cluster distance measure is a natural compromise between the nearest and the furthest neighbour cluster distance approaches. It does not impose a structure on the clustering effect either.

$$d(C_i, C_j) = d_{avg}(C_i, C_j) = \frac{1}{|C_i| * |C_j|} * \sum_{x \in C_i} \sum_{y \in C_j} d(x, y) \tag{4.16}$$

Ward's Method The distance d between two clusters C_i and C_j is defined as the loss of information (or: the increase in error) in merging two clusters (Ward, 1963), cf. Equation (4.17). The error of a cluster C is measured as the sum of distances between the objects in the cluster and the cluster centroid cen_C. When merging two clusters, the error of the merged cluster is larger than the sum or errors of the two individual clusters, and therefore represents a loss of information. But the merging is performed on those clusters which are most homogeneous, to unify clusters such that the variation inside the merged clusters increases as little as possible. Ward's method tends to create compact clusters of small size. It is a least squares method, so implicitly assumes a Gaussian model.

$$d(C_i, C_j) = d_{ward}(C_i, C_j) = \sum_{x \in (C_i \cup C_j)} d(x, cen_{ij}) \; - \; [\sum_{x \in C_i} d(x, cen_i) \; + \; \sum_{x \in cen_j} d(x, cen_j)] \tag{4.17}$$

B) Partitioning Clustering

Partitioning clustering methods partition the data object set into clusters where every pair of object clusters is either distinct (hard clustering) or has some members in common (soft clustering). Partitioning clustering begins with a starting cluster partition which is iteratively improved until a locally optimal partition is reached. The starting clusters can be either random or the cluster output from some clustering pre-process (e.g. hierarchical clustering). In the resulting clusters, the objects in the groups together add up to the full object set.

k-Means Clustering The k-Means clustering algorithm is an unsupervised hard clustering method which assigns the n data objects $o_1, ..., o_n$ to a pre-defined number of exactly k clusters $C_1, ..., C_k$. Initial verb clusters are iteratively re-organised by assigning each verb to its closest cluster (centroid) and re-calculating cluster centroids until no further changes take place. The optimising criterion in the clustering process is the sum-of-squared-error E between the objects in the clusters and their respective cluster centroids $cen_1, ..., cen_k$, cf. Equation (4.18).

$$E = \sum_{i=1}^{k} \sum_{o \in C_i} d(o, cen_i)^2 \tag{4.18}$$

The k-Means algorithm is sensitive to the selection of the initial partition, so the initialisation should be varied. k-Means imposes a Gaussian parametric design on the clustering result and generally works well on data sets with isotropic cluster shape, since it tends to create compact clusters. The time complexity of k-Means is $O(n)$ with n the number of objects. Several variants of the k-Means algorithm exist. Within this thesis, clustering is performed by the k-Means algorithm as proposed by Forgy (1965). Figure 4.2 defines the algorithm.

```
Given: a set of objects $O = \{o_1, ..., o_n\} \subseteq \mathbb{R}^m$;
         a function for distance measure $d : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$;
         a (random/pre-processed) clustering partition $C = \{C_1, ..., C_k\}$
do
    for all clusters $C_i \in C$ do
        calculate cluster centroid $cen_i \subseteq \mathbb{R}^m$
    for all objects $o \in O$ do
        for all clusters $C_i \in C$ do
            calculate $d(o, C_i) = d(o, cen_i)$
        let $best(o, C_o) = \forall C_j \in C : [d(o, cen_{C_o}) \leq d(o, cen_{C_j})]$
    undefine $C^{new}$
    for all objects $o \in O$ do
        for $best(o, C_o)$ do
            let $o \in C_o^{new}$
    if $C \neq C^{new}$ then $change = true$
    else $change = false$
until $change = false$
```

Figure 4.2: Algorithm for k-Means clustering

Other Clustering Methods k-Means is a hard clustering algorithm. But some clustering problems require the clustering objects being assigned to multiple classes. For example, to model verb ambiguity one would need a soft clustering algorithm. Examples for soft clustering algorithms which are based on the same data model as k-Means are such as fuzzy clustering (Zadeh, 1965), cf. also Höppner *et al.* (1997) and Duda *et al.* (2000), and the *Expectation-Maximisation (EM) Algorithm* (Baum, 1972) which can also be implemented as a soft version of k-Means with an underlying Gaussian model.

The above methods represent a standard choice for clustering in pattern recognition, cf. Duda *et al.* (2000). Clustering techniques with different background are e.g. the *Nearest Neighbour Algorithm* (Jarvis and Patrick, 1973), *Graph-Based Clustering* (Zahn, 1971), and *Artificial Neural Networks* (Hertz *et al.*, 1991). Recently, elaborated techniques from especially image processing have been transfered to linguistic clustering, such as *Spectral Clustering* (Brew and Schulte im Walde, 2002).

C) Decision on Clustering Algorithm

Within the scope of this thesis, I apply the hard clustering technique k-Means to the German verb data. I decided to use the k-Means algorithm for the clustering, because it is a standard clustering technique with well-known properties. In addition, see the following arguments.

- The parametric design of Gaussian structures realises the idea that objects should belong to a cluster if they are very similar to the centroid as the average description of the cluster, and that an increasing distance refers to a decrease in cluster membership. In addition, the isotropic shape of clusters reflects the intuition of a compact verb classification.

- A variation of the clustering initialisation performs a variation of the clustering parameters such as position, size, shape and density of the clusters. Even though I assume that an appropriate parametric design for the verb classification is given by isotropic cluster formation, a variation of initial clusters investigates the relationship between clustering data and cluster formation. I will therefore apply random initialisations and hierarchical clusters as input to k-Means.

- Selim and Ismail (1984) prove for distance metrices (a subset of the similarity measures in Section 4.1.3) that k-Means finds locally optimal solutions by minimising the sum-of-squared-error between the objects in the clusters and their respective cluster centroids.

- Starting clustering experiments with a hard clustering algorithm is an easier task than applying a soft clustering algorithm, especially with respect to a linguistic investigation of the experiment settings and results. Ambiguities are a difficult problem in linguistics, and are subject to future work. I will investigate the impact of the hard clustering on polysemous verbs, but not try to model the polysemy within this work.

- As to the more general question whether to use a supervised classification or an unsupervised clustering method, this work concentrates on minimising the manual intervention in the automatic class acquisition. A classification would require costly manual labelling (especially with respect to a large-scale classification) and not agree with the exploratory goal of finding as many independent linguistic insights as possible at the syntax-semantic interface of verb classifications.

4.2 Clustering Evaluation

A clustering evaluation demands an independent and reliable measure for the assessment and comparison of clustering experiments and results. In theory, the clustering researcher has acquired an intuition for the clustering evaluation, but in practise the mass of data on the one hand and the subtle details of data representation and clustering algorithms on the other hand make an intuitive judgement impossible. An intuitive, introspective evaluation can therefore only be plausible for small sets of objects, but large-scale experiments require an objective method.

There is no absolute scheme with which to measure clusterings, but a variety of evaluation measures from diverse areas such as theoretical statistics, machine vision and web-page clustering are applicable. In this section, I provide the definition of various clustering evaluation measures and evaluate them with respect to their linguistic application. Section 4.2.1 describes the demands I expect to fulfill with an evaluation measure on verb clusterings. In Section 4.2.2 I present a range of possible evaluation methods, and Section 4.2.3 compares the measures against each other and according to the evaluation demands.

4.2.1 Demands on Clustering Evaluation

An objective method for evaluating clusterings should be independent of the evaluator and reliable concerning its judgement about the quality of the clusterings. How can we transfer these abstract descriptions to more concrete demands? Following, I define demands on the task of clustering verbs into semantic classes, with an increasing proportion of linguistic task specificity. I.e. I first define general demands on an evaluation, then general demands on a clustering evaluation, and finally demands on the verb-specific clustering evaluation.

The demands on the clustering evaluation are easier described with reference to the formal notation of clustering result and gold standard classification, so the notation is provided in advance:

Definition 4.1 *Given an object set $O = \{o_1, ..., o_n\}$ with n objects, the clustering result and the manual classification as the gold standard represent two partitions of O with $C = \{C_1, ..., C_k\}$ and $M = \{M_1, M_2, ..., M_l\}$, respectively. $C_i \in C$ denotes the set of objects in the ith cluster of partition C, and $M_j \in M$ denotes the set of objects in the jth cluster of partition M.*

General Evaluation Demands Firstly, I define a demand on evaluation in general: The evaluation of an experiment should be proceeded against a gold standard, as independent and reliable as possible. My gold standard is the manual classification of verbs, as described in Chapter 2. The classification has been created by the author. To compensate for the sub-optimal setup by a single person, the classification was developed in close relation to the existing classifications for German by Schumacher (1986) and English by Levin (1993). In addition, the complete classification was finished before any experiments on the verbs were performed.

General Clustering Demands The second range of demands refers to general properties of a cluster analysis, independent of the clustering area.

- Since the purpose of the evaluation is to assess and compare different clustering experiments and results, the measure should be applicable to all similarity measures used in clustering, but possibly independent of the respective similarity measure.
- The evaluation result should define a (numerical) measure indicating the value of the clustering. The resulting value should either be easy to interpret or otherwise be illustrated with respect to its range and effects, in order to facilitate the evaluation interpretation.
- The evaluation method should be defined without a bias towards a specific number and size of clusters.
- The evaluation measure should distinguish the quality of (i) the whole clustering partition C, and (ii) the specific clusters $C_i \in C$.

Linguistic Clustering Demands The fact that this thesis is concerned with the clustering of linguistic data sharpens the requirements on an appropriate clustering evaluation, because the demands on verb classes are specific to the linguistic background and linguistic intuition and not necessarily desired for different clustering areas. The following list therefore refers to a third range of demands, defined as linguistic desiderata for the clustering of verbs.

(a) The clustering result should not be a single cluster representing the clustering partition, i.e. $|C| = 1$. A single cluster does not represent an appropriate model for verb classes.

(b) The clustering result should not be a clustering partition with only singletons, i.e. $\forall C_i \in C : |C_i| = 1$. A set of singletons does not represent an appropriate model for verb classes either.

(c) Let C_i be a correct (according to the gold standard) cluster with $|C_i| = x$. Compare this cluster with the correct cluster C_j with $|C_j| = y > x$. The evaluated quality of C_j should be better compared to C_i, since the latter cluster was able to create a larger correct cluster, which is a more difficult task.

Example:[1] C_i = *ahnen vermuten wissen*
C_j = *ahnen denken glauben vermuten wissen*

(d) Let C_i be a correct cluster and C_j be a cluster which is identical to C_i, but contains additional objects which do not belong to the same class. The evaluated quality of C_i should be better compared to C_j, since the former cluster contains fewer errors.

Example: C_i = *ahnen vermuten wissen*
C_j = *ahnen vermuten wissen laufen lachen*

[1] In all examples, verbs belonging to the same gold standard class are underlined in the cluster.

(e) Let C_i be a correct cluster with $|C_i| = x$. Compare this cluster with a non-correct cluster C_j with $|C_j| = x$. The evaluated quality of C_i should be better compared to C_j, since being of the same size as C_j the proportion of homogeneous verbs is larger.

Example: C_i = $\underline{\textit{ahnen vermuten wissen}}$
C_j = $\underline{\textit{ahnen vermuten}}$ *laufen*

(f) Let C_i be a correct cluster with $|C_i| = x$. Compare this cluster with the two correct clusters (obviously in a different partition) C_{i_1} and C_{i_2} with $C_i = C_{i_1} \cup C_{i_2}$. The evaluated quality of C_i should be better compared to the sum of qualities of C_{i_1} and C_{i_2}, since the former manages to cluster the same range of homogeneous verbs in the same cluster.

Example: C_i = $\underline{\textit{ahnen denken glauben vermuten wissen}}$
C_{i_1} = $\underline{\textit{ahnen denken glauben}}$
C_{i_2} = $\underline{\textit{vermuten wissen}}$

(g) Let C_{i_1} and C_{i_2} be two correct clusters. Compare these clusters with a single non-correct cluster (obviously in a different partition) C_i with $C_i = C_{i_1} \cup C_{i_2}$. The evaluated quality of C_i should be worse compared to the sum of qualities of C_{i_1} and C_{i_2}, since the smaller clusters are completely correct, whereas C_i merges the clusters into an incoherent set.

Example: C_i = $\underline{\textit{ahnen denken glauben}}$ $\underline{\textit{laufen rennen}}$
C_{i_1} = $\underline{\textit{ahnen denken glauben}}$
C_{i_2} = $\underline{\textit{laufen rennen}}$

Some of the linguistically defined demands are also subject to general clustering demands, but nevertheless included in the more specific cases.

The linguistically most distinctive demand on the clustering evaluation deserves specific attention. It refers to the representation of verb ambiguities, both in the manual and induced classifications. Two scenarios of verb ambiguity are possible:

1. The manual classification contains verb ambiguity, i.e. there are polysemous verbs which belong to more than one verb class. The cluster analysis, on the other hand, is based on a hard clustering algorithm, i.e. each verb is only assigned to one cluster.

2. The manual classification contains verb ambiguity, and the cluster analysis is based on a soft clustering algorithm, i.e. both verb sets contain verbs which are possibly assigned to multiple classes.

The third possible scenario, that the manual classification is without verb ambiguity, but the cluster analysis is a soft clustering, is not taken into consideration, since it is linguistically uninteresting. The second scenario is relevant for a soft clustering technique, but since this thesis is restricted to a hard clustering technique, we can concentrate on scenario 1: the manual classification as defined in Chapter 2 contains polysemous verbs, but k-Means only produces hard clusters.

4.2.2 Description of Evaluation Measures

In the following, I describe a range of possible evaluation measures, with different theoretical backgrounds and demands. The overview does, of course, not represent an exhaustive list of clustering evaluations, but tries to give an impression of the variety of possible methods which are concerned with clustering and clustering evaluation. Not all of the described measures are applicable to our clustering task, so a comparison and choice of the candidate methods will be provided in Section 4.2.3.

Contingency Tables Contingency tables are a typical means for describing and defining the association between two partitions. As they will be of use in a number of evaluation examples below, their notation is given beforehand.

Definition 4.2 *A $C \times M$ contingency table is a $C \times M$ matrix with rows $C_i, 1 \leq i \leq k$ and columns $M_j, 1 \leq j \leq l$. t_{ij} denotes the number of objects that are common to the set C_i in partition C (the clustering result) and the set M_j in partition M (the manual classification). Summing over the row or column values gives the marginal values $t_{i.}$ and $t_{.j}$, referring to the number of objects in classes C_i and M_j, respectively. Summing over the marginal values results in the total number of n objects in the clustering task.*

The number of pairs with reference to a specific matrix value x is calculated by $\binom{x}{2}$; the pairs are of special interest for a convenient calculation of evaluation results. For illustration purposes, a $C \times M$ contingency table is described by an example:

$$M = \{M_1 = \{a, b, c\}, M_2 = \{d, e, f\}\}$$
$$C = \{C_1 = \{a, b\}, C_2 = \{c, d, e\}, C_3 = \{f\}\}$$

$C \times M$ contingency table:

	M_1	M_2	
C_1	$t_{11} = 2$	$t_{12} = 0$	$t_{1.} = 2$
C_2	$t_{21} = 1$	$t_{22} = 2$	$t_{2.} = 3$
C_3	$t_{31} = 0$	$t_{32} = 1$	$t_{3.} = 1$
	$t_{.1} = 3$	$t_{.2} = 3$	$n = 6$

The number of pairs within the cells of the contingency tables is as follows.

	M_1	M_2	
C_1	$\binom{t_{11}}{2} = 1$	$\binom{t_{12}}{2} = 0$	$\binom{t_{1.}}{2} = 1$
C_2	$\binom{t_{21}}{2} = 0$	$\binom{t_{22}}{2} = 1$	$\binom{t_{2.}}{2} = 3$
C_3	$\binom{t_{31}}{2} = 0$	$\binom{t_{32}}{2} = 0$	$\binom{t_{3.}}{2} = 0$
	$\binom{t_{.1}}{2} = 3$	$\binom{t_{.2}}{2} = 3$	$\binom{n}{2} = 15$

Sum-of-Squared-Error Criterion

Summing over the squared distances between the clustering objects and their cluster representatives (i.e. the respective cluster centroids) is a standard cost function. The evaluation defines a measure for the homogeneity of the clustering results with respect to the object description data, but without reference to a gold standard.

The sum-of-squared-error E originally refers to Euclidean distance, but is applicable to further distance measures. The definition was given in Equation (4.18) and is repeated in Equation (4.19), with the cluster centroid of cluster C_i abbreviated as cen_i.

$$E(C) = \sum_{i=1}^{k} \sum_{o \in C_i} d(o, cen_i)^2 \qquad (4.19)$$

Silhouette Value

Kaufman and Rousseeuw (1990, pages 83ff) present the silhouette plot as a means for clustering evaluation. With this method, each cluster is represented by a silhouette displaying which objects lie well within the cluster and which objects are marginal to the cluster. The evaluation method also refers to the object data, but not to a gold standard.

To obtain the silhouette value sil for an object o_i within a cluster C_A, we compare the average distance a between o_i and all other objects in C_A with the average distance b between o_i and all objects in the neighbour cluster C_B, cf. Equations 4.20 to 4.22. For each object o_i applies $-1 \leq sil(o_i) \leq 1$. If $sil(o_i)$ is large, the average object distance within the cluster is smaller than the average distance to the objects in the neighbour cluster, so o_i is well classified. If $sil(o_i)$ is small, the average object distance within the cluster is larger than the average distance to the objects in the neighbour cluster, so o_i has been misclassified.

$$a(o_i) = \frac{1}{|C_A| - 1} \sum_{o_j \in C_A, o_j \neq o_i} d(o_i, o_j) \qquad (4.20)$$

$$b(o_i) = min_{C_B \neq C_A} \frac{1}{|C_B|} \sum_{o_j \in C_B} d(o_i, o_j) \qquad (4.21)$$

$$sil(o_i) = \frac{b(o_i) - a(o_i)}{max\{a(o_i), b(o_i)\}} \qquad (4.22)$$

In addition to providing information about the quality of classification of a single object, the silhouette value can be extended to evaluate the individual clusters and the entire clustering. The *average silhouette width* $sil(C_i)$ of a cluster C_i is defined as the average silhouette value for all objects within the cluster, cf. Equation 4.23, and the *average silhouette width for the entire data set* with k clusters $\overline{sil(k)}$ is defined as the average silhouette value for the individual clusters, cf. Equation 4.24.

$$sil(C_i) = \frac{1}{|C_i|} \sum_{o_j \in C_i} sil(o_j) \tag{4.23}$$

$$sil(C) = \overline{sil(k)} = \frac{1}{k} \sum_{i=1}^{k} sil(C_i) \tag{4.24}$$

Class-based Precision and Recall

What I call a class-based P/R evaluation has originally been defined by Vilain *et al.* (1995) as scoring scheme for the coreference task in MUC6. The evaluation method considers both the clustering and the manual classification as equivalence classes which are defined by the particular object links which are necessary to encode the equivalence relations. The precision and recall scores are obtained by calculating the least number of object links required to align the equivalence classes.

Let $c(M_i)$ be the minimal number of correct object links which are necessary to generate the equivalence class M_i in the manual classification: $c(M_i) = |M_i| - 1$. With $|p(M_i)|$ the number of classes in the clustering partition containing any of the objects in M_i, the number of missing object links in the clustering which are necessary to fully reunite the objects of class M_i is $m(M_i) = |p(M_i)| - 1$. Recall for a single cluster is defined as the proportion of existing object links of the relevant cluster compared to the minimal number of correct object links.

$$recall(M_i) = \frac{c(M_i) \ - \ m(M_i)}{c(M_i)} = \frac{|M_i| \ - \ |p(M_i)|}{|M_i| \ - \ 1} \tag{4.25}$$

Extending the measure from a single equivalence class to the entire classification of the object set S is realised by summing over the equivalence classes:

$$recall_S(C, M) = \frac{\sum_i \ |M_i| \ - \ |p(M_i)|}{\sum_i \ |M_i| \ - \ 1} \tag{4.26}$$

In the case of precision, we consider the equivalence classes C_i in the clustering and calculate the existing and missing object links in the manual classification with respect to the clustering.

$$precision(C_i) = \frac{c(C_i) \ - \ m(C_i)}{c(C_i)} = \frac{|C_i| \ - \ |p(C_i)|}{|C_i| \ - \ 1} \tag{4.27}$$

$$precision_S(C, M) = \frac{\sum_i \ |C_i| \ - \ |p(C_i)|}{\sum_i \ |C_i| \ - \ 1} \tag{4.28}$$

Classification		Clustering		Evaluation
Class	Link	Class	Link	
$M_1 = \{a, b, c\}$	a-b,	$C_1 = \{a, b\}$	a-b	$recall_S(C, M) = \frac{(3-2)\,+\,(3-2)}{(3-1)\,+\,(3-1)} = \frac{2}{4} = \frac{1}{2}$
	b-c			$precision_S(C, M) =$
$M_2 = \{d, e, f\}$	d-e,	$C_2 = \{c, d, e\}$	c-d,	$\frac{(2-1)\,+\,(3-2)\,+\,(1-1)}{(2-1)\,+\,(3-1)\,+\,(1-1)} = \frac{2}{3}$
	e-f		d-e	$f - score_S(C, M) = \frac{2*\frac{1}{2}*\frac{2}{3}}{\frac{1}{2}+\frac{2}{3}} = \frac{4}{7}$
		$C_3 = \{f\}$		

Table 4.1: Example evaluation for class-based P/R

The $f - score_S$ as given in Equation 4.29 is the harmonic mean between $precision_S$ and $recall_S$.

$$f - score_S(C, M) = \frac{2\ *\ recall_S\ *\ precision_S}{recall_S\ +\ precision_S} \tag{4.29}$$

Pair-wise Precision and Recall

Being closest to my clustering area, Hatzivassiloglou and McKeown (1993) present an evaluation method for clustering in NLP: they define and evaluate a cluster analysis of adjectives. The evaluation is based on common cluster membership of object pairs in the clustering and the manual classification. On the basis of common cluster membership, recall and precision numbers are calculated in the standard way, cf. Equations (4.30) and (4.31). True positives tp are the number of common pairs in M and C, false positives fp the number of pairs in C, but not M, and false negatives fn the number of pairs in M, but not C. I add the f-score as harmonic mean between recall and precision, as above. Table 4.2 presents an example of pair-wise precision and recall calculation.

$$recall = \frac{tp}{fn\ +\ tp} \tag{4.30}$$

$$precision = \frac{tp}{fp\ +\ tp} \tag{4.31}$$

Adjusted Pair-wise Precision

Pair-wise precision and recall calculation (see above) shows some undesired properties concerning my linguistic needs, especially concerning the recall value. I therefore use the precision value and adjust the measure by a scaling factor based on the size of the respective cluster. The definition of the adjusted pair-wise precision is given in Equation (4.32). A correct pair refers to a verb

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$	$C_1 = \{a, b\}$	number of common pairs in M and C (tp): 2
$M_2 = \{d, e, f\}$	$C_2 = \{c, d, e\}$	number of pairs in classification M ($fn + tp$): 6
	$C_3 = \{f\}$	number of pairs in clustering C ($fp + tp$): 4
		$recall = \frac{2}{6} = \frac{1}{3}$
		$precision = \frac{2}{4} = \frac{1}{2}$
		$f - score = \frac{2*\frac{1}{3}*\frac{1}{2}}{\frac{1}{3}+\frac{1}{2}} = \frac{2}{5}$

Table 4.2: Example evaluation for pair-wise P/R

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$	$C_1 = \{a, b\}$	$APP(C_1) = \frac{1}{3}$
$M_2 = \{d, e, f\}$	$C_2 = \{c, d, e\}$	$APP(C_2) = \frac{1}{4}$
	$C_3 = \{f\}$	$APP(C_3) = 0$
		$APP(C) = \frac{1}{3} * (\frac{\frac{1}{3}}{2} + \frac{\frac{1}{4}}{3}) = \frac{1}{3} * (\frac{1}{6} + \frac{1}{12}) = \frac{1}{12}$

Table 4.3: Example evaluation for adjusted pair-wise precision

pair which is correct according to the gold standard. The evaluation measure of the whole clustering is calculated by taking the weighted average over the qualities of the individual clusters, as defined in Equation (4.33). By inserting $|C_i|^{-1}$ as weight for each cluster $APP(C_i)$ I calculate the average contribution of each verb to $APP(C_i)$. And since the overall sum of $APP(C, M)$ for the clustering is first summed over all clusters (and therefore over the average contributions of the verbs) and then divided by the number of clusters, I calculate the average contribution of a verb to the clustering APP. The measure is developed with specific care concerning the linguistic demands, e.g. without the addend $+1$ in the denominator of $APP(C_i)$ the linguistic demands would not be fulfilled. Table 4.3 presents an example of adjusted pair-wise precision.

$$APP(C_i) = \frac{number\ of\ correct\ pairs\ in\ C_i}{number\ of\ verbs\ in\ C_i\ +\ 1} \tag{4.32}$$

$$APP(C, M) = \frac{1}{|C|} \sum_i \frac{APP(C_i)}{|C_i|} \tag{4.33}$$

Mutual Information

The way I define mutual information between the clustering and its gold standard is borrowed from Strehl *et al.* (2000) who assess the similarity of object partitions for the clustering of web documents. Mutual information is a symmetric measure for the degree of dependency between

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$	$C_1 = \{a, b\}$	$purity(C_1) = 1$
$M_2 = \{d, e, f\}$	$C_2 = \{c, d, e\}$	$purity(C_2) = \frac{2}{3}$
	$C_3 = \{f\}$	$purity(C_3) = 1$
		$MI(C, M) =$ $\frac{1}{6} \ * \ (2 * \frac{log\frac{2*6}{2*3}}{log(2*3)} \ + \ ... \ + \ 1 * \frac{log\frac{1*6}{1*3}}{log(2*3)}) = 0.27371$

Table 4.4: Example evaluation for mutual information

the clustering and the manual classification. It is based on the notion of cluster *purity*, which measures the quality of a single cluster C_i referring to p_i^j, the largest number of objects in cluster C_i which C_i has in common with a gold standard verb class M_j, having compared C_i to all gold standard verb classes in M.

$$purity(C_i) = \frac{1}{|C_i|} \, max_j(p_i^j) \tag{4.34}$$

The mutual information score between the clustering C and the manual classification M is based on the shared object membership, with a scaling factor corresponding to the number of objects in the respective clusters, cf. Equation 4.35. The second line in Equation 4.35 relates the definitions by Strehl *et al.* to the notation in the contingency table. Table 4.4 presents an example of mutual information evaluation.

$$\begin{aligned} MI(C, M) \ &= \frac{1}{n} \sum_{i=1}^{k} \sum_{j=1}^{l} p_i^j \ \frac{log(\frac{p_i^j \ * \ n}{\sum_{a=1}^{k} \ p_a^j \ \ \sum_{b=1}^{l} \ p_l^b})}{log(k \ * \ l)} \\ &= \frac{1}{n} \sum_{i=1}^{k} \sum_{j=1}^{l} t_{ij} \ \frac{log(\frac{t_{ij} \ * \ n}{t_{i.} \ * \ t_{.j}})}{log(k \ * \ l)} \end{aligned} \tag{4.35}$$

Rand Index

Rand (1971) defines an evaluation measure for a general clustering problem on basis of agreement vs. disagreement between object pairs in clusterings. He states that clusters are defined as much by those points which they do not contain as by those points which they do contain. Therefore, if the elements of an object-pair are assigned to the same classes in both the clustering and the manual classification, and also if they are assigned to different classes in both partitions, this represents a similarity between the equivalence classes. The similarity evaluation is based on the overlap in class agreement A, compared to the class disagreement D, as defined by Equation (4.36), with $A + D = n$. Table 4.5 presents an example of the Rand index.

$$Rand(C, M) = \frac{\sum_{i<j}^{n} \ \gamma(o_i, o_j)}{\binom{n}{2}} \tag{4.36}$$

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$	$C_1 = \{a, b\}$	agree : number object pairs together in both M and C: 2
$M_2 = \{d, e, f\}$	$C_2 = \{c, d, e\}$	agree : number object pairs separate in both M and C: 7
	$C_3 = \{f\}$	disagree : number object pairs mixed in M and C: 6
		$Rand(C, M) = \frac{2+7}{2+7+6} = \frac{9}{15} = 0.6$

Table 4.5: Example evaluation for Rand index

where

$$\gamma(o_i, o_j) = \begin{cases} 1 & \text{if there exist } C_A \in C \text{ and } M_B \in M \text{ such that objects } o_i \text{ and } o_j \text{ are in } C_A \text{ and } M_B, \\ 1 & \text{if there exist } C_A \in C \text{ and } M_B \in M \text{ such that } o_i \text{ is in both } C_A \text{ and } M_B \\ & \quad \text{while } o_j \text{ is in neither } C_A \text{ or } M_B, \\ 0 & \text{otherwise.} \end{cases} \tag{4.37}$$

Rand Index adjusted by Chance

Hubert and Arabie (1985) argue for a correction of the Rand index for chance, in the sense that the index would take on some constant value (e.g. zero) under an appropriate null model of how the partitions have been chosen. According to Hubert and Arabie, the most obvious model for randomness assumes that the $C \times M$ contingency table is constructed from the generalised hyper-geometric distribution, i.e. the C and M partitions are picked at random, given the original number of classes and objects.

The general form of an index corrected for chance is given in Equation (4.38).[2] The *index* refers to the observed number of object pairs on which the partitions agree. The expected number of object pairs with class agreement attributable to a particular cell in the contingency table is defined by the number of pairs in the row times the number of pairs in the column divided by the total number of pairs, cf. Equation (4.39). The maximum number of object pairs is given by the average number of possible pairs in the clustering and the manual classification. Other possibilities for the maximum index would be e.g. the minimum of the possible pairs in clustering and manual classification $min(\sum_i \binom{t_{i\cdot}}{2}, \sum_j \binom{t_{\cdot j}}{2})$ or simply the possible pairs in the manual classification $\sum_j \binom{t_{\cdot j}}{2}$ when considering the manual classification as the optimum. The corrected Rand index is given in Equation (4.40). The range of R_{adj} is $0 \leq R_{adj} \leq 1$, with only extreme cases below zero. Table 4.6 presents an example.

$$Index_{adj} = \frac{Index \; - \; Expected\,Index}{Maximum\,Index \; - \; Expected\,Index} \tag{4.38}$$

[2] In psychological literature, the index is referred to as *kappa statistic* (Cohen, 1960).

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$ $M_2 = \{d, e, f\}$	$C_1 = \{a, b\}$ $C_2 = \{c, d, e\}$ $C_3 = \{f\}$	$Rand_{adj} = \frac{2 - \frac{4*6}{15}}{\frac{1}{2}(4+6) - \frac{4*6}{15}} = \frac{2 - \frac{8}{5}}{5 - \frac{8}{5}} = 0.11765$

Table 4.6: Example evaluation for adjusted Rand index

$$Exp\binom{t_{ij}}{2} = \frac{\binom{t_{i.}}{2}\binom{t_{.j}}{2}}{\binom{n}{2}} \tag{4.39}$$

$$Rand_{adj}(C, M) = \frac{\sum_{i,j}\binom{t_{ij}}{2} - \frac{\sum_i \binom{t_{i.}}{2} \sum_j \binom{t_{.j}}{2}}{\binom{n}{2}}}{\frac{1}{2}\left(\sum_i \binom{t_{i.}}{2} + \sum_j \binom{t_{.j}}{2}\right) - \frac{\sum_i \binom{t_{i.}}{2} \sum_j \binom{t_{.j}}{2}}{\binom{n}{2}}} \tag{4.40}$$

Matching Index

Fowlkes and Mallows (1983) define another evaluation method based on contingency tables. Their motivation is to define a measure of similarity between two hierarchical clusterings, as a sequence of measures which constitute the basis for a plotting procedure, to compare different cut-combinations in the hierarchies. The measure B_k is derived from the $C \times M$ contingency table with C referring to a hierarchical clustering cut at level i, and M referring to a hierarchical clustering cut at level j. B_k compares the match of assigning pairs of objects to common clusters with the total number of possible pairs, the clustering marginals; B_k is defined as in Equation (4.41). Table 4.7 presents an example of the matching index, based on the contingency table.

$$B_k(C, M) = \frac{T_k}{\sqrt{P_k\ Q_k}} \tag{4.41}$$

where

$$T_k = \sum_{i=1}^{k} \sum_{j=1}^{l} t_{ij}^2 - n \tag{4.42}$$

$$P_k = \sum_{i=1}^{k} t_{i.}^2 - n \tag{4.43}$$

$$Q_k = \sum_{j=1}^{l} t_{.j}^2 - n \tag{4.44}$$

Classification	Clustering	Evaluation
$M_1 = \{a, b, c\}$ $M_2 = \{d, e, f\}$	$C_1 = \{a, b\}$ $C_2 = \{c, d, e\}$ $C_3 = \{f\}$	$T_k = 4 + 1 + 4 + 1 - 6 = 4$ $P_k = 4 + 9 + 1 - 6 = 8$ $Q_k = 9 + 9 - 6 = 12$
		$B_k = \frac{4}{\sqrt{8*12}} = \frac{4}{\sqrt{96}} = 0.40825$

Table 4.7: Example evaluation for matching index

4.2.3 Comparison of Evaluation Measures

Section 4.2.2 has described a variety of possible measures to evaluate the result of a cluster analysis. Following, the different measures are compared against each other and according to the demands of a clustering evaluation, as defined in Section 4.2.1. The comparison is performed in Table 4.8, which lists the evaluation methods against the demands. The demands are briefly repeated:

- Reference to gold standard (given:+ or not given:-)
- Applicable to all similarity measures (yes:+ or no:-)
- Independent of similarity measure (yes:+ or no:-)
- Value for specific cluster and whole clustering (yes:+ or no:-)
- Bias in cluster number (none:-)
- Sensibility to linguistic desiderata (list of failures; none:-), with a brief repetition of the desiderata from Section 4.2.1:

  (a) Clustering result should not be $|C| = 1$.
  (A failure of this desideratum corresponds to a bias towards few large clusters.)

  (b) Clustering result should not be singletons.
  (A failure of this desideratum corresponds to a bias towards many small clusters.)

  (c) Larger correct cluster should be better than smaller correct cluster.

  (d) Correct cluster should be better than same cluster with noise.

  (e) Correct cluster with x objects should be better than noisy cluster with x objects.

  (f) Correct union of correct clusters should be better than separate clusters.

  (g) Correct, separated clusters should be better than incorrect union.

  The success and failure of the desiderata have been evaluated on artificial clustering examples which model the diverse clustering outputs.

- Sensibility to error introduction (monotonic behaviour:+ or not:-)

  This issue refers to an experiment for illustrating the sensibility of the evaluation measures to a step-wise introduction of errors. First the manual classification is evaluated against itself, then I introduce an artificial error and evaluate the result again, etc. The error introduction is repeated 25 times, and an evaluation method sensible to the error introduction should react monotonically in its quality score. Figures 4.3 to 4.5 illustrate the error sensibility of APP, the pair-wise f-score $PairF$ and $Rand_{adj}$.

- Interpretation (minimum and maximum value, if existing, else:-)

The core distinction between the methods is their reference to the gold standard: The sum-of-squared-error and silhouette plot do not refer to the gold standard at all, they measure the quality of the cluster analysis with reference to the data definition and similarity measure. Class-based P/R's underlying idea is very different to any other evaluation method; it compares the distribution of verbs belonging to a common semantic class over the different sets in a partition. Both pair-wise P/R and the adjusted precision measure consider the verb pairs correctly formed by the cluster analysis, with APP incorporating the linguistic desiderata. All other evaluation methods concentrate on the number of verbs agreeing in the gold standard and guessed partitions, as provided by contingency tables; mutual information weights the score by the sizes of the respective sets, the Rand index by the number of possible pairs, and the adjusted Rand index and the matching index take the expected number of agreeing verbs into account.

Table 4.8 illustrates that the different methods have individual strengths and weaknesses. (a) Evaluation measures without general minimum and maximum of the quality scores are more difficult, but possible to interpret. (b) In general, it is better to have quality values for both the specific clusters and the whole clustering, but we can do without the former. (c) Not acceptable for my linguistic needs are evaluation methods which (i) do not refer to the gold standard, because I want to measure how close we come to that, (ii) are dependent on a specific similarity measure, because I want to be able to compare the clustering results based on a range of similarity measures, (iii) have a strong bias towards many small or few large clusters, (iv) fail on a variety of linguistic demands, or (v) do not behave monotonically on error introduction.

To conclude, applicable evaluation methods to my clustering task are the f-score of pair-wise P/R $PairF$, the adjusted pair-wise precision APP, the adjusted Rand index $Rand_{adj}$, and the matching index B_k. Empirically, there is no large differences in the judgement of these methods, so I decided to concentrate on three measures with different aspects on the cluster evaluation: APP as the most linguistic evaluation, $PairF$ which provides an easy to understand percentage (usually the reader is familiar with judging about percentages), and the $Rand_{adj}$ which provides the most appropriate reference to a null model.

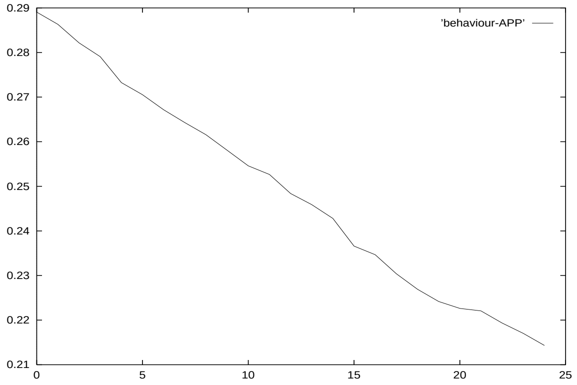

Figure 4.3: APP evaluation on introducing errors

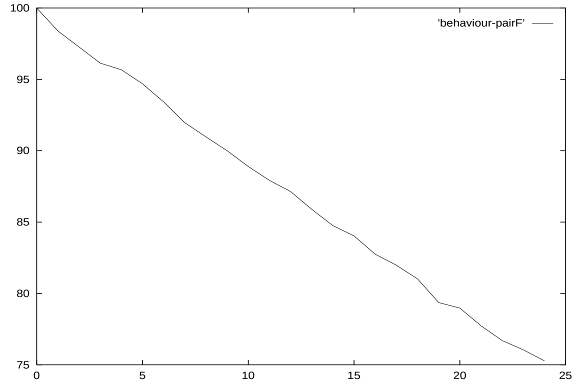

Figure 4.4: $PairF$ evaluation on introducing errors

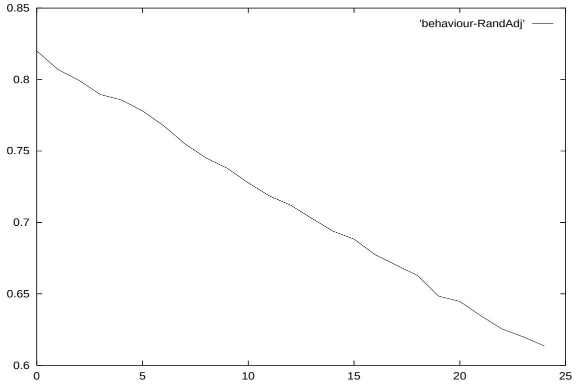

Figure 4.5: $Rand_{adj}$ evaluation on introducing errors

	gold standard	similarity measure		value	
		applicable	independent	specific	whole
Error	-	+	-	+	+
Silhouette	-	+	-	+	+
ClassR	+	+	+	-	+
ClassP	+	+	+	+	+
ClassF	+	+	+	-	+
PairR	+	+	+	-	+
PairP	+	+	+	-	+
PairF	+	+	+	-	+
APP	+	+	+	+	+
MI	+	+	+	+	+
Rand	+	+	+	-	+
Rand$_{adj}$	+	+	+	-	+
B-k	+	+	+	-	+

	bias	linguistics (failure)	error	interpretation	
				min	max
Error	many small	b, c, f	-	-	-
Silhouette	many small	b, f	-	-1	1
ClassR	few large	a, d, g	+	0	100
ClassP	many small	b, f	+	0	100
ClassF	few large	a	+	0	100
PairR	few large	a, d, g	+	0	100
PairP	-	c, f	+	0	100
PairF	-	-	+	0	100
APP	-	-	+	0	-
MI	many small	b	+	0	-
Rand	many small	b, d	+	0	1
Rand$_{adj}$	-	-	+	0	1
B-k	-	-	+	0	1

Table 4.8: Comparison of evaluation measures

4.3 Summary

This chapter has provided an overview of clustering algorithms and evaluation methods which are relevant for the natural language clustering task of clustering verbs into semantic classes. I have introduced the reader into the background of clustering theory and step-wise related the theoretical parameters for a cluster analysis to the linguistic cluster demands:

- The data objects in the clustering experiments are German verbs.
- The clustering purpose is to find a linguistically appropriate semantic classification of the verbs.
- I consider the alternation behaviour a key component for verb classes as defined in Chapter 2. The verbs are described on three levels at the syntax-semantic interface, and the representation of the verbs is realised by vectors which describe the verbs by distributions over their features.
- As a means for comparing the distributional verb vectors, I have presented a range of similarity measures which are commonly used for calculating the similarity of distributional objects.
- I have described a range of clustering techniques and argued for applying the hard clustering technique k-Means to the German verb data. k-Means will be used in the clustering experiments, initialised by random and hierarchically pre-processed cluster input.
- Based on a series of general evaluation demands, general clustering demands and specific linguistic clustering demands, I have presented a variety of evaluation measures from diverse areas. The different measures were compared against each other and according to the demands, and the adjusted pair-wise precision APP, the f-score of pair-wise P/R $PairF$, and the adjusted Rand index $Rand_{adj}$ were determined for evaluating the clustering experiments in the following chapter.