Evaluation of Ontology Enhancement Tools*

Myra Spiliopoulou[1], Markus Schaal[2,**], Roland M. Müller[1], and Marko Brunzel[1]

[1] Otto-von-Guericke-Universität Magdeburg
[2] Bilkent University, Ankara

Abstract. Mining algorithms can enhance the task of ontology establishment but methods are needed to assess the quality of their findings. Ontology establishment is a long-term interactive process, so it is important to evaluate the contribution of a mining tool at an early phase of this process so that only appropriate tools are used in later phases. We propose a method for the evaluation of such tools on their impact on ontology enhancement. We model impact as quality perceived by the expert and as statistical quality computed by an objective function. We further provide a mechanism that juxtaposes the two forms of quality. We have applied our method on an ontology enhancement tool and gained some interesting insights on the interplay between perceived impact and statistical quality.

1 Introduction

The manual establishment of ontologies is an intriguing and resource-consuming task. Efforts are made to enhance this process by unsupervised learning methods. However, as pointed out in [11], the semantic richness and diversity of corpora does not lend itself to full automation, so that the involvement of a domain expert becomes necessary. Hence, unsupervised tools undertake the role of providing useful suggestions, whereupon the quality of their contributions must be evaluated. Since ontology enhancement is a long-term process involving multiple corpora and possibly multiple iterations over the same corpus, this evaluation should be done at an early step, so that only appropriate tools are considered in later steps. In this study, we propose a method for the early evaluation of clustering tools that suggest correlated concepts for ontology enhancement.

Our method has two aspects: First, it evaluates the *impact* of the tool's suggestions as *perceived* by the domain expert. Second, it juxtaposes the *objective quality* of these suggestions to the perceived impact. While the objective quality refers to the statistical properties of the discovered patterns, such as the confidence of a rule or the homogeneity of a cluster, the impact is reflected in the ultimate decision of the expert to include the suggested pattern in the ontology or not. The juxtaposition of the objective, tool-internal notion of quality to the quality perceived by the expert indicates whether the tool and its quality measures will be helpful in further steps of the ontology establishment process.

In the next section, we discuss related work on the evaluation of unsupervised learning tools. In section 3 we describe our method for impact evaluation by the domain expert, followed by the method juxtaposing impact and statistical quality. In section 4, we briefly present the tool we have used as experimentation example. Section 5 describes our experiments and acquired insights. The last section concludes our study.

* Work partially funded under the EU Contract IST-2001-39023 Parmenides.
** Work done while with the Otto-von-Guericke-Universität Magdeburg.

M. Ackermann et al. (Eds.): EWMF/KDO 2005, LNAI 4289, pp. 132–146, 2006.

2 Related Work

Ontology learning tools as proposed in [1,2,4,6,10,8,14] serve different purposes. Many of them propose objects (concepts and relationships) that are found to be supported by a document collection relevant to the application at hand. We concentrate on tools that enhance an existing ontology by proposing (a) new concepts to be inserted in it and (b) relationships among existing concepts.

Usually, an ontology enhancement tool has an inherent quality assessment mechanism that rejects patterns according to some scheme. For tools based on association rules' discovery, quality assessment is often based on interestingness and unexpectedness, while cluster quality is often based on homogeneity or compactness. A rich collection of criteria for the statistical evaluation of unsupervised learners has appeared in [16]. It contains valuable criteria for the assesment of cluster quality, many of them based on indexes of cluster homogeneity. More oriented towards the needs of text clustering are the criteria considered in [15], in which a correlation between some cluster homogeneity indexes and the F-measure is identified when experimenting upon a gold standard. However, application-specific ontology learning cannot rely on gold standards developed for different applications. Moreover, cluster homogeneity does not guarantee or imply that the cluster labels will also be interesting to the domain expert.

Evaluation from the viewpoint of ontology learning is more challenging. Holsapple and Joshi proposed an evaluation method for collaborative manual ontology engineering, in which each suggestion made by one expert is evaluated by at least another expert [7]. Hence, good suggestions are those that enjoy the approval of multiple experts. While this is reasonable for ontology engineering among human experts, it cannot be transferred to non-human experts: Agreement among several ontology learners does not necessarily imply that human experts will find their suggestions useful, since ontology learners are based more on statistics than on background knowledge and expert insight.

The ECAI 2004 workshop on "Ontology Learning and Population" concentrated on the subject of "Evaluation of Text-based Methods in the Semantic Web and Knowledge Discovery Life Cycle" [1]. Faatz and Steinmetz proposed an elegant formalization of "ontology enrichment", followed by a method for automated evaluation on the basis of precision and recall [3], i.e. with respect to gold standards. The selection of those measures is in accordance with the task of evaluation *for algorithmic tuning*: The authors state that "only automatic evaluations of ontology enrichment meet the requirements of algorithmic tuning" and that "the automatization has to be aware of the task specific semantic direction, to which an ontology should evolve" [3]. In our study, we pursue a different goal: We want to assist an expert in deciding on the appropriateness of the tool rather than tune any tool. Moreover, we deliver a procedure that decides whether algorithmic tuning should be made or rather avoided as incompatible to the preferences/intuition of the expert.

Porzel and Malaka consider task-oriented evaluation of ontologies [13]. The process creating an ontology is not specified explicitly, but (semi-)automated processes seem to be permissible; a tool could be evaluated on the quality of the ontology it produces. The authors consider evaluation only with respect to a predefined task, since ontologies

[1] http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.

are indeed built to serve specific tasks. Their evaluation method is based on error-rates, namely superfluous, ambiguous or missing concepts with respect to the task [13]. For our objective of appropriateness evaluation for tools, this approach has some shortcomings. First, it evaluates whole ontologies, while we are interested in the stepwise enhancement of a preliminary ontology. Second, evaluation on the basis of error rates requires a gold standard tailored to the anticipated task. The establishment of such a standard is quite counterintuitive from the viewpoint of a domain expert that needs a tool to enhance an ontology with concepts she does not know in advance.

Kavalec and Svatek propose a method for the evaluation of relation labels, i.e. combinations of terms proposed by a text mining tool to a human expert [9]. According to this method, the expert proposes labels for identified relations and then the tool attempts to re-discover those labels (or synonyms of the terms in them) by mining the text collection. By nature, this approach is appropriate when evaluating a tool on the basis of existing, a priori known relations in a known ontology but less so when evaluating the appropriateness of a tool in expanding an ontology in unknown ways according to the demands of a given expert.

Navigli et al proposed an evaluation method for the OntoLearn system, encompassing a quantitative evaluation towards specific corpora and a qualitative evaluation by multiple domain experts [12]: Quantitative evaluation for the term extraction algorithm, the ontology learning algorithm and the semantic annotation algorithm was performed on predefined corpora which served as gold standards. While this type of evaluation allows for conclusions about the robustness of one tool or the relative performance of multiple tools, it does not allow for generalizations on the usefulness of a given tool to a given expert for the enhancement of a given ontology from a given document collection.

The qualitative evaluation proposed in [12] was based on a questionnaire, in which experts assessed the quality of the definitions of the concepts discovered by OntoLearn: The complex concepts found by the OntoLearn rules were combined with concept definitions from WordNet. The experts were then asked to rate the glosses thus generated as unacceptable, helpful or fully acceptable. This is closer to our one-expert evaluation. However, we do not consider concept definitions, because (a) an appropriate definition provider may or may not be available – the WordNet is not appropriate for specialized domains, (b) the interpretation of a complex concept is left to the expert and (c) a small or medium enterprise intending to enhance an ontology is more likely to dedicate one domain expert to this task rather than 10 or 25 experts. So, the approach is not applicable for *providing assistance* to *one* expert. Further, the appropriateness of the selected corpus was taken for granted; in our approach, this assumption is being put to test.

A method for the generation and evaluation of suggestions towards an ontology user is proposed in [5]. The authors propose a recommendation engine that explores the activities of multiple users, who expand their personal ontologies from a shared basic ontology and suggest metrics for the evaluation of the engine's suggestions. This approach is appropriate when ontologies are built collaboratively by people, since the actions of one person may be helpful to others. However, the metrics do not apply for the actions (suggestions) of one tool towards one domain expert.

3 A Posteriori Impact Evaluation for Ontology Enhancement

Our method evaluates the impact of text miners on the task of ontology enhancement. A text miner processes a text collection and suggests semantics that expand the ontology. These may be terms like "hurricane" or "hurricane warning", term groups that form a new concept or a relation among concepts, e.g. "hurricane warning area", or named relations like "expected within". We refer to them as *"concept constellations"* and focus on the evaluation of the process discovering them. We focus on tools for text clustering and labeling, but our method can be easily extended for association rules' discovery.

3.1 Objectives

We observe ontology enhancement as an iterative process performed by a text mining tool that is applied on an application-specific document collection. The initial input is a preliminary ontology to be enhanced with help of each collection. The final output should be an enriched ontology that is "complete towards the collection", in the sense that the collection cannot contribute new concept constellations to it. More specifically:

- The original input ontology contains a hierarchy or multiple hierarchies of concepts, that may be further connected with horizontal (labeled or unlabeled) relations.
- A text mining tool attempts to enrich the ontology by processing a document collection and identifying semantically correlated concepts. Such correlations are assumed to manifest themselves as concepts that appear frequently together, e.g. as collocates (physically proximal concepts in texts) or as groups of concepts that characterize a cluster of documents. As already noted, we concentrate on text clustering tools. We use the term "concept constellation" for a group of correlated concepts that is returned by the tool as label of a text cluster.
- The correlations among the concepts are used to enrich the ontology. The concepts themselves are already in the ontology, so the enrichment can take two forms, namely the insertion of horizontal relations among the involved concepts and the definition of a new concept that summarizes the correlated concepts.
- The tool finds these concept constellations by mining a document collection.
- The ontolgoy is enriched in many iterations. In each one, the input is the updated ontology. The iterative process ends when no further enrichment can be performed.

Our evaluation method is intended for the first iteration of this process and should answer the following questions:

1. Is the tool appropriate for the enhancement of *this* ontology – on *this collection*?
2. Is the collection appropriate for the enhancement of the ontology – with this tool?
3. Are the tool's quality evaluation functions aligned to the demands of *this* expert?

The motivation of the first question is that a tool may perform well for one collection and poorly for another. A collection can itself be inappropriate for the enhancement of the specific ontology and indeed for opposite reasons: At the one end, the collection may be only marginally relevant, as would be a document collection on outdoor sport

for an ontology on hurricanes. At the other end, the collection may have already served as inspiration for the ontology, whereupon it cannot be used any more to enhance the ontology further.

The last question stresses the subjectivity of the ontology enhancement process. This subjectivity cannot be expelled altogether. However, by modeling the evaluation process on the basis of those three questions, we ensure that the implicit preferences of the expert are partially explicated when dealing with the first two questions. Those preferences that remain tacit are reflected in the outcome of the last question.

We present the evaluation model with respect to the first two questions hereafter and focus on the third question in Section 3.4.

3.2 Perceived Quality as Relevance + Appropriateness

We evaluate the tool's impact on ontology enhancement as *perceived* by the ontology expert. We use two criteria, "relevance to the application domain" and "appropriateness for the ontology O", where D stands for the collection *as representative of the application Domain*. To this purpose, we define two functions $R(D)$ and $A(O, D)$: They are used to measure the relevance of a collection D, resp. its appropriateness for enhancing O within the application domain.

Relevance to the Application Domain. The ontology enhancement is assumed to take place in the context of an application domain and that the collection is representative of that domain. For this criterion, the domain expert is asked to characterize each suggestion (concept constellation) made by the tool as relevant or irrelevant to that domain, *independently of whether she considers the suggestion as appropriate for the ontology*.

The term "relevance" is known to be very subjective. However, the intention of this criterion is not to assess the relevance of the individual suggestions but rather the appropriateness of the tool and of the collection for the application domain. In particular, consider the task of discovering correlated concepts in the following excerpt from the National Hurricane Center at `www.noaa.com`:

```
A HURRICANE OR TROPICAL STORM WARNING MEANS THAT HURRICANE OR
TROPICAL STORM CONDITIONS ... RESPECTIVELY ... ARE EXPECTED
WITHIN THE WARNING AREA WITHIN THE NEXT 24 HOURS. PREPARATIONS
TO PROTECT LIFE AND PROPERTY SHOULD BE RUSHED TO COMPLETION IN
THE HURRICANE WARNING AREA.
```

For the application area of extreme weather warnings, a tool applied on the text collection might suggest the following concepts / constellations, listed here in alphabetical order: (I) "storm, tropical, warning", "area, hurricane, warning", "preparations, protect", (II) "hurricane", "storm", (III) "are, expected", "area". Note that we do not check whether the tool can assess that e.g. "hurricane warning area" is one or two concepts.

- Suggestions of type I are relevant. If most suggestions are of this type, then the tool is appropriate for the collection.
- Suggestions of type III are irrelevant and indicate that the tool cannot find relevant concept constellations upon this collection. If most suggestions are of this type, it should be checked whether the collection itself is appropriate. If yes, then the tool is not appropriate for it.

– Type II suggestions are more challenging. An expert may reject the suggestion "hurricane" as uninformative for an application domain on hurricanes. However, with respect to our criterion, such suggestions should be marked as relevant: Informativeness and appropriateness for the ontology are addressed by our next criterion.

Appropriateness for the Ontology. The Appropriateness criterion $A(O, D)$ refers to the expansion of ontology O for the application domain D. It builds upon the relevance criterion $R(D)$: only relevant concept constellations are considered. For a relevant concept constellation $Y = Y_1, \ldots, Y_m$, the following cases may occur:

– Y is already in the ontology. Then it should be rejected as inappropriate.
– Y contains some concepts that are appropriate for the ontology, either as individual concepts or as a group. Then Y should be accepted; each appropriate concept/group should be named.
– Y contains no concept that is appropriate for the ontology. It should be rejected.

According to this scheme, a concept constellation may contribute one or more concepts to the ontology. Hence, $A(O, D)$ delivers two lists of results: $A(O, D) = \{S, S_+\}$, where $S \subseteq R(D)$ is the set of accepted concept constellations and S_+ is the set of concept groups appropriate for the ontology.

We use the result $A(O, D).S$ to assess the appropriateness of the tool for further iterations in the ontology enhancement process. The result $A(O, D).S_+$ is used in 3.4, where we juxtapose the quality criteria of the tool to the impact perceived by the expert.

3.3 Combining Relevance and Appropriateness Ratings

Let $T(D)$ be the set of concept constellations suggested by the tool T for the application domain. We combine the results on relevance $R(D) \subseteq T(D)$ and appropriateness for the ontology $A(O, D).S$ to figure out whether the tool T should be further used for the enhancement of the ontology on domain D, whereupon we consider the collection already analyzed as representative for domain D. The following cases may occur:

– The ratio $\frac{|R(D)|}{|T(D)|}$ is close to zero.
Then, the tool is not appropriate for this collection and thus for the domain.
– The ratio $\frac{|R(D)|}{|T(D)|}$ is closer to one and the ratio $\frac{|A(O,D).S|}{|R(D)|}$ is close to zero.
Then, the tool is capable of analyzing documents in the application domain but the collection does not deliver informative concept constellations for the ontology. This may be due to the tool or to the relationship between ontology and collection. To exclude the latter case, the domain expert should again verify the appropriateness of this collection *for ontology enhancement*: If all concepts in the collection are already in the ontology, the collection is still relevant but cannot enrich the ontology any more. Hence, the tool should be tested upon another representative collection.
– Both ratios are closer to one than to zero.
Then, the tool is able to contribute to ontology enhancement for this collection and is thus appropriate for the application domain.

By this procedure, we can assess whether a given tool should be further used for the gradual enhancement of the ontology. For a more *effective* ontology enhancement process, it is also reasonable to know to which extent the tool's suggestions can be trusted without close inspection. This would be the case if the enhancements proposed by the tool fit to the expectations of the human expert (cf. Question 3 in Section 3.1). To this purpose, we juxtapose the evaluation by the expert to the internal quality evaluation by the tool. Obviously, this juxtaposition is only possible for tools that disclose the values assigned to their suggestions by their internal evaluation criteria. For tools delivering only unranked suggestions, no juxtaposition is possible.

3.4 Juxtaposition of Statistical and Perceived Quality

Each (text clustering) tool has some internal or external criterion for the rejection of potentially poor patterns and the maintenance, respectively further exploitation, of good patterns. The results of any clustering algorithm encompass both good and less good clusters, whereby goodness is often measured in terms of compactness, homogeneity, informativeness etc [15,16]. We name such criteria "statistical quality criteria".

Towards our objective of ontology enhancement, we say that a statistical quality criterion $SQ()$ *"is aligned to the perceived quality"* when the likelihood that the domain expert considers a concept group as appropriate for the ontology increases (resp. decreases) with the statistical quality of the cluster with respect to that criterion.

As basis for the statistical quality, let $SQ()$ be a statistical quality criterion that assigns to each cluster generated by T a value. Without loss of generality, we assume that the range of these values is $[0, 1]$ and that 1 is the best value. As basis for the perceived quality, we consider the concept groups characterized by the domain expert as appropriate for the ontology, i.e. the set $A(O, D).S_+$ defined in 3.2.

Associating Concept Groups and Constellations with Clusters. To compare the perceived with the statistical quality of the concept groups and constellations, we compute the distribution of statistical quality values for the concept groups accepted by the expert and for the concept constellations rejected by her.

Since an accepted concept group, i.e. an element of $A(O, D).S_+$ may appear in more than one concept constellations, it can be supported by one or more clusters of the clustering $T(D)$ generated by the tool and these clusters may be of different statistical quality. Hence, we associate each concept group $x \in A(O, D).S_+$ to the best among the clusters supporting it, C_x and then to the quality value of this cluster $SQ(C_x)$. We denote the set of pairs $(x, SQ(C_x))$ thus computed as $expertApproved$.

Similarly, we associate each rejected concept constellation $x \in T(D) \setminus A(O, D).S$ to the cluster C_x from which it was derived. Differently from the concept groups which may be supported by several clusters, a concept constellation corresponds to exactly one cluster, so the assignment is trivial. We denote the set of pairs $(x, SQ(C_x))$ thus computed as $expertRejected$.

In Table 1, we show the algorithm that computes the two sets $expertApproved$ and $expertRejected$. For each concept group $x \in A(O, D).S_+$ all clusters that deliver concept constellations containing x. It selects among them the cluster with the highest quality value according to $SQ()$ and associates x to this $maxSQ(x)$ (lines 3-7). The filling of the two sets of value pairs in the lines 8, 11 is straightforward.

Table 1. Associating each concept group to the best quality cluster

```
1  expertApproved:=expertRejected=∅
2  For each concept group x in A(O,D).S+
3     maxSQ := 0
4     For each cluster C in T(D) that supports x
5        if maxSQ less than SQ(C)
6           then maxSQ := SQ(C)
7     Endfor
8     expertApproved:=expertApproved∪{(x,maxSQ)}
9  Endfor
10 For each concept constellation x in T(D)\A(O,D).S
11    expertRejected:=expertRejected∪{(x,SQ(C_x))}
12 Endfor
```

Comparing Distributions. The two sets $expertApproved$ and $expertRejected$ can be mapped into distributions of statistical quality values for the accepted, resp. rejected clusters. We denote these distributions as dA and dR respectively. To check whether statistical quality and perceived quality are aligned, we should compare those distributions. However, since the concrete distributions are not known, we can either (a) derive histograms hA and hR for them by partitioning the valuerange of $SQ()$ into k intervals for some k or (b) compute the mean and standard deviation of each dataset. Then, the form of the histograms or the values of the means are compared. For the comparison of histograms, we consider the cases depicted in Table 2.

By this juxtaposition we can assess whether a statistical quality criterion used by the tool is aligned to the implicit perceived quality function of the domain expert. If some criteria are aligned, they should take priority over misaligned ones in subsequent ontology enhancement steps. Even if all criteria are misaligned, the tool can still be

Table 2. Comparison of histograms - four cases

1. Both histograms are unimodal, hA is shifted towards the best quality value for $SQ()$, while hR is shifted towards the worst value.
   This is the best case: The likelihood that a cluster contributes to ontology enhancement increases with its quality and vice versa. $SQ()$ is aligned to perceived quality.
2. Both histograms are unimodal, hR is shifted towards the best value and hA is shifted towards the worst value.
   This is the second best case. The statistical quality criterion is *consistently* counterproductive. One might reasonably argue that this $SQ()$ is a poor criterion, but it is also true that $1-SQ()$ is aligned to the perceived quality and is thus very useful.
3. The two histograms have the same shape and are shifted in the same direction.
   Then the likelihood of having a good cluster accepted or rejected by the expert is the same as for a bad cluster. Thus, $SQ()$ is misaligned to the perceived quality.
4. No pattern can be recognized. Then $SQ()$ is misaligned to the perceived quality.

used. However, it should then deliver to the domain expert the poor quality clusters as well, since she may find useful information in them.

A comparison based on histograms depends on the selected number of intervals k and on the specific partitioning of the valuerange of $SQ()$. An alternative, simpler approach would be to compute the proximity of the median to the best, resp. worst value of $SQ()$: Similarly to the comparison of the histograms, if the median of $expertApproved$ is close to the best value of $SQ()$ and the median of $expertRejected$ is close to the worst value, then $SQ()$ is aligned; if the reverse is the case, then $SQ()$ is consistently counterproductive. Otherwise, $SQ()$ is misaligned.

4 An Example Tool and Its Quality Evaluation Criteria

As a proof of concept, we have applied our evaluation method upon the tool "RELFIN Learner" [14]. We describe RELFIN and its internal quality evaluation criteria below, mostly based on [14]. We stress that RELFIN is only an example: Our method can be applied on arbitrary tools that suggest concepts for ontology enhancement. Obviously, the juxtaposition to a tool's statistical quality is only feasible if the tool reports its quality assessment values as required in 3.4.

RELFIN is a text clustering algorithm using Bisecting-K-means as its clustering core and a mechanism for cluster evaluation and labeling. RELFIN discovers new concepts as single terms or groups of terms characterizing a cluster of text units. These concepts, resp. concept constellations can be used to expand the existing ontology, to semantically tag the corresponding text units in the documents or to do both. RELFIN can take as input both concepts from an initial, rudimentary ontology and with additional terms it extracts automatically from the collection. Accordingly, its suggestions are new concepts consisting of terms in the collection and constellations consisting of terms from either the ontology or the collection. The labels / concept constellations suggested by RELFIN should be appropriate as semantic markup on the text fragments. This is reflected in the quality criteria of RELFIN.

4.1 Definitions

A *text unit* is an arbitrary text fragment extracted by a linguistic tool, e.g. by a sentence-splitter; it is usually a paragraph or a sentence. Text units are composed of terms. For our purposes, a *text collection* $\mathcal{A}$ is a set of text units.

A term is a textual representation of a *concept*. A *feature space* $\mathcal{F}$ consists of concepts from the existing ontology, terms extracted from the collection by some statistical method or both. We assume a feature space with d dimensions and a *vectorization* $\mathcal{X}$ in which each text unit i is represented as vector of TFxIDF weights $x_i = (x_{i1}, \ldots, x_{id})$. Obviously, concepts of the ontology that do not appear in the collection are ignored.

Given is a *clustering scheme* or *clusterer* $\mathcal{C}$. For a cluster $C \in \mathcal{C}$, we compute the in-cluster-support of each feature $f \in \mathcal{F}$ as

$$ics(f, C) = \frac{|\{x \in C | x_f \neq 0\}|}{|C|} \quad (1)$$

Definition 1 (Cluster Label). *Let $C \in \mathcal{C}$ be a cluster over the text collection $\mathcal{A}$ for the feature space $\mathcal{F}$. The* label *of C $label(C)$ is the set of features $\{f \in \mathcal{F} | ics(f, C) \geq \tau_{ics}\}$ for some threshold τ_{ics}.*

A feature satisfying the threshold constraint for a cluster C is a *frequent feature* for C.

4.2 Quality Measures

A label might be specified for any cluster. To restrict labeling to good clusters only, we use one criterion on cluster compactness and one on feature support inside clusters.

Definition 2 (Average distance from centroid). *Let $C \in \mathcal{C}$ be a cluster over the text collection $\mathcal{A}$ for the feature space $\mathcal{F}$ and let $d()$ be the distance function for cluster separation. The average intra-cluster distance from the centroid is defined as $avgc(C) = \frac{\sum_{x \in C} d(x, centroid(C))}{|C|}$, whereupon lower values are better.*

Definition 3 (Residue). *Let $C \in \mathcal{C}$ be a cluster and let τ_{ics} be the in-cluster support threshold for the cluster label. Then, the "residue" of C is the relative in-cluster support for infrequent features:*

$$residue(C, \tau_{ics}) = \frac{\sum_{f \in \mathcal{F} \setminus label(C)} ics(f, C)}{\sum_{f \in \mathcal{F}} ics(f, C)} \tag{2}$$

The residue criterion serves the goal of using cluster labels for semantic markup. Consider text units that support the features X and Y and text units that support Y and Z. If the algorithm assigns them to the same cluster, then both pairs of features can be frequent, depending on the threshold τ_{ics}. A concept group "X,Y,Z" may well be of interest for ontology enhancement, but it is less appropriate as semantic tag. We allow for low τ_{ics} values, so that such constellations can be generated. At the same time, the residue criterion favours clusters dominated by a few frequent features shared by most cluster members, while all other features are very rare (values close to zero are best).

5 Experiments

We performed an experiment on ontology enhancement involving a domain expert who used the RELFIN Learner for the enhancement of an existing ontology. The expert's goal was to assess usability of the tool. The complete usability test is beyond the scope of this study, so we concentrate only on the impact assessment criteria used in the test. The juxtaposition to the statistical criteria of the tool was not part of the usability test.

5.1 The Case Study for Ontology Enhancement

Our method expects a well-defined application domain. This was guaranteed by a predefined case study with a given initial ontology on biotechnology watch and two domain-relevant collections of business news documents. We used a subcollection of BZWire news (from 1.1.2004 to 15.3.2004), containing 1554 documents. The vectorization process resulted in 11,136 text fragments.

The feature space consisted of 70 concepts from the initial ontology and 230 terms extracted from the collection. These terms were derived automatically as being more frequent for the collection than for a reference general purpose corpus. The target number of clusters was set to 60 and the in-cluster-support threshold for cluster labeling τ_{ics} was set to 0.2. Setting τ_{ics} to such a rather low value has turned to be helpful for our observations, because high values would reduce the set of suggestions considerably.

5.2 Evaluation on Relevance and Appropriateness

RELFIN delivered 60 clusters of varying quality according to the tool's internal criteria. For the impact assessment by the domain expert, though, these criteria were switched off, so that all cluster labels subject to $\tau_{ics} = 0.2$ were shown to the domain expert. This implies that RELFIN suggested the labels of all 60 clusters, so that $|T(D)| = 60$.

The domain expert was asked to assess the relevance of each cluster label, i.e. constellation of frequent features. A label was relevant if it contained at least one relevant feature. The appropriateness of the features in relevant cluster labels was assessed next: The domain expert was asked whether NONE, ALL or SOME of the concepts in the relevant label were also appropriate. The answers were:

- *Relevance to the case study:* YES: 43, NO: 17 $|R(D)| = 43$
- *Appropriateness for the ontology:* NONE: 2, ALL: 4, SOME: 37 $|A(O, D).S| = 41$

We combined these values as described in 3.3. To compute $A(O, D).S_+$, we enumerated the concept groups in the labels characterized as SOME, using the following rules:

1. The expert saw a label with several concepts and named n concept groups that he considered appropriate for the ontology. Then, we counted n appropriate objects.
2. The expert found an appropriate concept or concept group and marked it in *all* cluster labels containing it. Then, we counted the appropriate object only once.
3. The domain expert saw a label "A,B,C,…", and wrote that "A,B" should be added to the ontology. Then, we counted one appropriate object only, even if the terms "A" and "B" did not belong to the ontology.
4. The expert saw a label of many concepts and marked them "ALL" as appropriate. This case occured 4 times. For three labels, we counted one appropriate object only, independently of the number of new concepts and possible combinations among them. For the 4th label, we counted two appropriate objects: the label as a whole and one specific term X. X belongs to a well-defined set of terms and the expert had encountered and accepted three further members of this set when evaluating other clusters. So we added this term, too.

In Table 3 we show the relevance and appropriateness ratios according to those rules. These ratios allow for an assessment (positive in this case) of the tool's appropriateness for further iterations. In the last rows, we have computed the average number of appropriate concept groups, as contributed by the RELFIN clusters. The last ratio is peculiar to RELFIN, which can exploit both concepts from the ontology and terms from the collection. The ratio says that 87% of the approved concept groups were not in the ontology. The remaining 23% are combinations of concepts from the ontology.

Table 3. Relevance and appropriateness ratios

Tool suggestions	$\lvert T(D)\rvert$	60
Relevance ratio	$\frac{\lvert R(D)\rvert}{\lvert T(D)\rvert}$	$43/60 \approx 0.72$
Appropriateness ratio	$\frac{\lvert A(O,D).S\rvert}{\lvert R(D)\rvert}$	$41/43 \approx 0.95$
Avg contribution of concept groups per relevant cluster		$62/43 \approx 1.44$
Avg contribution of concept groups per cluster		$62/60 \approx 1.03$
Contribution of the collection to the ontology		$54/62 \approx 0.87$

5.3 Impact Versus Statistical Quality

For the juxtaposition of the impact evaluation with the statistical quality criteria of RELFIN, we used the approach described in 3.4. Both criteria used by RELFIN range in the interval $[0, 1]$; 1 is the worst value and 0 is the best one. We have adjusted the generic procedure accordingly for the experiment.

In Table 4 we show the histograms for RELFIN. We have set the number of intervals to $k = 10$. However, we have noticed that all values of relevance according to Table 2 were in the intervals between 0.3 and 0.5 for the criterion "average distance from the centroid" $avgc$ and between 0.2 and 0.6 for the criterion "residue". Therefore, we have summarized the corresponding $SQ()$ values for the first two intervals into $[0, 0.2)$ and for the last intervals into $[0.5, 1)$ for the $avgc$ and into $[0.6, 1)$ for the residue.

Table 4. Quality values for approved vs rejected clusters

	Avg Distance to centroid					
	[0,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)	[0.5,1]	
Approved concept groups	2	7	19	27	6	
expertApproved clusters	2	5	12	17	7	
expertRejected clusters	1	1	1	4	10	
	Residue					
	[0,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)	[0.5,0.6)	[0.6,1]
Approved concept groups	0	2	6	16	12	25
expertApproved clusters	0	2	4	9	9	19
expertRejected clusters	1	3	4	0	3	6

For each criterion, the first row shows the distribution of cluster quality values for the approved concept groups. As pointed out in Section 3.4, a concept group may be supported by more than one clusters, from which the one with the highest quality is chosen (cf. Table 1). The second row shows the cluster quality values per interval for the approved clusters, i.e. for the set $expertApproved$. The third row shows the corresponding distribution for the clusters in $expertRejected$.

For the criterion $avgc()$, most values of hA (clusters in $expertApproved$) are in $[0.3, 0.5)$; a steep decrease occurs afterwards. For the hR (clusters in $expertRejected$),

most values are in $[0.5, 1)$. The median of $expertApproved$ is in the interval $[0.4, 0.5)$, the median of $expertRejected$ is larger than 0.5. These observations are indicatory of the first case in Table 2, hence the "average distance to the centroid" $avgc()$ is aligned to the expert's evaluation.

In the first row of the criterion "residue", we can see a modus in the interval $[0.4, 0.5)$. It is followed by a smaller modus in the next interval $[0.5, 0.6)$, which also contains the median. It must be stressed that the last interval is an aggregate; there is no modus there. The value distribution in the hA for the $expertApproved$ clusters is in the second row: The modus spans over the two intervals $[0.4, 0.5)$ and $[0.5, 0.6)$; the latter contains the median. However, the histogram of $expertRejected$ clusters has at least two modi, one before the interval $[0.4, 0.5)$ and at least one afterwards; this interval is itself empty. Hence, the likelihood of a cluster rejection is high both before and after this interval. So, we conclude that the criterion is misaligned.

One explanation of the misalignment of the residue is that the labels of clusters with higher residue contain more concepts. When the human expert identified appropriate concept groups for the ontology, he had more candidates to choose from. Those concept groups are not appropriate as semantic tags but this does not affect their appropriateness for the ontology. We consider this as indicatory for impact assessment: If a concept (group) appeals to the domain expert, i.e. is informative with respect to her background knowledge, she will approve it independently of its statistical support.

6 Conclusions

We have proposed a method that evaluates the appropriateness of text clustering tools for ontology enhancement on the basis of their suggestions to the domain expert. Our approach is intended as an instrument to help the domain expert decide at the beginning of the ontology enhancement process whether the tool is appropriate for further steps of this process. To this purpose, we combine subjective impact assessment with a more objective relevance test and we finally check whether the statistical evaluation instruments used by the tool are aligned to the subjective preferences of the expert. We have performed a first test of our method for a text clustering tool on the enhancement of the ontology of a real case study and we gained some rather interesting insights on the interplay of statistical "goodness" and subjective "appropriateness".

The juxtaposition of statistical quality and impact assessment might be observed as a classification task, where statistical criteria serve as predictors of impact. We intend to investigate this potential. We further plan to enhance the impact assessment with more elaborate criteria. Moreover, we want to evaluate further tools with our method: This implies conducting an experiment in which the expert works with multiple tools on the same corpus and the same basic ontology.

Acknowledgement. We would like to thank the domain expert Dr. Andreas Persidis of the company BIOVISTA for the impact evaluation and for many insightful comments on the expectations towards interactive tools used in ontology enhancement.

References

1. Philipp Cimiano, Steffen Staab, and Julien Tane. Automatic acquisition of taxonomies from text: Fca meets nlp. In *Proc. of the ECML/PKDD Workshop on Adaptive Text Extraction and Mining*, pages 10–17, Cavtat, Croatia, Sept. 2003.
2. Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. SemTag and seeker: Bootstrapping the semantic web via automated semantic annotation. In *Proc. of the 12th Int. World Wide Web Conf.*, pages 178–186, Budapest, Hungary, 2003. ACM Press.
3. Andreas Faatz and Ralf Steinmetz. Precision and recall for ontology enrichment. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
4. David Faure and Claire Nédellec. Knowledge acquisition of predicate argument structures from technical texts using machine learning: The system ASIUM. In Dieter Fensel and Rudi Studer, editors, *Proc. of 11th European Workshop on Knowledge Acquisition, Modeling and Management (EKAW'99)*, volume LNAI 1621, pages 329–334, Dagstuhl, Germany, May 1999. Springer-Verlag, Heidelberg.
5. Peter Haase, Andreas Hotho, Lars Schmidt-Thieme, and York Sure. Collaborative and usage-driven evolution of personal ontologies. In *Proc. of European Conference on the Semantic Web (ESWC 2005)*, LNCS 3532, pages 486–499. Springer Verlag Berlin Heidelberg, May/June 2005.
6. Siegfried Handschuh, Steffen Staab, and F. Ciravegna. S-CREAM – Semi-automatic CREation of metadata. In *Proc. of the European Conf. on Knowledge Acquisition and Management*, 2002.
7. Clyde Holsapple and K.D. Joshi. A collaborative approach to ontology design. *Communications of ACM*, 45(2):42–47, 2005.
8. Andreas Hotho, Steffen Staab, and Gerd Stumme. Explaining text clustering results using semantic structures. In *Proc. of ECML/PKDD 2003*, LNAI 2838, pages 217–228, Cavtat-Dubrovnik, Croatia, Sept. 2003. Springer Verlag.
9. M. Kavalec and V. Svatek. A study on automated relation labelling in ontology learning. In P. Buitelaar, P. Cimiano, and B. Magnini, editors, *Ontology Learning and Population*. IOS Press, 2005.
10. Jianming Li, Zhang Lei, and Yong Yu. Learning to generate semantic annotation for domain specific sentences. In *Proc. of the "Knowledge Markup and Semantic Annotation" Workshop of the K-CAP 2001 Conference*, 2001.
11. Alexander Maedche and Steffen Staab. Semi-automatic engineering of ontologies from text. In *Proc. of 12th Int. Conf. on Software and Knowledge Engineering*, Chicago, IL, 2000.
12. Roberto Navigli, Paola Velardi, Alessandro Cucchiarelli, and Francesca Neri. Quantitative and qualitative evaluation of the ontolearn ontology learning system. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
13. Robert Porzel and Rainer Malaka. A task-based approach for ontology evaluation. In *Proc. of ECAI-2004 Workshop on Ontology Learning and Population*, Valencia, Spain, Aug. 2004. http://olp.dfki.de/ecai04/cfp.htm, accessed at July 26, 2005.
14. Markus Schaal, Roland Mueller, Marko Brunzel, and Myra Spiliopoulou. RELFIN - topic discovery for ontology enhancement and annotation. In *Proc. of European Conference on the Semantic Web (ESWC 2005)*, LNCS 3532, pages 608–622, Heraklion, Greece, May/June 2005. Springer Verlag Berlin Heidelberg.

15. Benno Stein, Sven Meyer zu Eissen, and Frank Wißbrock. On Cluster Validity and the Information Need of Users. In M.H. Hanza, editor, *3rd IASTED Int. Conference on Artificial Intelligence and Applications (AIA03)*, pages 216–221, Benalmadena, Spain, September 2003. ACTA Press.
16. Michalis Vazirgiannis, Maria Halkidi, and Dimitrios Gunopoulos. *Uncertainty Handling and Quality Assessment in Data Mining*. Springer, 2003.