From Anomaly Reports to Cases

Stewart Massie[1], Nirmalie Wiratunga[1], Susan Craw[1], Alessandro Donati[2], and Emmanuel Vicari[2]

[1] School of Computing
The Robert Gordon University
Aberdeen AB25 1HG, Scotland, UK
{nw,sm,smc}@comp.rgu.ac.uk
[2] European Space Agency
European Space Operations Centre
64293 Darmstadt, Germany

Abstract. Creating case representations in unsupervised textual case-based reasoning applications is a challenging task because class knowledge is not available to aid selection of discriminatory features or to evaluate alternative system design configurations. Representation is considered as part of the development of a tool, called CAM, which supports an anomaly report processing task for the European Space Agency. Novel feature selection/extraction techniques are created which consider word co-occurrence patterns to calculate similarity between words. These are used together with existing techniques to create 5 different case representations. A new evaluation technique is introduced to compare these representations empirically, without the need for expensive, domain expert analysis. Alignment between the problem and solution space is measured at a local level and profiles of these local alignments used to evaluate the *competence* of the system design.

1 Introduction

In this paper we review the development of a case-based reasoning (CBR) application applied to the complex task of anomaly report matching for the European Space Agency (ESA). The cases are presented as semi-structured textual documents consisting, largely, of several sections of text describing the problem and one section of text describing the solution. In particular, we focus on the problem of deriving a structured case representation from unsupervised text data using feature selection and extraction techniques and on evaluating alternative design configurations.

Case representation is a key design issue for the successful development of any CBR system. This is particularly true for a Textual CBR (TCBR) system which generally requires the application of feature selection or extraction techniques to reduce the dimensionality of the problem by removing non-discriminatory and sometimes detrimental features. Dimensionality reduction has been shown to be successful in improving accuracy and efficiency for supervised tasks in unstructured domains [23]. However, in an unsupervised setting feature selection/extraction is a far more challenging task because class knowledge is not available to evaluate alternative representations.

We compare a TFIDF feature selection approach with a novel technique in which similarity between words is calculated by analysing word co-occurrence patterns

R.O. Weber and M.M. Richter (Eds.): ICCBR 2007, LNAI 4626, pp. 359–373, 2007.
© Springer-Verlag Berlin Heidelberg 2007

followed by seed word selection using a footprint-based feature selection method. Applying feature selection only can result in sparse representations so we investigate feature extraction techniques using rules induced by either Apriori or from feature similarity neighbourhoods to generalise the seed words and reduce sparseness. The techniques are implemented in a prototype CBR Anomaly Matching demonstrator, called CAM, which retrieves similar reports when presented with a new anomaly and incorporates intuitive visualisation techniques to convey case similarity knowledge.

Evaluation in unsupervised TCBR systems also presents difficulties because the typical approach, which involves a domain expert rating a small number of retrievals, is very time consuming and depends on the availability of a willing domain expert. Evaluation is especially troublesome when following a typical incremental development approach in which a series of small changes are made to the design with evaluation required to measure the effect of the change after each stage. We introduce a novel approach to evaluation that measures the extent to which *similar problems have similar solutions* by investigating the alignment between local neighbourhoods in the problem and solution space. This approach reduces the requirement for human evaluations.

The problem domain is described in more detail in Section 2 along with CAM's key objectives. Section 3 discusses several feature selection and extraction techniques used to create alternative case representations. We describe how the prototype was implemented in Section 4. Evaluation results comparing five alternative system designs by measuring the alignment between problem and solution space are presented in Section 5. Related work is discussed in Section 6 before we provide conclusions and recommendations for future work in Section 7.

2 Anomaly Reporting

ESA is Europe's gateway to space. Its mission is to shape the development of Europe's space capability and ensure that investment in space continues to deliver benefits to the citizens of Europe. ESOC, the European Space Operations Centre, is responsible for controlling ESA satellites in orbit and is situated in Darmstadt, Germany. ESOC works in an environment in which safety and Quality Assurance is of critical importance and, as a result, a formal Problem Management process is required to identify and manage problems that occur both within the operations of the space segment and of the ground segment. Observed incidents and problems (the cause of the incidents), are recorded by completing anomaly reports.

Anomaly reports are semi-structured documents containing both structured and unstructured data. There are 27 predefined structured fields containing information such as: the originator's name; key dates relating to the report and the physical location of the anomaly. Structured fields are used to group and sort reports, for example by urgency or criticality. Importantly, for knowledge reuse purposes the anomaly reports also have four text sections: observation (the title of the report), description (facts observed), recommendation (first suggestion on recovery), and resolution (how the problem is analysed and disposed). These four unstructured sections contain free text that are not necessarily always spell-checked or grammatically perfect but contain valuable knowledge.

The work described in this paper involves the organisation and extraction of knowledge from anomaly reports maintained by the ARTS system. The overall goal is to extract knowledge and enable decision support by reusing past-experiences captured in these reports. An initial prototype CAM supports report linking and resolution retrieval.

- *Task 1:* Report linking aims to discover similar technical problems across multiple reports and to generate links between reports across projects. Reports can be related because they either describe symptoms of the same problem within the same project (indicating the re-occurrence of incidents associated with the same cause) or they report a similar anomaly shifted in time occurring in different projects / missions. Relating anomalies can highlight single problems that result in multiple incidents which are recorded in different (and sometimes un-related) reports. One goal is to find relationships in an automatic way.
- *Task 2:* Report reuse aims to retrieve similar reports so that their resolution can be re-applied to the current problem. This involves retrieval and reuse of anomaly reports with the requirement to compare new anomaly descriptions with past anomaly reports. In standard CBR terminology the resolution section provides the problem solution while the remaining sections decompose the problem description. Determining a suitable resolution for an anomaly is currently a manual decision making process (using Anomaly Review Boards) requiring considerable domain expertise. The prototype aids this decision-making process by providing the user with a list of anomaly reports that have similar problem descriptions to the current anomaly.

3 Report Representation

The first task for developing our prototype CBR system was to create a case representation for anomaly reports. The structured fields in the document were reduced to 13 relevant features following discussions with the domain experts.

Representation of the textual parts of the reports is a far harder task. The unstructured text has to be translated into a more structured representation of feature-value pairs. This involves identifying relevant features that belong to the problem space and solution space. The translation from text into a structured case representation can not be performed manually because the dimensionality of the problem is too great: there is a large vocabulary in the training sample of 960 reports which forms just 20% of the ESA's report database. An approach which can identify relevant features from the corpus is required. There are numerous approaches to feature selection and extraction on supervised problems where class knowledge can be used to guide the selection [10,23]. However since we are faced with an unsupervised problem the selection needs to be guided by knowledge other than class.

Our approach to unsupervised feature extraction (Figure 1) consists of three stages: an initial vocabulary reduction by pre-processing text using standard IR and NLP techniques; next seed word selection using word frequency counts or word distribution profiling; and finally feature extraction by considering word co-occurrence to avoid sparse representations using Apriori rules or seed word similarity neighbourhoods.

3.1 Text Pre-processing

The initial vocabulary is reduced to 2500 words by applying the following document pre-processing techniques:

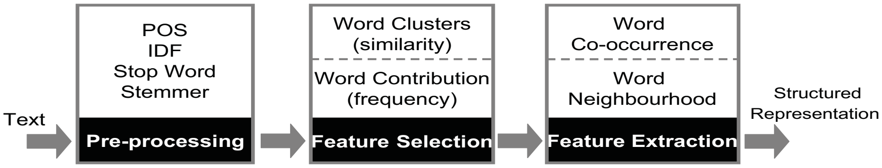


Fig. 1. Processing unstructured text to create structured representation

- Part of Speech Removal: text is first tokenised to identify word entities then tagged by its part of speech. Only nouns and verbs are retained.
- Stop Word Removal and Stemming: removes commonly occurring words and reduces remaining words to their stem by removing different endings, e.g., both anomaly and anomalous are stemmed to their root anomaly.
- Frequency Based Pruning: reduces the vocabulary, from approximately 8000 words to 2500 words, by considering the inverse document frequency (idf) of each word to determine how common the word is in all of the documents. We accept words that are common across several documents but not too frequent by accepting words with an idf value of between 3 and 6.

3.2 Feature Selection

Feature selection for structured data can be categorised into filter and wrapper methods. Filters are seen as data pre-processors and generally, unlike wrapper approaches, do not require feedback from the final learner. They tend to be faster, scaling better to large datasets with thousands of dimensions, as typically encountered in text applications.

Unlike with supervised methods, comparative studies into unsupervised feature selection are very rare. One of the few approaches explicitly dealing with unsupervised feature selection for text data [10] relies on heuristics that are informed by word frequency counts over the text collection. We compare this word contribution method with a novel similarity clustering approach that can consider contextual information.

Seed Word Selection by Word Contribution

Word frequency information can be used to gauge a word's contribution towards similarity computation for case comparison. Ideally we wish to ignore words that distribute over the entire case base whilst preferring those that are discriminatory of similar reports. TFIDF is commonly used in IR research to measure the discriminatory power of a word for a given document. The unsupervised feature selection approach introduced in [10] uses these TFIDF values to arrive at a feature ranking score. For a given word all its TFIDF values are combined using the vector product so that a word that is consistently discriminatory of small subsets of cases are preferred over those words that are discriminatory of only individual cases.

Seed Word Selection by Similarity Clustering

Seed words should be representative of areas of the problem space but also diverse so that together they provide good coverage of the problem space. Knowledge about word similarity enables the search process to address both these requirements. The question then is how do we define similarity between words and thereafter how do we select representative but diverse words.

- **Word Similarity**
  One approach is to consider the number of times words co-occur in documents [14], however, a problem is that similar words do not necessarily co-occur in any document, due to sparsity and synonymy, and will not be identified as similar.

  Our approach is to analyse word co-occurrence patterns with the set of words contained in the solution, i.e., the remaining words from the resolution field. For example, to calculate the similarity between words in the observation field of the anomaly report, the conditional probability of co-occurrence is first calculated between each word in the observation field with each word in the resolution field. A distribution of these probabilities is then created for each observation word. A comparison between these distributions can then be made using the α-Skew metric derived from information theory [8]. This comparison provides an asymmetric similarity estimate between words in the observation field. We repeat the same process for all the text fields. Essentially similar words are those that have similar co-occurrence patterns with resolution words. A full description of this word similarity approach is given in [21].
- **Representative but Diverse Selection**
  We use the similarity knowledge derived from the conditional probability distributions to aid the search for a representative but diverse set of seed words. These words form the dimensions for the case representation. Smyth & McKenna developed a footprint-based retrieval technique in which a subset of the case base, called footprints, is identified to aid case retrieval [16]. We use a similar technique to cluster words and then select representative seed words from word clusters.

  Word clusters are created by first forming coverage and reachability sets for each word. In our scenario, the coverage set of a word contains all words within a predefined similarity threshold. Conversely, the reachability set of a word is the set of words that contains this word in its coverage set. Clusters of words are then formed using the reachability and coverage sets to group words that have overlapping sets. In Figure 2, six words (w_1 to w_6) are shown spaced in relation to their similarity to each other. The coverage of each word is shown by a circle with a radius corresponding to the similarity threshold. It can be seen that two clusters are formed: w_1 to w_5 in one cluster and w_6 in the other. A representative set of seed words is selected for each cluster by first ranking the words in descending order of relative coverage [16]. Each word is then considered in turn and only selected if it is not covered by another already selected word. The words are shown in Figure 2, in ranked order, with their coverage sets and related coverage scores. Hence w_1, w_5 and w_6 will be selected as the seed words. The composition of the coverage sets depends upon the similarity threshold chosen and so the number of seed words formed can be varied by adjusting this threshold.

3.3 Feature Extraction

Feature selection techniques are successful in reducing dimensionality, however, they tend to produce very sparse representations of text that can harm retrieval performance.

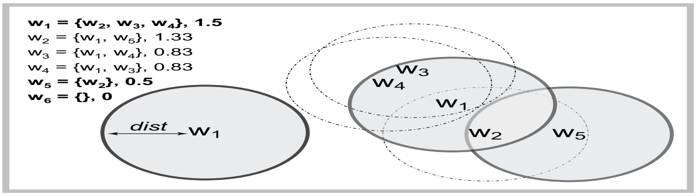
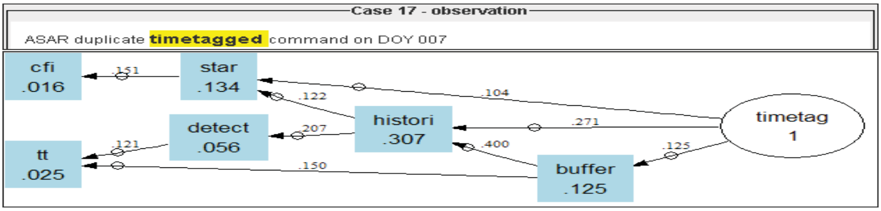

$w_1 = \{w_2, w_3, w_4\}$, 1.5
$w_2 = \{w_1, w_5\}$, 1.33
$w_3 = \{w_1, w_4\}$, 0.83
$w_4 = \{w_1, w_3\}$, 0.83
$w_5 = \{w_2\}$, 0.5
$w_6 = \{\}$, 0
dist
w_1
w_3
w_4
w_1
w_2
w_5

Fig. 2. Seed word selection using the footprint technique

We investigate two feature extraction techniques, that form a new set of features from the original features, to address the issue of sparseness.

Feature Extraction using Word Co-Occurrence

Each seed word forms a feature in the case representation. A feature value is derived based on the presence of a seed word in the report. Using seed words alone in this way to represent free text results in a sparse representation. This is because reports may still be similar even though they may not contain seed words. One way around this problem is to embed the context of the seed within the case. We achieve this by the induction of feature extraction rules [19,20].


Case 17 - observation

ASAR duplicate timetagged command on DOY 007

Fig. 3. Feature extraction rules

Each rule associates words with a selected seed word, such that the rule conclusion contains the seed word and the rule body (or conditions) consists of associated words. The presence of associated words in a report (in the absence of the seed word) activates the rule, inferring a degree of seed word presence in the report. Essentially with increasing rule activations the problem with sparse representation decreases.

Consider the text snippet in Figure 3 taken from the observation section of a particular anomaly report. Here the snippet happens not to contain any of the seed words discovered for observation parts of reports. This would typically lead to an *empty* representation, if only a feature selection approach is employed. However an associated word, “timetagged”, identified by rule induction, and highlighted in the text, has led to a series of rule activations as shown in the lower part of Figure 3. The outcome of this is that six seed words, shown in the highlighted boxes, can now be instantiated because of their association with “timetagged”. Importantly for case comparison this means that other cases containing these seed words that previously would have been considered distant can now be considered more similar.

	Context Relevance					Expert's Comments
Feedback for Seed Word	1	2	3	4	5	
TLM						
TLM <= lost				x		Also a type of problem about telemetry: we loose it!
TLM <= process & receive		x				We receive it and we process it. These are actions
TLM <= lock					x	Bingo: We sometimes have "Telemetry lock problems"
TLM <= telemetry						It is the abbreviation / synonym
TLM <= available & generated	x					a bit weird
TLM <= product & generated		x				A more elaborated processing of the telemetry

Fig. 4. Feature extraction rules concluding the seed word, TLM

Similarity computation requires that a mechanism is in place to facilitate the comparison of feature values. With CAM the process of translating rule activations into feature values involves combining evidence from multiple rule activations and propagating this evidence through rule chains. Key to this are rule accuracy values also referred to as confidence scores shown on the arcs between terms. Essentially when a rule with high confidence is activated it suggests higher belief in the presence of the seed word. We use a basic spreading-activation mechanism to propagate these confidence scores using an aggregation mechanism similar to the MYCIN approach to combining evidence for medical reasoning [4]. Here, if two rules x and y activate concluding the same seed word, then the confidences are aggregated to generate the feature value for the feature represented by the seed as follows: conf(x) + conf(y) - conf(x)*conf(y). The aggregated confidence values derived from this approach, for our example, are shown below each seed word in Figure 3. The resulting representation now has 6 features instantiated with values between 0 and 1. It is interesting to note that when rules are triggered they implicitly capture latent higher order relationships (e.g. "timetag" associated with "tt" and "cfi" via "buffer"). These discovered relationships provide a more informed case comparison compared to one that is based solely on seed word presence alone.

We use the Apriori [1] association rule learner to extract feature extraction rules. Apriori typically generates many rules, and requires that confidence, support and discriminatory thresholds be set before useful rules are generated. Here expert feedback on the quality of generated rules is vital. The explicit nature of rules is an obvious advantage both to establish context and also to acquire expert feedback (see Figure 4).

Feature Extraction using Similarity Neighbourhoods

Our *seed word selection by similarity clustering* feature selection approach identifies seed words that are representative of a set of similar words and uses the seed words to represent the document. Similar words are those that have similar co-occurrence patterns with words contained in the solution. Rather than instantiating the feature only if the seed word itself is present it would appear sensible to instantiate the feature if either the seed word or any of the words it represents are present in the document. Feature extraction using similarity neighbourhoods does this. If the seed word is present the feature is given the value 1 as with the feature selection approach. However, sparseness is reduced by instantiating the feature if any related word contained in the seed word's coverage or reachability sets are present in the document. The feature value is set to equal the similarity between the seed word and the related word. Where multiple related words are present in the document the similarities are combined using the MYCIN approach discussed earlier.

4 The CAM Prototype

Our CAM demonstrator uses a structured representation of the anomaly reports, created by one or a combination of the feature selection/extraction processes described in Section 3. The representation process provides a 5 part case representation for each report. One of these contains the 13 features from the original structured report fields, while the remaining four parts are representations of the text data in the observation, description, recommendation and resolution fields of the report. These are represented by 70, 103, 94, and 156 features respectively, and correspond to the number of seed words extracted by the footprint-based feature selection. The similarity threshold controlling this extraction was set to encourage balanced word clusters. We are currently working with a sample of 960 reports, supplied by ESA.

The retrieval strategy implemented on CAM uses the *k* Nearest Neighbour algorithm (*k*-NN) to identify the k most similar cases to the current problem. The relative importance of each section or form (1 structured + 4 text) can be established by setting a form weight while at a more fine-grained level the importance of each feature within a form can be set with feature weighting. Three alternative distance measures can be selected in CAM to measure the relationship between anomaly reports: Manhattan, Euclidean, and Cosine. Because the representations are sparse the Manhattan and Euclidean measures have been adapted to consider only instantiated features and ignore zero valued features when calculating the distance between reports.

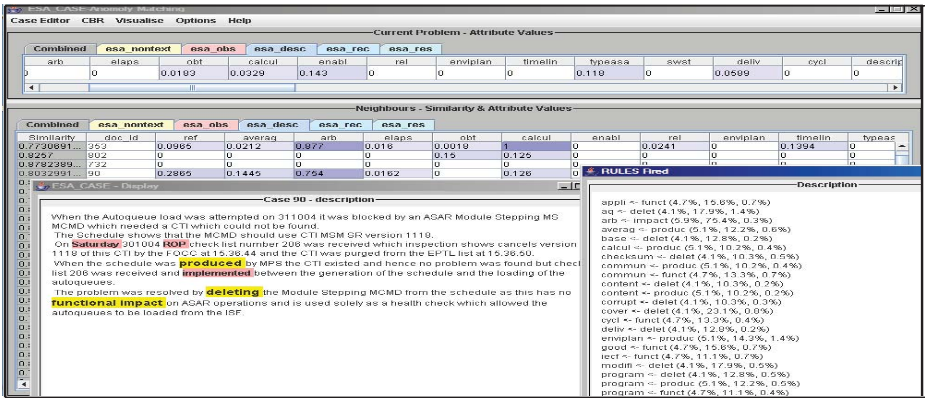
CAM provides an interface (Figure 5) that displays the current target report at the top with a ranked list of similar, retrieved reports below along with their similarity scores. Individual forms can be viewed by selecting the tab for the appropriate pane. Given the sparse representation, instantiated fields are colour highlighted to allow relationships between reports to be easily viewed. A gradient (darker for higher values) is applied to the highlighting to identify the confidence in a words presence in a report. The selected seed words are used in the structured representation to label the features.

The structured representation is the default report view, however, alternative views display the original text as displayed on the bottom left of Figure 5. A two colour word annotation is applied to the text. Seed words are annotated in yellow while any terms forming the body of induced rules are annotated in pink. Feature extraction rules induced from the text as part of the representation process, can be viewed as a list (displayed on the bottom left of Figure 5) or as a graph as shown in Figure 3.

Two additional visualisations are available to assist the user compare similarities and differences between retrieved reports [22]. A parallel co-ordinate plot shows the similarity of the retrieved nearest neighbours to the current report while a second visualisation uses the spring-embedder model to preserve the similarity relationship between cases as on-screen distances.

5 Experimental Evaluation

It is generally accepted that evaluation is a challenge for TCBR systems. Standard IR systems advocate precision and recall based evaluation on tagged corpuses. The manual tagging involves not only class assignment but often assignment of relevance



arb	elaps	obt	calcul	enabl	rel	enviplan	timelin	typeasa	swst	deliv	cycl	descrip
0	0	0.0183	0.0329	0.143	0	0	0	0.118	0	0.0589	0	0

Similarity	doc_id	ref	averag	arb	elaps	obt	calcul	enabl	rel	enviplan	timelin	typeas
0.7730691...	353	0.0965	0.0212	0.877	0.016	0.0018	1	0	0.0241	0	0.1394	0
0.8257	802	0	0	0	0	0.15	0.125	0	0	0	0	0
0.8782389...	732	0	0	0	0	0	0	0	0	0	0	0
0.8032991...	90	0.2865	0.1445	0.754	0.0162	0	0.126	0				

Case 90 - description

When the Autoqueue load was attempted on 311004 it was blocked by an ASAR Module Stepping MS MCMD which needed a CTI which could not be found.
The Schedule shows that the MCMD should use CTI MSM SR version 1118.
On Saturday 301004 ROP check list number 206 was received which inspection shows cancels version 1118 of this CTI by the FOCC at 15.36.44 and the CTI was purged from the EPTL list at 15.36.50.
When the schedule was **produced** by MPS the CTI existed and hence no problem was found but checl list 206 was received and **implemented** between the generation of the schedule and the loading of the autoqueues.
The problem was resolved by **deleting** the Module Stepping MCMD from the schedule as this has no **functional impact** on ASAR operations and is used solely as a health check which allowed the autoqueues to be loaded from the ISF.

```
appli <- funct (4.7%, 15.6%, 0.7%)
aq <- delet (4.1%, 17.9%, 1.4%)
arb <- impact (5.9%, 75.4%, 0.3%)
averag <- produc (5.1%, 12.2%, 0.6%)
base <- delet (4.1%, 12.8%, 0.2%)
calcul <- produc (5.1%, 10.2%, 0.4%)
checksum <- delet (4.1%, 10.3%, 0.5%)
commun <- produc (5.1%, 10.2%, 0.4%)
commun <- funct (4.7%, 13.3%, 0.7%)
content <- delet (4.1%, 10.3%, 0.2%)
content <- produc (5.1%, 10.2%, 0.2%)
corrupt <- delet (4.1%, 10.3%, 0.3%)
cover <- delet (4.1%, 23.1%, 0.8%)
cycl <- funct (4.7%, 13.3%, 0.4%)
deliv <- delet (4.1%, 12.8%, 0.2%)
enviplan <- produc (5.1%, 14.3%, 1.4%)
good <- funct (4.7%, 15.6%, 0.7%)
iecf <- funct (4.7%, 11.1%, 0.7%)
modifi <- delet (4.1%, 17.9%, 0.5%)
program <- delet (4.1%, 12.8%, 0.5%)
program <- produc (5.1%, 12.2%, 0.5%)
program <- funct (4.7%, 11.1%, 0.4%)
```

Fig. 5. Screen shot of CAM's interface

judgements on retrieved sets. In practical situations it is clearly prohibitive to expect a domain expert to tag substantial numbers of cases with relevance judgements.

Our initial evaluation approach was to acquire qualitative feedback on a few selected test cases. A structured representation was created for each case using our preprocessing techniques, seed word selection by word similarity clustering, and feature extraction rules. Five probe reports were randomly selected and for each probe the 3 most similar reports were retrieved by CAM. A further 3 randomly selected reports were then added to create a retrieval set size of 6. Each probe and corresponding retrieval set was presented to the domain expert to obtain our expert's feedback. The results of our initial study [22] show reasonable cases are being retrieved. However, qualitative evaluations are expensive, in terms of domain expert time, and only consider a small sample of available documents.

CAM requires choices to be made between major factors such as alternative feature selection and extraction techniques in addition to fine tuning numerous other factors that have an effect on retrieval performance e.g. between alternative distance measures or neighbourhood sizes to develop appropriate similarity knowledge. It soon became clear that manual evaluation of the iterative development cycle required to optimise our design would require an excessive level of involvement by the domain expert to the extent it was impractical. We require an empirical evaluation measure to allow us to choose between alternative system designs.

5.1 Alignment Measure

"Similar problems have similar solutions" is one of the fundamental assumptions that underpins CBR as a suitable problem-solving methodology for a particular problem domain. This assumption is often taken for granted, whereas, in fact it is a measure not only of the suitability of CBR for the domain but also of the competence of the system design in terms of case representation and similarity knowledge. If we can measure the alignment between the problem and solution space in terms of extent to which *"similar problems have similar solutions"* holds true for different system design configurations we have a measure of design competence.

In previous work, on supervised problems, the alignment between problem and solution space has been measured by looking at the mix of solution classes present among a case's neighbours in the problem space [11]. In unsupervised tasks cases are not assigned class labels, however, it is still possible to measure the local mix in solutions where the similarity between solutions can be measured e.g. where the solutions are in textual form. Weber et al. [18] use similarity in the solution space to cluster the case base and provide information on feature importance in the problem space. Case cohesion [9] measures level of overlap in retrieval sets retrieved independently from the problem and solution space, however, it is unclear on how to set suitable similarity thresholds that control the retrieval set sizes. We measure the alignment between the problem and solution space by considering the mix of similarities among solutions present in a set of neighbours retrieved in the problem space.

In a *good* design cases identified as having the most similar problems to a target case will also have the most similar solutions. This is exactly what our case alignment measures. If a CBR system processes problems in a problem space P and solutions for these problems belong to a solution space S. Let C be the set of cases in the case base containing cases $\{c_1, ..., c_n\}$. Cases consist problem/solution pairs such that $c_i = \{p_i, s_i\}$ where $p_i \in P$ and $s_i \in S$. Using the case base to represent future problems that will be faced, i.e. the representative assumption, each case becomes the target problem t in turn and we measure the alignment between P and S in the local neighbourhood of t. A distance function $D(t, p_i)$ or $D(t, s_i)$ measures the distance between t and c_i in either the problem or solution space giving a value between 0 and 1.

In Figure 6, t is identified in both P and S. Using $D(t, p_i)$, t's three nearest neighbours (NN) in P are found, shown as $p_1, p_2, \&p_3$. If we initially consider only c_1, consisting $\{p_1, s_1\}$, we can calculate the alignment of t in relation to c_1 ($Align(t, c_1)$) by comparing the distance in the solution space of t to s_1 with the distance to the nearest (Ds_{min}) and most distant solutions (Ds_{max}) in the case base, as shown below.

$$Align(t, c_1) = 1 - \frac{(D(t, s_1) - Ds_{min})}{(Ds_{max} - Ds_{min})}$$

The overall case alignment for t (CaseAlign(t)) is found by taking a weighted average of the alignment with its individual NN retrieved in the problem space. The size of the neighbourhood used would typically be the same as used for retrieval; a neighbourhood size of 3 is shown in Figure 6.

$$\text{CaseAlign(t)} = \frac{\sum_{i=1} (1 - D(t, p_i)) * Align(t, c_i)}{\sum_{i=1} (1 - D(t, p_i))}$$

In local areas where a case's NN in the problem space are also its NN in the solution space there is a strong alignment between problem and solution space and case alignment values will be close to 1; conversely, in areas in which a case's NN in the problem space are not close in the solution space the alignment is poor and case alignment values will be low. Case alignment allows us to evaluate alternative system design configuration and make informed maintenance decisions about individual cases.

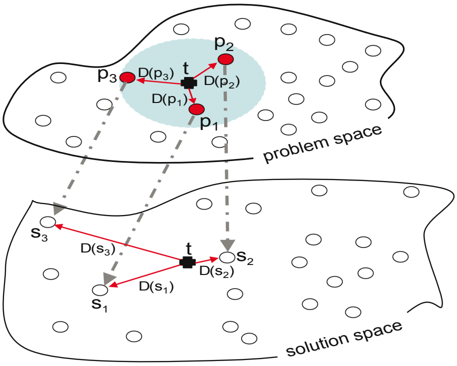


Fig. 6. Case alignment calculation

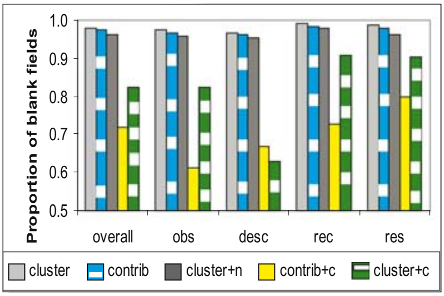


Fig. 7. Comparison of sparseness of different case representations

5.2 Representations

It is our contention that a *good* design will exhibit better alignment between the problem and solution space. Thus, looking at the mix of case alignments present in the case base provides an empirical evaluation technique and an alternative to the typical approach in which a domain expert manually tags a small set of retrieved documents. Using case alignments to evaluate alternative design configurations has several advantages: the burden on domain experts can be reduced; a more comprehensive evaluation is obtained rather being limited to a small sample; and fine-tuning of design variables becomes possible by adopting an iterative development process.

Our new alignment measure can now be used to evaluate 5 alternative case representations, constructed using a combination of the techniques described in section 3. The representations, all of which share the same pre-processing technique but differ in their combination of feature selection and extraction techniques, are described below.

- **CLUSTER:** Created using word similarity clustering to identify seed words only with no feature extraction rules.
- **CONTRIB:** Features are selected by the word contribution technique alone.
- **CLUSTER+N:** Case representation combines word similarity clustering with the word neighbourhood extraction technique.
- **CONTRIB+C:** Features selected by word contribution with feature extraction rules.
- **CLUSTER+C:** Uses word similarity clustering with feature extraction rules.

One of the key problems that needs to be addressed in creating structured representations of text is how to deal with the inherent data sparsity. Our five representations each take a different approach with the latter three representations, which include feature extraction techniques, being developed specifically to address the sparsity problem. Figure 7 gives a measure of sparsity, showing the proportion of blank fields present in the case base for each form and representation. The feature selection only representations are very sparse (0.981 for CLUSTER and 0.974 for CONTRIB). Surprisingly, the word similarity neighbourhood extraction technique used in CLUSTER+N does little to reduce sparsity with an overall value about 1% lower at 0.963. However, both

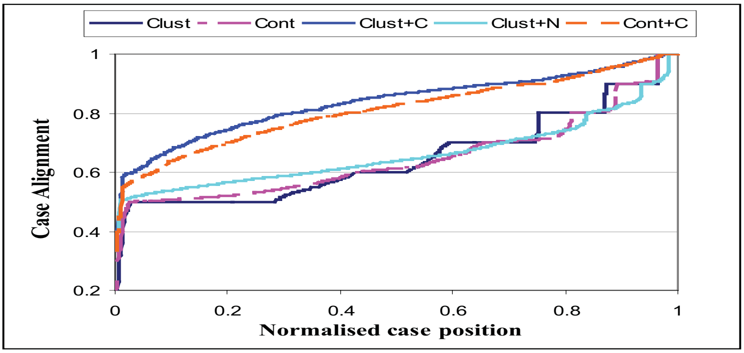


Fig. 8. Profiles of case alignment for alternative representations

representations using word co-occurrence extraction show a substantial reduction in sparsity with CONTRIB+C being least sparse at 0.720 compared to 0.825.

We create a profile of case alignments for each of the five representations. Case alignment is calculated for each case in the case base and the cases are ranked in ascending order of alignment. The profile shows the mix of individual case alignments present in the case base for a particular representation. Figure 8 plots the profile of the five representations being considered. Each profile is created by plotting case alignment against the normalised position of the case in the ranked list of case alignments. Thus each curve is a profile of the case alignments for one representation and a point on a curve gives the alignment value on the y-axis for a particular case whose relative position in the ranked list is shown on the x-axis. Representations that give profiles with higher case alignments are considered *better* designs. The alignment profiles fall into 2 groups coinciding with representation sparsity.

– representations incorporating the word co-occurrence feature extraction technique (CONTRIB+C and CLUSTER+C) give superior results showing a better alignment between problem and solution space providing support for the use of this approach. CLUSTER+C slightly outperforms CONTRIB+C with an average alignment of 0.836 compared to 0.810 even although its representation is more sparse. CLUSTER+C is our chosen representation for the domain.
– there is little to choose between the three representations showing poorer alignment. The 2 feature selection only approaches(CLUSTER and CONTRIB) both have an average alignment of 0.646 although CONTRIB gives a more even distribution across the case base. CLUSTER+N gave disappointing results with only marginal improvements in alignment as a result of the feature extraction stage with an average alignment of 0.662.

A similar evaluation approach was undertaken to evaluate and choose between alternative distance measure used in identifying the relationship between cases and for selecting a suitable neighbourhood size for the number of retrieved cases to return at the retrieval stage. Manhattan distance was seen to exhibit slightly better alignment than the Euclidean or Cosine measures while a neighbourhood size of 5 was found to give a good compromise between good alignment, found in smaller neighbourhoods, and minimising the impact of noise with larger neighbourhoods.

6 Related Work

A common problem for TCBR system development is the demand on knowledge acquisition. For instance in the ExperienceBook project (aimed at supporting computer system administrators) all knowledge was acquired manually. This is not an exception, because current practice in TCBR system development show that the indexing vocabulary and similarity knowledge containers are typically acquired manually [17]. Consequently maintenance remains a problem since these systems are not able to evolve with newer experiences. These difficulties have created the need for fully or semi automated extraction tools for TCBR.

Tools such as stemming, stop word removal and domain specific dictionary acquisition are frequently used to pre-process text and are mostly automated. Acquiring knowledge about semantic relationships between words or phrases is important but is harder to automate. Although NLP tools can be applied they are often too brittle partly because they tend to analyse text from a purely linguistic point of view. Furthermore the reliance on deep syntactic parsing and knowledge in the form of generative lexicons still warrants significant manual intervention [6].

Research in text classification and information retrieval typically adopts statistical approaches to feature selection and extraction. The main pre-requisite is access to a significant number of cases. With the anomaly reporting problem domain case base size is not a constraint. Consequently, word co-occurrence based analysis becomes particularly attractive for automated indexed vocabulary acquisition. A common approach to determining representative features involve the use of distributional clustering approaches [13], and has since been adopted for feature extraction with supervised tasks [15,2]. Of particular importance for word clustering are distributional distance measures. These measures ascertain distance by comparison of word distributions conditioned over a disjoint target set. Typically, class labels are the set of targets and so cannot be applied to unsupervised tasks. However, in the Sophia retrieval system reliance on class labels was dropped by comparing word distributions conditioned on other co-occurring words (instead of class labels) [12]. Unlike with anomaly reports, Sophia operates on IR like documents, hence there is no requirement to learn from the differences between solution and problem space vocabulary. Our approach to calculating distributional distances is novel in that words from the problem space are compared conditioned on the solution space. This creates a distance measure that is guided by both the similarity and differences between problem and solution vocabularies.

Formation of newer and improved dimensions for case representation fall under feature extraction research. Lsi is a popular dimensionality reduction technique particularly for text. Extracted features are linear combinations of the original features which unfortunately lack in expressive power [5]. Modelling keyword relationships as rules is a more successful strategy that is both effective and remains expressive. A good example is Ripper [3], which adopts complex optimisation heuristics to learn propositional clauses for classification. Unlike Ripper rules, association rules do not rely on class information and incorporates data structures that are able to generate rules efficiently making them ideal for large scale applications [24,7]. The seed generalisation approach discussed in this paper is similar to that employed by the PSI tool introduced in [20], but unlike PSI here generalisation does not rely on class knowledge.

7 Conclusions and Future Work

The paper presents an approach to case retrieval applied to anomaly reports and is implemented in the CAM prototype. It is a first step towards developing a CBR system to support the ESA's anomaly report processing task. Like most text applications, anomaly processing is unsupervised and requires automated knowledge acquisition tools that are not reliant on class knowledge.

The paper introduces a novel unsupervised index vocabulary acquisition mechanism to map unstructured parts of text data to a structured case representation. For this purpose word pair-wise distances are calculated according to similarity in co-occurrence patterns over the solution space. This facilitates problem space words to be considered similar with specific reference to the solution space vocabulary.

Seed words are identified using word clusters and forms the features vector for the case representation. The idea of using a footprint-based feature selection strategy is novel. It facilitates selection of representative and diverse words but importantly does not require that the number of seed words be pre-specified. It does however require a similarity threshold to be in place which directly controls the feature vector size.

A novel case alignment technique measures the extent to which *similar problems have similar solutions*. Alignment to some extent depends on the underlying characteristic of the problem domain, however, it is also a measure of the effectiveness of the particular system design configuration being evaluated. The problem and solution space was shown to be most aligned with CLUSTER+C, which combined word similarity clustering with feature extraction using co-occurrence rules. In addition to a global profile of the case base, individual case alignments also provide local information about the relationship between the problem and solution space. It is planned to utilise this local knowledge to develop maintenance approaches for unsupervised domains.

A common approach for setting retrieval weights in supervised problems is to learn feature importance from the available cases. Our alignment measure gives the opportunity to apply similar techniques to learn weights that improve alignment and will be investigated in future work. In a similar vein we have yet to establish a principled approach to setting Apriori's parameters and the similarity threshold for the feature vector size. Case alignment can assist in the optimisation of these design parameters. Future work will also extend CAM for the reuse and revision stages of the CBR cycle.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.: Fast discovery of association rules. In: Advances in Knowledge Discovery and DM, pp. 307–327 (1995)
2. Baker, L., McCallum, A.: Distributional clustering of words for text classification. In: Proceedings of the 21st ACM Int Conf on IR, pp. 96–103. ACM Press, New York (1998)

3. Cohen, W., Singer, Y.: Context-sensitive learning methods for text categorisation. ACM Transactions in Information Systems 17(2), 141–173 (1999)
4. Davis, R., Buchanan, B., Shortliffe, E.: Production Rules as a Representation for a Knowledge-Based Consultation Program. Artificial Intelligence 8, 15–45 (1977)
5. Deerwester, S., Dumais, S., Landauer, T., Furnas, G., Harshman, R.: Indexing by latent semantic analysis. American Society of Information Science 41(6), 391–407 (1990)
6. Gupta, K., Aha, D.: Towards acquiring case indexing taxonomies from text. In: Proceedings of the 7th Int FLAIRS Conf, pp. 307–315 (2004)
7. Kang, N., Domeniconi, C., Barbara, D.: Categorization and keyword identification of unlabelled documents. In: Proceedings of the 5th IEEE Int Conf on Data Mining (2005)
8. Lee, L.: On the effectiveness of the skew divergence for statistical language analysis. In: Artificial Intelligence and Statistics, pp. 65–72 (2001)
9. Lamontagne, L.: Textual CBR Authoring using Case Cohesion. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, Springer, Heidelberg (2006)
10. Liu, T., Liu, S., Chen, Z., Ma, W.: An evaluation on feature selection for text clustering. In: Proc. of the 12th Int. Conf. on ML, pp. 488–495 (2003)
11. Massie, S., Craw, S., Wiratunga, N.: Complexity profiling for informed Case-Base Editing. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, pp. 325–329. Springer, Heidelberg (2006)
12. Patterson, D., Rooney, N., Dobrynin, V., Galushka, M.: Sophia: A novel approach for textual case-based reasoning. In: Proc of the 19th IJCAI Conference, pp. 1146–1153 (2005)
13. Pereira, F., Tishby, N., Lee, L.: Distributional clustering of English words. In: Proc of the 30th Annual Meeting of the Association for Computational Linguistics, pp. 183–190 (1993)
14. Salton, G., McGill, M.: An introduction to modern IR. McGraw-Hill, New York (1983)
15. Slonim, N., Tishby, N.: The power of word clusters for text classification. In: Proc of the 23rd European Colloquium on IR Research (2001)
16. Smyth, B., McKenna, E.: Footprint-based Retrieval. In: Althoff, K.-D., Bergmann, R., Branting, L.K. (eds.) Case-Based Reasoning Research and Development. LNCS (LNAI), vol. 1650, pp. 343–357. Springer, Heidelberg (1999)
17. Weber, R., Ashley, K., Bruninghaus, S.: Textual case-based reasoning. To appear in The Knowledge Engineering Review (2006)
18. Weber, R., Proctor, J.M., Waldstein, I., Kriete, A.: CBR for Modeling Complex Systems. In: Muñoz-Ávila, H., Ricci, F. (eds.) ICCBR 2005. LNCS (LNAI), vol. 3620, pp. 625–639. Springer, Heidelberg (2005)
19. Wiratunga, N., Koychev, I., Massie, S.: Feature Selection and Generalisation for Retrieval of Textual Cases. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 806–820. Springer, Heidelberg (2004)
20. Wiratunga, N., Lothian, R., Chakraborty, S., Koychev, I.: Propositional approach to textual case indexing. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 380–391. Springer, Heidelberg (2005)
21. Wiratunga, N., Lothian, R., Massie, S.: Unsupervised Feature Selection for Text Data. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, pp. 340–354. Springer, Heidelberg (2006)
22. Wiratunga, N., Massie, S., Craw, S., Donati, A., Vicari, E.: Case Based Reasoning for Anomaly Report Processing. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, pp. 44–49. Springer, Heidelberg (2006)
23. Yang, Y., Pedersen, J.: A comparative study on feature selection in text categorisation. In: Proc. of the 14th Int. Conf. on ML, pp. 412–420 (1997)
24. Zelikovitz, S.: Mining for features to improve classification. In: Proc. of ML Models, Technologies and Applications (2003)