NTNU
Kunnskap for en bedre verden

Department of Computer Science

TDT4173 - Assignment 1

Case Based Reasoning

MAC/FAC: Many Are Called, Few Are Chosen

Group:
Group 6

Authors:
Jonatan Engstad (jonatane)
Adrian Langseth (adrianwl)
Miriam Woldseth (miriamvw)

July 14, 2021

Abstract

This paper explores the MAC/FAC retrieval method and its use in the retrieval step of the CBR cycle. The goal of the method is to retrieve the k cases in the case base that are the most similar to a given query. The method attempts to accomplish this goal by first collecting a candidate set in the MAC stage, and then assessing this set and retrieving the most similar case(s) from it in the FAC stage. The performance of the approach is highly dependent on the MAC stage's ability to retrieve a candidate set consisting of few but relevant cases, i.e. cases that are similar to the target problem. This is as opposed to the most basic retrieval method, sequential retrieval, which evaluates each case in the case base one by one, thus guaranteeing retrieval of the most similar case(s). A well-implemented MAC/FAC approach can significantly increase retrieval speed compared to sequential retrieval, while simultaneously retrieving relevant cases. However, the method cannot guarantee retrieval of the most relevant case(s) as sequential retrieval can. The paper also explores other approaches that, like MAC/FAC, cannot guarantee retrieval of the optimal case(s), but can significantly increase retrieval speed compared to sequential retrieval when well-implemented. These methods are footprint-based retrieval and index-based retrieval, and can be considered as an alternative to the MAC/FAC approach. However, the MAC/FAC approach can be advantageous when there is a clear split in the case features, with one part being cheap to perform similarity measures on, and the other being more expensive. In these cases the MAC stage can be used for the part assessed with a computationally inexpensive similarity measure, thus reducing the amount of cases that needs to be assessed with a computationally expensive similarity measure in the FAC stage. An application that utilizes the approach on cases with clearly split features, is selfBACK. In selfBACK each case consists of both subjective data and activity data, where the subjective data is assessed with a computationally inexpensive similarity measure, and the activity data with a computationally expensive one. Since approximately 1% of the data is subjective data, the number of cases assessed with the computationally expensive similarity measure is greatly reduced by utilizing MAC/FAC, thus significantly increasing the retrieval speed of selfBACK.

Table of Contents

List of Figures

1 Introduction

Ever since the invention the modern computer, researchers have attempted to simulate intelligence - trying their best to create machines that can perceive, experience, and then learn from those experiences. And while they aren't there yet, major strides have been made since the 50's and 60's, when the field was just getting started. Modern AI programs can learn to master video games, recognize thousands of different objects in pictures, and generate text that is almost indistinguishable from that written by humans.

One technique in the field of machine learning – one that has helped lead to machine learning's resurgence in modern times – is supervised learning. Supervised learning aims to make programs that learn by giving them large sets of problems along with their answers and then tasking the programs with synthesizing rules (often called the hypothesis or model) for distinguishing problems based on their answers [Russel and Norvig, 2010].

Supervised learning algorithms may be either eager or lazy, depending on if they build models based upon the data they were given. While eager learners first build a model and then query that model when given a new problem, lazy learners do the opposite - building no model and searching the raw data when given a query [Mitchell, 1997]. One method that employs lazy learning is instance-based learning, a method which constructs its hypothesis directly from the instances themselves (hence the name) [Russel and Norvig, 2010].

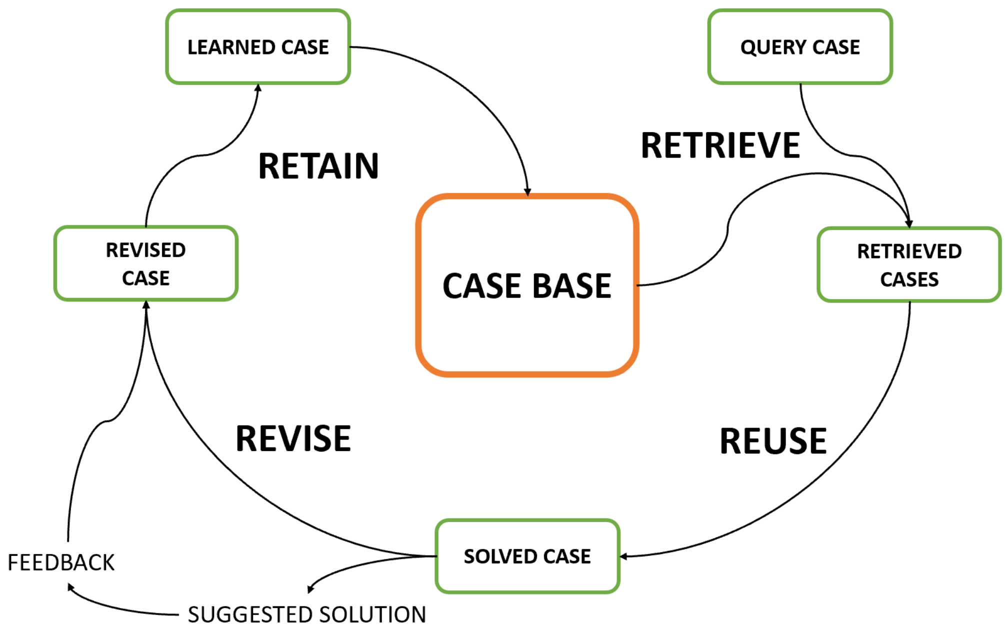

Figure 1: The four Rs of the CBR cycle: Retrieve, Reuse, Revise, Retain. The system is given a new case (a "query"), retrieves similar cases, reuses the solution of the retrieved cases and receives feedback on how well the solution fit. Depending on the feedback, the retrieved solution may be revised, before then being stored in the case base together with the query.

Source: Adapted from Aamodt and Plaza [1994]

An old, but still commonly used approach to instance based learning is that of Case Based Reasoning (CBR). CBR first emerged in the late 1980's building on various ideas from research on schema-oriented memory models [Richter and Weber, 2013]. The core idea of the method is that one can adapt knowledge of how previous problems were solved to solve new, but similar problems. This is done in a 4-phase cyclical process, illustrated in Figure 1, often referred to as the four R's: Retrieve, Reuse, Revise and Retain. When a CBR system encounters a new problem, it first retrieves similar problems from its database of previous problems (the so called "case base"), and then reuses the solution to these problems to solve the problem at hand. Depending on the outcome, the system may then revise the solution to better fit the problem before storing the new problem and solution in the case base for later retrieval [Richter and Weber, 2013].

To have a well-working CBR system, it is absolutely vital to have both an efficient and accurate retrieval phase: The system needs to be able to respond in a timely fashion, and the problems it retrieves must have solutions that are applicable to the encountered problem [Müller and Bergmann, 2014]. While there are several techniques that, given the right data and requirements, can achieve this, this paper will focus on one of them: the MAC/FAC method [1] of retrieval.

2 Foundations

MAC/FAC, short for "many are called, but few are chosen", is a retrieval algorithm used to find the most similar items in a dataset, in relation to some query [Richter and Weber, 2013]. In the context of CBR, MAC/FAC is used at the retrieval stage of the "4R" CBR cycle. Although not originally introduced as a method for CBR, it was quickly picked up by the CBR research community and has since then seen widespread use within the field.

MAC/FAC is a two-stage retrieval process, meaning that it performs the retrieval process in two separate stages. The first stage, MAC, performs a lightweight selection on the whole case base, selecting items deemed potentially relevant and placing them into a candidate set. The second stage, FAC, goes through the candidate set generated in the first stage and performs a more computationally heavy similarity measure on the items within it, returning the item(s) with the highest score. Figure 2 illustrates this process for a simple two-dimensional, euclidian-space dataset.

Figure 2: MAC/FAC retrieval visualized for a simple, two-dimensional dataset. Note that this is a simplified example. It is not unusual for a CBR application to track non-euclidian features that require heavy computation to get a proper similarity measure.

The two-stage process can be likened to the job of a customs agent who's on the lookout for a very infamous smuggler. The agent cannot herself know that whoever she apprehends truly is the smuggler, so her task is simply to apprehend anyone whose passport details fit some criteria. But the airport is very busy, and if she were to stop every passing person and inspect their passport, no one would be able to catch their flights. Luckily the agent has an ace up her sleeve: She knows that

[1] It is important to note that MAC/FAC, much like CBR itself, is as much a concept as it is a concrete algorithm.

smugglers are very likely to be wearing a Marius sweater and aviator sunglasses. Using this much faster visual approximation, she can quickly assess whose passport she should be inspecting. And, as most of the people passing through are not wearing this specific combination of sweater and glasses, she can let most of the crowd pass through customs without any hassle. Of course, there's always a chance that the smuggler in question on this day decided to wear a bunad and clubmaster sunglasses instead. Then the customs agent may instead end up apprehending a smaller fry who also was out smuggling on that day, or maybe apprehending no smugglers at all.

2.1 Origin

MAC/FAC was first discussed in a 1991 paper by Gentner and Forbus in *Proceedings of the Cognitive Science Society*. In their paper, Gentner and Forbus discussed their attempt at capturing the way in which people retrieve information by examining how participants recognized both superficial and structural similarities in stories they had read.

In the paper, Gentner and Forbus describe asking subjects to read a large number of stories. The subjects were then asked back two weeks later, to read a new set of stories that matched the previous ones in various ways - some were true analogies, while others only shared surface details. The subjects were then asked to rate the analogousness of these pairs of stories. Gentner and Forbus found that while subjects were much more likely to retrieve surface matches, they also rated true analogies as being considerably more sound.

From this, Gentner and Forbus concluded that the processes of analogical access and analogical inference may be distinct - the first: quick and superficial, the second: slow, and more considerate of deeper connections. Extracting this information into a similarity based retrieval process, they landed on the two-stage MAC/FAC retrieval model.

Mirroring the way in which people quickly find superficial connections between stories, Gentner and Forbus used the dot product of the stories' term frequency vectors (see section 17.5.1 of Richter and Weber [2013]) as their measure of superficial similarity. This choice of similarity measure for the MAC stage meant that the similarity rating of the stories would be proportional to the number, and frequency, of words they had in common.

Stories deemed similar enough were then sent to the FAC stage, where the structural mapping engine, an implementation of a tool developed by Gentner [Gentner, 1983] to perform analogical matching, was used to rate the structural similarities of the stories.

2.2 Implementation

Since first being mentioned in the 1991 paper by Gentner and Forbus, the MAC/FAC method has been generalized so that it can be applied in many more contexts. Richter and Weber [2013] describes the steps of the MAC/FAC methods as follows:

> *The initial step for the candidates selects $C_q \subseteq CB$ by:*
>
> $$C_q := \{p \in CB \,|\, SIM(q,p)\}$$
>
> *The FAC stage uses ... the results of the MAC stage, selecting the best match.*

Here, q is the problem description, also known as the query, and p is a candidate match from the case base CB. C_q is the candidate set that is produced by applying the MAC-stage retrieval algorithm to the case base, with respect to the query. SIM(q,p) is a predicate that considers the query q and the case p, accepting the case into the candidate set if it is deemed similar enough [Richter and Weber, 2013].

The most important aspect of MAC/FAC is the similarity measures that are employed during the two stages. For the method to be advantageous, the MAC stage's similarity measure should be computationally inexpensive and discard many, if not most, of the case base cases. Having a MAC stage that complies with these requirements will allow the FAC stage's similarity measure to be as in-depth and accurate as possible, without being too much of a computational burden.

One way to achieve this aforementioned advantage is to assess cases by *surface* similarity during the MAC stage, and *structural* similarity during the FAC stage. Typically, surface features are represented as attribute-value pairs for each case. These attribute-value pairs can often be too simple to represent the complexity of the cases in practice and it can therefore be necessary to represent the internal structure of the cases as well [Cunningham, 2009]. Based on internal structures, one can assess cases on their structural similarity, which is more likely to result in relevant cases being retrieved [de Mántaras et al., 1997]. For example, in the original MAC/FAC paper by Gentner and Forbus [1991], the FAC stage uses Gentner's structure mapping engine [Gentner, 1983] to assess the analogical similarities [2] of different texts.

Richter and Weber [2013] provided some examples of types of SIM(q, p) predicates that can be useful in the MAC stage:

- Partial equality: p and q agree in at least one attribute.
- Local similarity: The attributes of p and q are similar to some degree above a user defined θ.
- Partial local similarity: The attributes of p and q are sufficiently similar (above some θ) for at least one attribute.

Listing 1 contains a simple Python implementation of a MAC/FAC similar to the one described by Richter and Weber. The example code is meant to illustrate the retrieval stage of a CBR application used to determine whether a fruit is edible.

```python
def retrieve(q, CB, k=1):
    def mac_sim(p, q=q):  # predicate
        return p.genus == q.genus  # assert partial surface similarity

    def fac_sim(p, q=q):  # numerical
        return compare_chemical_makeup(p, q)  # computationally heavy

    candidates = filter(mac_sim, CB)
    return sorted(candidates, key=fac_sim)[-k:]
```

Listing 1: A short MAC/FAC implementation in Python. The MAC stage discards all fruits that are not of the same genus as the query, while the FAC stage selects the k fruits whose chemical makeup is most similar to that of the query fruit.

2.3 Drawbacks

There are two main challenges facing any project that wants to make use of the MAC/FAC method. The first challenge is that it is possible for the MAC stage to disregard very few of the candidates, resulting in the FAC stage receiving a candidate set of a similar size to that of the case base. This will lead to a major slowdown of the system and make the two-fold solution moot.

The second challenge is that the MAC stage may judge actually similar cases as dissimilar enough to not include them in the candidate set. Such misjudgements can severely impact the accuracy of the retrieval process in cases where these excluded-but-actually-relevant cases would be (or be part of) the returned solution if the FAC stage selection algorithm were to be run on the whole case base.

[2] As opposed to *literal similarities* (surface matches). An analogical similarity would be the claim that religion is like opium for the masses due to its ability to relieve immediate suffering in an ultimately harmful way. A surface similarity would be claiming that the two are similar because they share the letters "o" and "i".

2.4 Use Cases

The obvious situation in which the MAC/FAC method is useful is when the case base is large. A MAC stage using a sufficiently accurate and efficient similarity measure could make the workload of the retrieval process many times lighter compared to using the FAC stage similarity measure on the whole data set. In this situation, MAC/FAC should be considered when one or more of the following conditions hold:

- Discerning which cases are relevant can be done quickly and with very high (or perfect) accuracy, but choosing the *most* relevant case of these requires much more computation.
- Finding the *most* relevant case is not that important, as long as the retrieved cases are still highly similar.
- The MAC stage similarity method can be proven to disregard only cases that would not be judged similar by the FAC stage similarity method, and at a much lower computational cost.

3 Alternative Methods

The main goals of the CBR retrieval step is to ensure retrieval completeness and acceptable retrieval time [Müller and Bergmann, 2014]. Retrieval completeness is ensured if all of the k most similar cases to the target problem are retrieved, based on the given similarity measures. An acceptable retrieval speed is hard to define and depends on the system, but the retrieval efficiency should be as high as possible. With these two goals in mind, this section will compare the MAC/FAC retrieval approach to three other retrieval approaches, namely: sequential retrieval, footprint-based retrieval, and index-based retrieval.

3.1 Sequential retrieval

Sequential retrieval is the simplest retrieval approach, and assesses all the cases in the case base sequentially [Richter and Weber, 2013, p. 171-172]. This approach can be used with any similarity measure, but suffers from poor performance when working with larger case bases.

An advantage of sequential retrieval is that it can guarantee retrieval completeness [de Mántaras et al., 2005]. The reason for this is that sequential retrieval assesses all the cases in the case base, computing the similarity between them and the target problem. Thus, the approach can retrieve the cases that are the most similar to the target problem and achieve retrieval completeness. MAC/FAC, however, cannot guarantee retrieval completeness, since the MAC stage might not include all the k most similar cases in the selection it gives to the FAC stage [Richter and Weber, 2013, p. 172-173]. Sequential retrieval may therefore achieve better results than MAC/FAC when considering the completeness of the retrieved cases.

The retrieval speed of MAC/FAC, however, can be superior to that of sequential retrieval when the similarity measure assessed in the FAC stage is the same as the one assessed in the sequential retrieval approach [de Mántaras et al., 2005]. Consider, for example, a case base assessed using structural similarity, which is computationally expensive to assess and thus time consuming. With sequential retrieval, all the cases in the case base will be assessed with this computationally expensive similarity measure, while in MAC/FAC, structural similarity is only assessed in the FAC stage on the candidate set collected in the MAC stage. As a result, the computational time used by the FAC stage depends on the size of the candidate set it receives from the MAC stage. If this set is similar in size to that of the case base, there may not be a significant difference between sequential retrieval and MAC/FAC, and, in the worst case, sequential retrieval may actually be faster than MAC/FAC. This is the result of the sequential retrieval method only computing the structural similarity once per case, while MAC/FAC computes the structural similarity in the FAC stage in addition to the simpler similarity measure used in the MAC stage. However, if the MAC stage produces a candidate set that is small in size compared to the case base, MAC/FAC will have a better retrieval time than sequential retrieval. Even though the MAC stage does assess all the cases in the case base, it does so by assessing a much less computationally expensive similarity measure than the structural similarity assessed in both the sequential retrieval approach and the FAC stage. Thus, the efficiency of MAC/FAC increases as the candidate set generated by the MAC stage decreases.

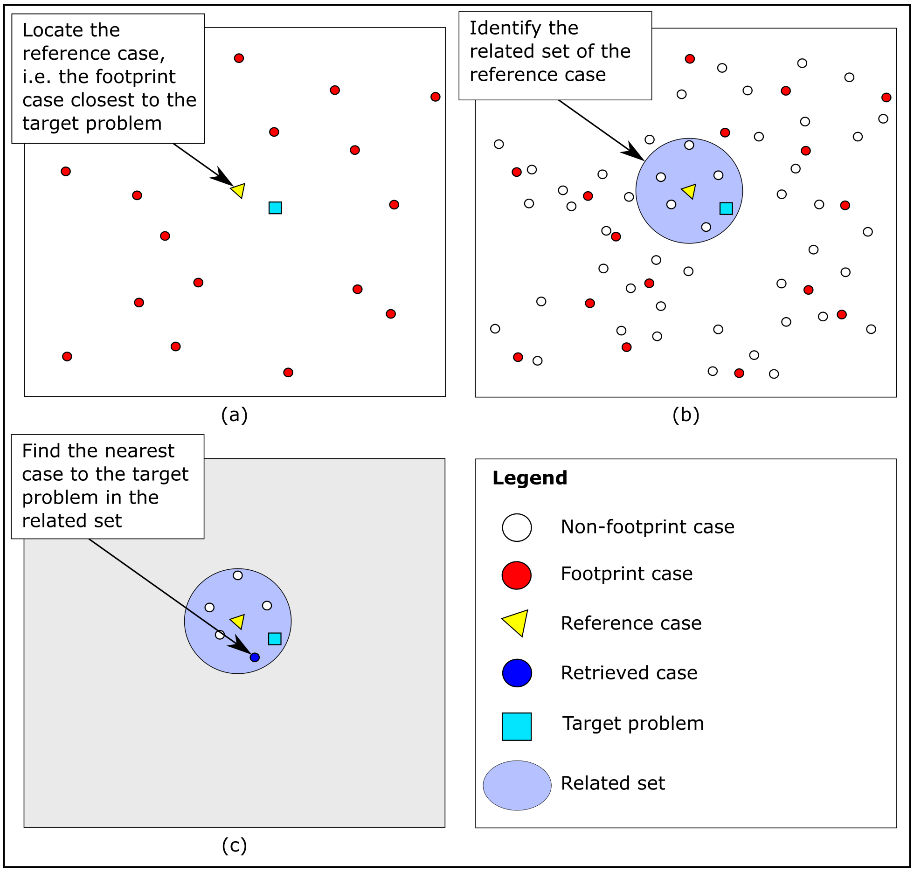

Figure 3: The two-stage retrieval approach footprint-based retrieval. In the first stage (a) the reference case is located, and in the second stage the reference case's related set is found (b), and the most similar case to the target problem in the related set is retrieved (c).

Source: Adapted from Figure 4 in Smyth and McKenna [1999].

3.2 Footprint-based retrieval

The two-stage approach footprint-based retrieval gets its name from the use of a footprint set [de Mántaras et al., 2005, Smyth and McKenna, 1999]. A footprint set is a small subset of all the cases in the case base, that provides the same coverage as the entire case base. This means that the footprint set can solve the same set of problems as the entire case base, however, the case base might contain a more similar case to the target problem than the footprint set. The footprint set is produced by first identifying the competence groups, i.e. clusters of cases where the set of target problems they can solve overlap. After identifying the competence groups, each group's footprint set is computed by first sorting the cases in the group in descending order, so that the cases with larger competence contributions comes first. Then, each case is considered in turn, and added to the group footprint if and only if the current group footprint does not cover the set of target problems the considered case can solve. When computing the group footprint, it might be necessary to go through the group several times, as an addition of a case might lead to some target problem not being covered any more. After all the group footprint sets are found, their union becomes the footprint set of the case base.

The first stage of the footprint-based approach searches the footprint set and identifies the footprint case that is the most similar to the target problem. This footprint case becomes the reference case, as illustrated in Figure 3(a). In stage two, the reference case is used to find its related set, i.e. the set of cases that can either be solved by the reference case or can solve the reference case. Figure 3(b) shows the related set to the reference case found in Figure 3(a). Further in stage two, the related set is assessed, and the case that is the most similar to the target problem, is retrieved. Figure 3(c) illustrates this last step.

Both footprint-based retrieval and MAC/FAC are two-stage approaches aiming to increase the retrieval speed, however they approach the problem in different ways [de Mántaras et al., 2005]. Footprint-based retrieval searches two subsets of cases and assesses their similarity based on a similarity measure. MAC/FAC, however, searches the whole case base with a computationally inexpensive process, before assessing the candidate set based on a more computationally expensive similarity measure. Which approach is the fastest depends on the case base, the case description, the similarity measures, the target problem, and the implementation of the approaches. However, both approaches can produce significantly reduced retrieval time compared to a sequential approach if implemented well. Furthermore, Smyth and McKenna [1999] have shown that footprint-based retrieval produces close to optimal results, i.e. the retrieved case is either the most similar to the target problem or close in similarity to the most similar. However, neither footprint-based retrieval nor MAC/FAC can guarantee retrieval completeness.

3.3 Index-based retrieval

Another two step retrieval approach is index-based retrieval. One way this retrieval approach differs from both MAC/FAC and footprint-based retrieval, is that its first step is executed during the retain step rather than the retrieval step of the CBR cycle [Aamodt and Plaza, 1994]. This is done in order to avoid computationally expensive processing during the retrieval step [Negny et al., 2010]. The first step of the index-based approach is a preprocessing step where an index structure is generated. This index structure is used to guide the search for the most similar case(s) in the second step of the approach – the retrieval step [Richter and Weber, 2013].

Most index-based methods are restricted to attribute-value representations, while MAC/FAC is applicable for both attribute-value representations and more complex case representations [Müller and Bergmann, 2014]. However, Müller and Bergmann [2014] have proposed an index-based retrieval approach that can be applied beyond pure attribute-value representations. This approach preprocesses the case base into a hierarchical cluster-tree, i.e. the case base is partitioned into sets of similar cases which are connected in a tree-like manner. This cluster-tree is used as the index structure during the retrieval stage, and traversing this cluster-tree allows one to find clusters of similar cases to the query while simultaneously reducing the number of similarity computations needed. Müller and Bergmann [2014] developed a clustering algorithm and a retrieval algorithm for this, and their approach was shown to require less development and maintenance effort than MAC/FAC. Furthermore, when it was compared to a MAC/FAC approach specified for semantic workflows, its retrieval quality and speed was close to that of the MAC/FAC approach when the case base had groups of similar cases. However, MAC/FAC was significantly better for case bases without a cluster structure, i.e. case bases that did not contain groups of similar cases.

3.4 Summary

There are many aspects to consider when choosing a retrieval method, including:

- The representation of the cases.
- The size of the case base.
- The similarity measure(s) to be used.
- The requirements for retrieval accuracy.
- The requirements for retrieval efficiency.

While sequential retrieval can guarantee retrieval completeness, it does so at the cost of retrieval efficiency, especially for larger case bases. MAC/FAC may give a higher retrieval speed than sequential retrieval, but implementing a MAC stage that does not disregard actually relevant cases and still produces a candidate set that is smaller than the case base, can be challenging. Footprint-based retrieval, however, has been shown to increase the retrieval efficiency, compared to sequential retrieval, while retrieving near optimal cases. Furthermore, Müller and Bergmann [2014] have proposed an index- and clustering-based approach that can rival MAC/FAC given domain restrictions and a case base with a clustered structure.

4 A Current Application: selfBACK

The selfBACK system is a predictive decision support system for self-management of lower-back problems. The system uses Case-Based Reasoning methods to retain and reuse patient cases to provide decision support on activity plans for lower back pain management and self-treatment [Bach et al., 2016]. The system, which is currently under development, is highly relevant in a world with an ever-growing sedentary work force. This section examines the selfBACK system, as well as the implementation of, and the reasoning for using, MAC/FAC.

4.1 Case Representation in selfBACK

The selfBACK case representation consists of the problem description, dually based on a subjective and an objective description of the patients health situation, and the solution, a recommendation for pain-relief and management. The objective description is a continuous stream of activity data obtained from a wristband worn by the patient. Each data point is classified into one of the four activity categories: sleep, sedentary, moderate and vigorous. The subjective portion is gathered from the patient's description of their own state, which includes demographic data, quality-of-life rating, pain level and functionality [Bach et al., 2016].

The subjective data is very simple to use in similarity measurements as simple similarity metrics can be used on the numerical and symbolic values in the description. However, the activity data is much more computationally expensive to compare, meaning that the comparison requires a relatively large amount of computational power. Therefore, the selfBACK system has a cheap, but broad similarity metric on the subjective data, and a more computationally expensive comparison on the activity data [Bach et al., 2016]. This description is one of the foundations of the MAC/FAC methodology, and MAC/FAC is a natural method to consider using for the retrieval stage.

4.2 MAC/FAC in selfBACK

The MAC/FAC approach divides similarity comparisons into two stages. The first stage is a broad, computationally cheap, stage to narrow the field of candidates, cases which are similar enough to be further compared. Subsequently, the second stage is a much more picky and computationally heavy stage to use on the narrowed field of candidates, in the aim of pushing through the prime candidates.

The MAC/FAC method is important in the case of selfBACK, as 99 % of the data is activity data [Bach et al., 2016], and would be used to compute only a few attribute similarities. Therefore, the MAC stage reduces the amount of cases to run the comparison on, greatly reducing the computational expense. In addition, when the MAC stage inputs are unchanged since last query, the MAC stage can reuse the previous retrieval. This is because the candidates for the FAC stage are chosen based on the weighting between the attributes which the similarity to a case is judged on, in the MAC and FAC stage.

In the version of MAC/FAC implemented in selfBACK, which is an altered version of the original MAC/FAC, a value resembling a score is evaluated for each case. During the comparison in the MAC stage, a score M_i is calculated from the similarity measure between the attributes of the case, and those of the target case. Similarly in the FAC stage, the similarity measure compares between the attributes designated to the FAC-stage to find the FAC score, F_i, with a different similarity measure, giving a total global similarity score of $M_i + F_i$. However, to avoid computing score for all cases, one must apply the discussed methodology to reduce the time and computational resources used.

Therefore, after the MAC stage, the candidates still viable are those that are still able to become the highest scoring when the score from the FAC stage is added. This would require the candidates to lag behind the case with the highest MAC score by at most the maximum attainable score of the FAC stage. The requirement is defined by Bach et al. [2016] as

$$M_i \geq M_{highest} - F_{max} \tag{1}$$

where M_i is the score of the case from the MAC stage, $M_{highest}$ is the highest score from the MAC stage, and F_{max} is the maximum attainable score of the FAC stage. A necessary prerequisite of this equation and the viability of using a MAC/FAC methodology is that

$$M_{max} \geq F_{max} \tag{2}$$

where M_{max} is the maximum attainable score of the MAC stage. Without this prerequisite, the MAC stage would never filter any cases, as by Equation (1) the requirement for passing the MAC hurdle would be to have a non-negative score, which would include all cases. However, with this prerequisite, this instantiation remedies one of the larger drawbacks of the method and ensures returning the case with the highest global similarity, rather than the case with the highest local similarity.

4.3 Summary

selfBACK is a predictive CBR-system for decision support in self-management and treatment of lower back pain. This system uses the MAC/FAC method on computationally cheap subjective metrics to filter out unsuitable case candidates, and then a subjective, computationally heavy, activity data comparison on the viable candidates to result in a single best-fit case and recommendation. Through specific heuristics for the MAC filtration, completeness is achieved, ensuring selection of the case with the largest global similarity to the query case.

5 Conclusions and Further Work

MAC/FAC was developed as an attempt at modelling the cognitive and mnemic behaviour seen in humans [Gentner and Forbus, 1991]. This led to a design based on two distinct comparative processes: A computationally cheap and superficial process, and a structural, computationally expensive process. This bisection of the retrieval process is especially efficient with large case bases where assessing similarity is cheap, but finding the case with the highest relevancy is hard. Although quite efficient in certain cases, it is inefficient in case bases of similar cases and when differentiating cases is hard. Adding to this, the general MAC/FAC implementation is not complete, meaning the selected case might not be the most similar to the query, which could incentivize to choose other approaches.

As discussed in Chapter 3, when considering retrieval methods there are several aspects to the field of applications to consider. These would yield varying best-approaches based on which aspects and qualities are key to the field of application. One such field is in the development of the selfBACK system. The case representation is bisected into the subjective part of description and attributes, and an objective, continuous stream of activity data. In such a case with a relatively clear split on computationally cheap data and the computationally expensive data, MAC/FAC is well-equipped. However, the implementation of MAC/FAC is slightly altered to ensure completeness through judging on global similarity.

A natural next step could be to consider a method combining the core concepts of footprint-based retrieval and MAC/FAC. An interesting topic would be using the footprint-based preprocessing step of finding competence groups as a preprocessing step for MAC/FAC. This would be used to find the competence group within which the target problem falls, and MAC/FAC would then be used on the set of cases within this group. It would be interesting to see if the potential gain of reducing MAC/FAC computational expense as a result of the preprocessing of the case base, is greater than the expense of employing the footprint-based preprocessing step.

Bibliography

A. Aamodt and E. Plaza. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, 7(1):39–59, 1994.

K. Bach, T. Szczepanski, A. Aamodt, O. E. Gundersen, and P. J. Mork. Case Representation and Similarity Assessment in the SELFBACK Decision Support system. *Case-Based Reasoning Research and Development Lecture Notes in Computer Science*, page 32–46, 2016. doi: 10.1007/978-3-319-47096-2_3.

P. Cunningham. A Taxonomy of Similarity Mechanisms for Case-Based Reasoning. *IEEE Transactions on Knowledge and Data Engineering*, 21(11):1532–1543, 2009.

R. L. de Mántaras, D. Bridge, and D. McSherry. Retrieval in Case-Based Reasoning: An Overview. *AI communications*, 10(1):21–29, 1997.

R. L. de Mántaras, D. McSherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M. L. Maher, M. T. Cox, K. Forbus, M. Keane, A. Aamodt, and I. Watson. Retrieval, reuse, revision, and retention in Case-Based Reasoning. *The Knowledge Engineering Review*, 20(3):215–240, 2005.

D. Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2):155 – 170, 1983. ISSN 0364-0213.

D. Gentner and K. Forbus. "MAC/FAC: A model of similarity-based retrieval". *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 504–509, 1991.

T. Mitchell. *Machine Learning*. McGraw Hill, 1997. ISBN 0071154671.

G. Müller and R. Bergmann. A cluster-based approach to improve similarity-based retrieval for process-oriented case-based reasoning. In *21st European Conference on Artificial Intelligence*, ECAI, 2014. doi: 10.13140/2.1.4346.0163.

S. Negny, H. Riesco, and J. M. Le Lann. Effective retrieval and new indexing method for case based reasoning: Application in chemical process design. *Engineering Applications of Artificial Intelligence*, 23(6):880–894, 2010.

M. M. Richter and R. O. Weber. *Case-Based Reasoning*. Springer, 2013.

S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2010.

B. Smyth and E. McKenna. Footprint-Based Retrieval. *Lecture Notes in Artificial Intelligence*, 1650:343–357, 1999.