University of Auckland

CS760 Datamining and Machine Learning

Assignment 1

Case-based reasoning - Travel Case

Author:
Joel Glemne

Student-id:
145076454

September 8, 2016

Abstract

Your abstract.

1 Introduction to CBR

This section is divided into three sections; one giving an everyday life example with the purpose of giving an intuitive understanding of case-based reasoning (**CBR**), one section with a more theoretically general description followed by a mathematical description of the global similarity metrics.

1.1 Intuitive approach

In your everyday life you meet many small problems that you probably would consider quite simple. Let us imagine that you are at your boyfriend's house for the first time, having dinner with his family. The mother asks you to go fetch one more plate. What do you do?

If it was me, I would first go to the kitchen. Then, I would start to look around if I see any *obvious* place for the plates to be (be sure to notice the word *obvious* for later). After that, I would start looking in the cabins. What is then funny, is that the possibility of me finding the right cabin on the first try would be high. How is that possible?

The reason why, is that we often base our decisions on previous *experiences* and the outcome of those experiences. I have been to a lot of kitchens before and it is common that different families - at least families from the same culture - organize their kitchens in similar ways. What I did, was simply evaluating (**reasoning**) my **base** of previous experiences (**cases**) in order to find a similar solution for the given problem.

1.2 General approach

CBR is mainly a technique for solving a large varieties of problems. Though, to fully understand the concept of CBR it is important to explain the meanings of *case-base* and *reasoning*.

A *case-base* is a collection of experienced solutions to similar problems. *Reasoning* is the procedure of drawing conclusions based on previous cases in order to solve a given problem. It is, however, important to note that it is not necessary to find an exact replica of the considered problem in order to find a solution, but only a *similar* problem. What is then important in the CBR methodology is how to decide what *similar* means.

In the example in section 1.1 the word *obvious* was used. What obvious means is basically that the given case problem is very similar to a, or several, previous experienced case(s). For humans, it might seem easy to compare different cases and identify all key *features* in an instant, but for a computer it is crucial to pre-identify key features, define and assign similarity metrics to each feature and then organize everything into a reusable model for comparing different cases and predict outcome.

1.3 Mathematical approach

The similarity between two cases is calculated according to the k-nearest neighbors algorithm explained here below.

Assume a case-base of n cases y_i where $i = 1, 2, \ldots, n$ and each case having the same m pre-defined features with different values. For each feature a *local* similarity function sim_j is defined where $j = 1, 2, \ldots, m$. The functions takes two arbitrary cases a, b as arguments so that $0 \leq sim_j(a, b) \leq 1$. Every feature is also assigned a weight ω_j which corresponds to the *local* simililarity's importance to the *global* similarity.

When comparing an arbitrary **target case** x with all the **source cases** in the assumed case-base, n *global* similarities S_i are produced where $0 \leq S_i \leq 1$. S_i is defined as

$$S_i = \frac{s_i}{W}$$

where

$$s_i = \sum_{1}^{j=m} (\omega_j * sim_j(x, y_i))$$

and

$$W = \sum_{1}^{j=m} \omega_j$$

In order to find the most similar cases in the case-base, the cases y_i corresponding to the highest global similarities S_i are chosen.

One of the most interesting things in this algorithm is how the local similarity functions sim_j are defined individually. This is what is handled in the next section of this report.

2 Features

In this section, all the local similarity functions for the given case-base are presented and motivated.

As described in section 1.3, every case has a couple of pre-defined *features* or *variables*. In this application the cases considered are different options for travel plans and they all have the following features:

1. Accommodation
2. Case name
3. Duration
4. Holiday type
5. Hotel
6. Journey code
7. Number of persons
8. Price
9. Region
10. Season
11. Transportation

Every feature has its own local similarity function, producing values within a range of $[0, 1]$, and an assigned weight within the range $[1, \infty[$. If a feature is considered important for the global similarity calculation, it is assigned a high value and vice versa.

2.1 Accommodation

The possible values for the accommodation feature were:

1. HolidayFlat
2. OneStar

3. TwoStars
4. ThreeStars
5. FourStars
6. FiveStars

2.1.1 Local similarity function

To calculate the local similarity of accommodation between two arbitrary cases, some sort of ranking was here needed. In this case, HolidayFlat was considered the least exclusive kind of accommodation and thereafter rising with number of stars. All of the feature values were therefore assigned an integer value I_{acc} corresponding to above given table numbers, e.g. for an arbitrary case a which has the feature value TwoStars, $I_{acc}(a) = 3$.

In order to produce a similarity sim_{acc} so that $0 \leq sim_{acc} \leq 1$, it was here assumed a *more-is-perfect* approach with a linear calculation of the similarity. This means, that if $I_{acc}(target\,case) \leq I_{acc}(source\,case)$, then $sim_{acc} = 1$. Otherwise, the similarity is calculated as

$$sim_{acc} = \frac{range - (I_{acc}(target\,case) - I_{acc}(source\,case))}{range}$$

where $range = 6 - 1 = 5$.

2.1.2 Feature weight

When considering a vacation option, the accommodation type is normally considered a feature which gives an extra bonus to the experience rather than being a feature vital for the selection. Therefore, the accommodation feature is given a relatively low weight of 3.

2.2 Case name

Since the case name by definition is unique for each case, e.g. *Journey984*, and really does not give any information about the case, this feature does

2.3 Sections

Use section and subsection commands to organize your document. LaTeX handles all the formatting and numbering automatically. Use ref and label commands for cross-references.

2.4 Comments

Comments can be added to the margins of the document using the todo command, as shown in the example on the right. You can also add inline comments too:

Here's a comment in the margin!

This is an inline comment.

2.5 Tables and Figures

Use the table and tabular commands for basic tables — see Table **??**, for example. You can upload a figure (JPEG, PNG or PDF) using the files menu. To include it in your document, use the includegraphics command as in the code for Figure **??** below.

2.6 Mathematics

LaTeX is great at typesetting mathematics. Let $X_1, X_2, \ldots, X_n$ be a sequence of independent and identically distributed random variables with $\mathrm{E}[X_i] = \mu$ and $\mathrm{Var}[X_i] = \sigma^2 < \infty$, and let

$$S_n = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n} \sum_{i}^{n} X_i$$

denote their mean. Then as n approaches infinity, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $\mathcal{N}(0, \sigma^2)$.

2.7 Lists

You can make lists with automatic numbering ...

1. Like this,
2. and like this.

. . . or bullet points . . .

- Like this,
- and like this.

We hope you find writeLaTeX useful, and please let us know if you have any feedback using the help menu above.

References

[1] Michael M. Richter & Rosina O. Weber, *Case-Based Reasoning*, Springer-Verlag Berlin Heidelberg, 2013.