

cuacs

Matching Algorithm Overview



**ARCLIGHT ENTERTAINMENT**

Brian grickites

SHALIN LATHIGRA

BRIAN LEBLANC

AARON RAMOS-LAZETTE

Submitted to:  
Dr. Christine Laurendeau  
COMP 3004 Object-Oriented Software Engineering  
School of Computer Science  
Carleton University

March 5, 2019

1. Re-Introduction to the context of the system
2. Changes made to system model (object model)
3. Purpose of algorithm & main considerations
4. Maps of attributes
5. High level overview of algorithm
6. Explanation of production system
7. Explanation of heuristic
8. Explanation of pruning mechanism
9. Glossary
10. Introduction
11. Object Model
12. Attributes
    1. List of Attributes
    2. Comparing Attributes
13. Matching Algorithm
    1. Candidate Filter
    2. Best-First Search Heuristic
    3. Production System
    4. Pruning System
14. Glossary
15. Introduction

… Animal Client Matching Algortihm (ACM)

**what’s best for the animal** \*if there is a choice between an animal being in an incompatible home or keeping the animal in a shelter, the algorithm will choose to keep the animal until a more suitable match can be found. The animal’s overall happiness and well-being is to be prioritized at all costs.

1. Object Model

Since the previous iteration of cuACS, some minor adjustments have been made to our system model, illustrated by the diagram below. The primary change involved merging our ClientTraits and Preferences into a single ClientProfile object, broadly defined as containing all Client attributes outside of basic contact info.

\*Image\*

**Figure** the current Object Model

1. Attributes

The following attributes

1. Matching Algorithm

The ACM seeks to provide an optimal set of matches by using the compatibility factor for candidates to find the greatest net sum over all of them. There are, however, a few exceptions to this rule that will be elaborated on momentarily. After using the above techniques to determine the compatibility factors of each candidate, they are manipulated by the following four processes to determine the greatest set of matches: Candidate Filter, Heuristics, Production System, and Pruning System.

* 1. Candidate Filter

The Candidate Filter seeks to discard unacceptable matches prior to the Production System performing the bulk of the work. In doing so, it reduces the number of combinations the system has to account for and, resultantly, the size of the graph reduces as well.

The filter takes the candidates and a threshold, say 30% for the purpose of this example. Any candidate found to have a compatibility score lower than this threshold of 30% will be immediately discarded and not considered throughout the rest of the algorithm. This helps to set a hard standard for matches, with the ultimate purpose of preventing Solutions that have a high compatibility average but contain many lackluster matches. This threshold may be adjusted as needed, with higher thresholds yeilding Solutions with lower statistical variance but increasing the chance that some Animals or Clients may be completely excluded from the Solution because their compatibility level was not tolerated by the filter.

* 1. Best-First Search Heuristic

This Heuristic is used to prioritize which Potential Solutions get evaluated first. It assumes that Potential Solutions that contain the most exceptionally compatible candidates are more likely to yeild a better set of matches. For this reason, these Potential Solution paths will be explored first while less desirable matches will be explored later on. This prioritization is what is known as a Best-First Search.

From here, the Production System handles the exploration of Potential Solutions.

* 1. Production System

The Production system reflects the present state of the search and outlines the rules to proceed to the next generation(s) of the system, each generation reflecting a unique state of the search tree.

At this point in the algorithm, any subset of the candidates that do not contain the same Animal more than once or the same Client more than once are considered to be Potential Solutions.

Using the Best-First-Search Heuristic describes aboved, the Production System will attempt to evaluate any Potential Solution that has not previously been considered along the search path. The Production System will, upon selecting the next most promising Candidate, pass on only Candidates which contain neither the selected Candidate’s Animal nor Client for the next generation of the Production System to evaluate.

* 1. Pruning System

Before the Production System may

\*purpose\*

\*Paired Attributes\* - Note non-physical attributes especially

Figure high level overview of algorithm

\*deep dive into specifics\*

\*describe filter\*

\*Heuristic\* - Prioritize the set of all candidates (remaining after fliter applied) by compatibility factor

\*Filter outdated \*

\*Production System\* - See, based on the current state of the system (given by the node state on the tree) if it should evaluate the subtree rooted at an unexplored node

\*\*\* Refer to FRs, NFRs

Matching Algorithm

1. Glossary

Provides a description of key terms used throughout the document.

|  |  |
| --- | --- |
| **Staff (Member)** | Shelter user with access to the Management account. |
| **Client** | Shelter user given a Client profile page by a Staff Member. |
| **Animal** | Shelter animal given an Animal profile page by a Staff Member. |
| **Animal-Client Match (“Match”)** | A pairing of a compatible Animal and Client suggested by the ACM Algorithm. A member of the Solution Set. |
| **Candidate** | A potential but unconfirmed match marked for consideration prior to the execution of the ACM Algorithm. |
| **ACM Algorithm** | Animal Client Matching Algorithm. Suggests a match Solution. This optimization for this solution is determined by a combination of the Search Heuristic, the Production System and the Pruning System. |
|  |  |
| **Solution (Set)** | An optimized list of compatible Matches generated by the ACM. |
| **Potential Solution** | A set of matches that the system has not yet determined to be optimal. |
| **Production System** | How the algorithm determines what subtrees to evaluate. |
| **(Search) Heuristic** | A strategy for arriving at the Solution quicker. |
| **Pruning System** | The system which decides whether a particular search path can be ignored. Ignoring a path guarantees that it could not have yeilded a solution better than any previous solutions found. |