Analogy by Alignment: On Structure Mapping and Similarity

Nick Hawes and John Kelleher
Common Sense Group, Media Lab Europe
{nick.hawes, john.kelleher}@medialabeurope.org

Abstract. This paper presents a novel domain independent algorithm for constructing analogies using relationship-based structure-mapping. This algorithm is used as a core component in a system that solves visual analogy IQ test problems.

1 Introduction

Forming an analogy involves mapping elements from a particular situation onto elements in a separate situation in a way that preserves the relationships between the elements in each situation [1]. In this paper, the representations that analogies are constructed between are referred to as *complexes*. An analogy is always constructed between two complexes: a *background* complex and a *foreground* complex. The background complex is viewed as a construct that is already present in memory, while the foreground complex represents new input to the system. The elements that make up a complex are referred to as *atoms*. A *relationship* between two atoms denotes how the atoms are linked in the complex. The set of atoms an atom has relationships with are referred to as its *peers*. Properties of atoms are called *features*. When an atom in one complex is analogous to an atom in another, there is there is a *mapping* between them. Constructing an analogy between two complexes involves finding a set of mappings between their constituent atoms. The process of mapping atoms across complexes *aligns* the atoms (and therefore the complexes). Due to this, we will refer to a set of mappings between atoms in two complexes (i.e. an analogy) as an *alignment*.[1]

Gentner and Markman [6] distinguish between feature-based alignment as similarity and relationship-based alignment as analogy. However, we see an algorithm that creates alignments using both relational and featural information as the most useful way of comparing two complexes. In such an algorithm, strong atom-to-atom mappings could be created using either method, then either method could be used to extend the alignment. As a stepping stone to this hybrid alignment algorithm, we have developed a domain-independent algorithm based purely on relationship-based alignments. Any desirable mappings that may not be determined by a purely relation-based approach can be passed into the algorithm and used as the starting point of the alignment (see $seed\text{-}alignments$ in Section 2.3). Although this preprocessing approach solves the practical considerations of combining relation-based and feature-based alignment in a single algorithm, we believe that more alignment problems could be solved by

[1] Not all computational approaches to analogy have defined analogy in this way (e.g. [2]). Approaches that do take a structure-mapping approach to analogy include: [3], [4], [5].

having the two alignment methods working in tandem in the algorithm. The rest of this paper will present this relationship-based structure-mapping algorithm and an application that combines our relationship-based alignment algorithm with feature-based structural alignment to solve visual analogy problems.

2 A Relationship-Based Structure-Mapping Algorithm

In order to find suitable alignment mappings across complexes, each atom from the foreground complex must be compared to each atom in the background complex. This comparison involves attempting to align the foreground atom's peer relationships with the background atom's peer relationships. A naïve approach to alignment would involve matching each of the relationships that the foreground atom features in with each of the relationships that the background atom features in. Consequently, the process of forming a complete alignment is prone to combinatorial explosions in terms of both atoms (due to the exhaustive atom-to-atom comparisons) and inter-atom relationships (as a greater number of relationships will require a greater number of comparisons during the structure alignment).

We assume that it is not necessary to make every possible comparison between the relationships present in the complexes; networks of relationships contain some redundancy, e.g. symmetric and transitive relationships. Following this, our algorithm aligns pairs of atoms based on the relationships common between pairs of atoms in each complex. Once aligned pairs of atoms have been added to the alignment they are not considered during the rest of the alignment process. This approach tames the complexity of the alignment problem by ignoring potentially redundant relationships.

Our structure-mapping algorithm uses *micro-relationships*. These are the most basic relationship possible between two atoms in a complex. The term was chosen to loosely parallel the meaning of *micro-feature* used in other literature on analogy and similarity (e.g. [7]). It is desirable to use micro-relationships because they represent the kind of information that can be easily extracted from original sources to form complexes and they encourage a greater degree of domain independence than higher order relationships. We augment the concept of a micro-relationship by associating a fidelity measure with each micro-relationship. The fidelity measure represents how well the micro-relationship between two atoms matches the prototype of that relationship. Fidelity measures range from zero (low fidelity) to one (high fidelity). They allow us to perform more detailed comparisons between pairs of related atoms: related pairs with similar fidelity measures are deemed to have stronger mappings between their atoms than related pairs of atoms with less similar fidelity measures.

2.1 Informal example

To ground the presentation of our algorithm, we present an informal example of its operation in an image domain. In this domain, the atoms in a complex represent the blobs extracted from an image. The micro-relationships in the domain are *left*, *right*, *above*, *below*, *in-front*, *behind*, *enclosing*, *enclosed*, *bigger* and *smaller*. In this domain, the concept of micro-relationship fidelity is used to represent how well a micro-relationship encodes its describing preposition. For example, given two blobs B_1 and B_2 related by a *left* micro-relationship, if B_2 is precisely to the left of B_1 then the *left* micro-relationship will have a fidelity measure of one, otherwise the fidelity measure will be lower.

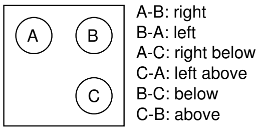

Figure 1: Background Image and Complex

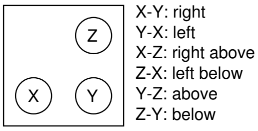

Figure 2: Foreground Image and Complex

Our approach to analogy involves identifying the structural roles played by atoms in the background and foreground complexes and then determining the similarity of these roles. If the roles are similar, then the atoms are aligned by creating a mapping between them. The role played by an atom in a complex is specified by its micro-relationships to other atoms in the complex. For example, if we are trying to align the complex in Figure 1 with the complex in Figure 2 we can start with the background complex pair **A-B** and look for an analogous pair in the foreground complex. **A-B** consists solely of the micro-relationship *right* and is matched exactly by the foreground pair **X-Y**. It is also partially matched by **X-Z**, so this mapping should also be considered at some point in the construction of the alignment. Taking the exact match **X-Y** gives us alignments of **A** to **X** and **B** to **Y**. To extend this alignment we need to first look for an atom from the background complex that is related to the background atoms in the alignment. Because atom **C** is the only remaining unmapped atom, we will pick this for the extension of the alignment. If there were other atoms related to **A** or **B** in the complex then they would also be possible choices to extend the alignment. Atom **C** is related to both atoms in the background half of the current alignment, but for the example we will choose to use the set of micro-relationships **A-C** to relate it to current alignment (in practise, **B-C** should also be examined). The foreground complex must be searched for a matching pair of atoms that adds a new atom to the alignment. The only matching set of micro-relationships that adds a new atom to the alignment while also being related to the current atoms in the foreground half of the alignment is **X-Z** (matched because of a common *right* micro-relationship). The alignment cannot be extended any further because there are no more free atoms in either complex, so our final alignment mappings are **A** to **X**, **B** to **Y** and **C** to **Z**.

In the above example, it should be noted that there are intra-complex micro-relationships that are not checked against the corresponding micro-relationships in the opposing complex. For example, after **C** is added to the alignment based on its micro-relationships with **A**, its micro-relationships with **B** are never considered. Neither are the corresponding micro-relationships in the foreground complex (those between **Z** and **Y**). This reduces the complexity required to exhaustively check every set of micro-relationships each newly introduced atom is involved in. Other ways of adding atoms to the alignment are considered by branching the search when more than one match is found for a particular set of micro-relationships in the other complex (e.g. when **A-B** can match with either **X-Y** or **X-Z**). Each alternative way of introducing an atom to the alignment is ascribed a strength (as discussed in Section 2.2), enabling alternative alignments (containing identical inter-complex mappings or different inter-complex mappings) to be compared based on their composition.

2.2 Pair alignment and strength

Our approach to analogy allows any alignment between two pairs of related atoms to be ascribed a value representing the strength of the alignment. This strength is calculated by determining the percentage of micro-relationships that are common between the pairs of atoms being mapped in both complexes. This is represented in Equation 1:

$$s(r_b, r_f) = \frac{|M_s|}{|M_t|} \tag{1}$$

where s returns the strength value (between zero and one), r_b is a related pair of atoms from the background complex, r_f is a related pair of atoms from the foreground complex, M_s is the set of micro-relationships common across the background and foreground pairs, and M_t is the union of the sets of micro-relationships between the atoms in the background and foreground pairs. The quality of a complete structural alignment is derived by summing the strength of its pair alignments. This method of calculating the strength of an alignment is extended to include fidelity measures associated with micro-relationships by comparing how similar the fidelity of the shared micro-relationships are. This is shown in Equation 2:

$$s(r_b, r_f) = \frac{\sum_{m \in M_s} 1 - abs(fidelity(m, r_b) - fidelity(m, r_f))}{|M_t|} \tag{2}$$

where $fidelity$ returns the fidelity measure of a particular micro-relationship, and the other terms are identical to those used in Equation 1.

2.3 Algorithm

In this section we present a domain independent formal definition of our relationship-based structure-mapping algorithm. The algorithm takes two complexes as input and returns a list of alignments, where each alignment is a list of mappings between an atom in the background complex and an atom in the foreground complex. The algorithm involves the following terms: B the set of atoms in the background complex; F the set of atoms in the foreground complex; $s((b_i, b_j), (f_p, f_q))$ see Equations 1 and 2, where $b_i, b_j \in B$ and $f_p, f_q \in F$; $align(b, f)$ creates an alignment mapping between b and f, where $b \in B$ and $f \in F$; $background(map)$ returns b of alignment mapping map; $foreground(map)$ returns f of alignment mapping map; $background\text{-}atoms(alignment)$ returns the set of background atoms involved in all of the mappings in $alignment$; $foreground\text{-}atoms(alignment)$ returns the set of foreground atoms involved in all of the mappings in $alignment$; $expand\text{-}alignment(alignment)$ expands the alignment $alignment$ by every valid mapping and returns the set of all possible expanded alignments, see below for details; $seed\text{-}alignments(B, F)$ returns a list of single-mapping alignments that can be computed using either an exhaustive pair mapping between atoms in the input complexes, or a preprocessing step (e.g. feature alignment or the application of domain specific knowledge).

The search framework used by the structure-alignment algorithm is as follows:

```
Let $alignments$ = {}, $O = seed\text{-}alignments(B, F)$
**while** $O \neq \{\}$
    $alignment_l = head(O)$
    $O = tail(O)$
    $E = expand\text{-}alignment(alignment_l)$
    insert each $e \in E$ into $O$
    insert each $e \in E$ to $alignments$
**return** $alignments$
```

Once the search is complete, the resulting alignments have their strength calculated using the strength measure s. This strength measure could also be retrieved from the values calculated during the execution of $expand\text{-}alignments$. The list of alignments is then sorted by strength to find the best alignment.

The function $expand\text{-}alignment$ represents the core processing in the structure-alignment algorithm. It's input, $alignment$, is a list of mappings. This list is expanded once by each valid pair alignment, to return a list of successor alignments. A valid expansion is the addition of any new pair alignment that has a non-zero strength (i.e. the pairs share at least one micro-relationship). $Expand\text{-}alignment$ is defined as follows:

```
forall $mp \in alignment$
    $m_b = background(mp)$
    $m_f = foreground(mp)$
    forall $a_b \in B - (m_b \cup background\text{-}atoms(alignment))$
        forall $a_f \in F - (m_f \cup foreground\text{-}atoms(alignment))$
            if $s((m_b, a_b), (m_f, a_f)) > 0$
              $alignment\prime = alignment + align(a_b, a_f)$
              $alignments = alignments + alignment\prime$
return $alignments$
```

3 Example Application: Visual Analogy Problem Solver

To demonstrate the processing of our algorithm we will use the IQ test visual analogy problem domain that Evans used for his analogy research [8]. An example problem can be seen in Figure 3. The idea of such problems is to look at the way in which the shapes in image **A** have been rearranged to create image **B** and then pick which of the target images **D** to **G** best reflect the same rearrangement being applied to image **C**. In Figure 3 the answer is image **E**; the shape that was enclosed in the first image is placed to the left of the previously enclosing shape and the extra shape is removed.

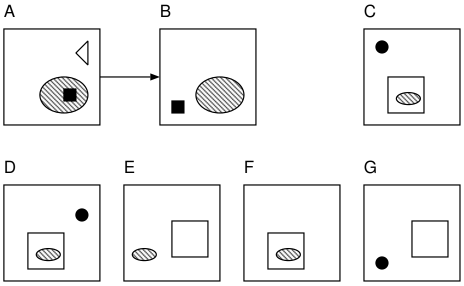

Figure 3: An example of a visual analogy problem

Our approach to solving these problems involves discrete steps of feature-based and relationship-based alignment. First, complexes representing the images are derived using the domain definitions used by the example in Section 2.1. Complexes **A** and **B** are then aligned by atom features. This gives us an identity mapping from the atoms in **A** to the atoms in **B**, allowing us to determine how the original atoms are related in "transformed" complex **B**. We then align complexes **A** and **C** based on atom relationships using the relationship-based structure alignment algorithm presented in Section 2.3. The resulting mappings define which atoms from complex **C** should be taken as representing which atoms from complex **A**. In the example given in Figure 3, the enclosed black square in **A** is mapped on to the enclosed striped ellipse in **C**, the striped ellipse in **A** is mapped to white square in **C**, and the triangle in **A** is mapped to the black circle in **C**. Next, complex **C** is aligned with each of the possible solution complexes based on atom features. This feature-based alignment tells the program which atoms from **C** are in each possible solution complex. Finally, a constrained relational alignment between complex **B** and each of the target complexes is performed. This alignment is constrained by the fact that the only alignments allowed are the ones consistent with both the **A** to **B** and the **C** to target feature alignments, and the **A** to **C** relationship alignment. In the example problem in Figure 3 this means that only alignments mapping the large striped ellipse onto the white square and the black square onto the small striped ellipse are allowed (ruling out the potentially confusing answer of image **G**). To find a solution, the program uses the strength measurement presented in Section 2.2 to pick the image that produced the target complex with the best structural alignment to complex **B**.

4 Conclusion

In this paper, we presented a domain-independent relationship-based structure-mapping algorithm. The algorithm aligns complexes based on pairs of atoms. This approach reduces the complexity inherent in structure alignment by ignoring potentially redundant micro-relationships. The algorithm judges the quality of an alignment using the ratio of shared to non-shared micro-relationships between pairs of aligned atoms and fidelity measures which provide a measurement of how similar micro-relationships are. We combined this relationship-based alignment with a simple feature-based alignment step to create an application that solves visual analogy problems through the interplay of these methods.

References

[1] D. Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7:155–170, 1983.

[2] M. Mitchell. Analogy-making as a complex adaptive system. In Irun Cohen and Lee Segel, editors, *Design Principles for the Immune System and Other Distributed Autonomous Systems*, pages 335–360. Oxford University Press, New York, 2001.

[3] B. Falkenhainer, K. Forbus, and D. Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1):1–63, 1989.

[4] D. Gentner and K.D. Forbus. MAC/FAC: a model of similarity-based retrieval. In *Proceedings of the 13th Cognitive Science Conference*, pages 504–509, Chicago, 1991. Erlbaum, Hillsdale.

[5] R.L. Goldstone. Hanging together: A connectionist model of similarity. In Jonathan Grainger and Arthur M. Jacobs, editors, *Localist Connectionist Approaches to Human Cognition*, pages 283–325. Lawrence Erlbaum Associates, Mahwah, NJ, 1998.

[6] D. Gentner and A.B. Markman. Similarity mapping in analogy and similarity. *American Psychologist*, 52(1):45–56, 1997.

[7] R. Sun. A microfeature based approach towards metaphor interpretation. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95*, pages 424–431, Montreal, Quebec, Canada, 1995. Morgan Kaufmann.

[8] T.G. Evans. A program for the solution of a class of geometric-analogy intelligence-test questions. In Marvin Minsky, editor, *Semantic Information Processing*, pages 271–353. MIT Press, Cambridge, MA, 1968.