Discovering the Dimension of Abstractness: Structure-Based Model that Learns New Categories and Categorizes on Different Levels of Abstraction

Georgi I. Petkov, Ivan I. Vankov, Yolina A. Petrova

***Abstract*—**A structure-based model of category learning and categorization at different levels of abstraction is presented. The model compares different structures and expresses their similarity implicitly in the forms of mappings. Based on this similarity, the model can categorize different targets either as members of categories that it already has or creates new categories. The model is novel using two threshold parameters to evaluate the structural correspondence. If the similarity between two structures exceeds the higher threshold, a new sub-ordinate category is created. Vice versa, if the similarity does not exceed the higher threshold but does the lower one, the model creates a new category on higher level of abstraction.

***Keywords*—**Analogy-making, categorization, learning of categories, abstraction, hierarchical structure.

I. Introduction

We can say without hesitation that the ability to abstract is one of the most important characteristics of human thinking. The puppies are dogs; the dogs are mammals; they in turn are animals; who are living creatures, etc. There are plenty of evidences suggesting that the human mental representations are hierarchically organized along the dimension of abstractness. For example, the seminal studies of Collins and Quillian [1] showed that people can recognize faster that a canary is yellow (characteristic of the canaries) than that it flies (birds' characteristic), which in turn is faster than recognizing that canaries eat (characteristic of all animals). Collins and Quillian also proposed the idea of cognitive economy, stating that it is better to represent only properties specific to a level of abstractness, rather than all properties for every single concept (for more specifications see [2]). Representing knowledge at various levels of abstraction is not only economical, but also essential for reasoning. By hierarchically representing our concepts, we can infer information which is currently not present in the environment. For example, by knowing that something is a bird, we may conclude that it flies, breathes, etc. The necessity of tree-like structured representations is emphasized by researchers (for example, see Tenenbaum et al. [3]) to such an extent that it became one of the indisputable assumptions in the cognitive science.

Georgi Petkov, Ivan Vankov and Yolina Petrova are with the East and Central European Center of Cognitive Science; Department of Cognitive Science and Psychology, New Bulgarian University, 21 Montevideo Street, Sofia 1618, Bulgaria (e-mail: gpetkov@nbu.cogs.bg, i.i.vankov@gmail.com, yolina.petrovaa@gmail.com).

Also, there is an agreement that, to categorize, we basically compare the representation of the uncategorized item with various categories that we already have. However, there is a disagreement about the nature of this comparison – whether it is a kind of vector multiplication (or isomorphic mathematical operation) or it is a structural comparison.

II. Abstraction of Common Properties

Recently, deep neural networks using variations of the back-propagation training algorithm [4] have achieved impressive performances in tasks such as object recognition [5], face recognition [6], and speech [7]. Some deep neural networks have also been trained on more abstract categories, such as places [8] and semantic relations [9]. However, these are all cases of supervised learning, in which the list of categories is predefined and supplied by the model's designer. The models cannot categorize on different levels of abstraction by themselves. For example, they cannot categorize a puppy called Zara simultaneously as a dog, as a mammal, and as an animal. Even less they can recognize the difference between the categories at different levels of abstractness.

III. Abstraction of Common Relational Structures

Models such as SEQL [10] are focused on the relational structures instead of the properties. SEQL is a category learning model, which represents categories through structural descriptions. When the model is presented with a new input, by using the structure mapping engine [11], it tries to find the best possible structural mapping between what it has on the input and the generalizations stored in its long-term memory. If the structural mapping between the input and the representation of a certain category is good enough, then the input is categorized as belonging to this category. Otherwise, SEQL can either create a new category (if the relational mapping between the input and a single stored exemplar is good enough), or it just stores the input representation as a single uncategorized exemplar. However, which mapping is "good enough" depends on a pre-defined parameter of the model. If this threshold is relatively low, the model is more prone to generalize in abstract categories, whereas if the threshold is high, it either prefers concrete categories or does not categorize at all. Yet, SEQL has a problem – it cannot categorize simultaneously at different levels of abstraction without changing its threshold parameter first. A specific value of the parameter will make SEQL categorize the puppy Zara as a dog; another value will make the model categorize it as a mammal; and a third one – as an animal. However, a single run of SEQL cannot classify Zara both as a dog, as a mammal, and as an animal at the same time.

IV. Model of Category Learning and Categorization at Different Levels of Abstractness

To sum up, the *feature-based* models (such as the neural networks) are very good at representing categories with intrinsic properties, whereas the *structure-based* models represent better categories that are defined by their extrinsic relations. For example, to define the category *predator*, one needs information about its relations to other categories (like the information that it *hunts other animals*).

The cognitive architecture *DUAL* [12] adopts a structure-based approach to categorization, similarly to *SEQL*. *DUAL* takes structural descriptions on its input and through structurally aligning it to the relevant memorized bases, it tries to find the most similar (analogous) one. A mechanism called *spreading of activation* determines which knowledge is relevant. During the process of structural comparison, various mappings, capturing the episodes' commonalities, emerge. Mappings can be created in two ways – through the *marker passing* mechanism which finds semantically similar elements; and through the *structural correspondence* mechanism, finding structural commonalities. All contradicting mappings inhibit each other, whereas the coherent ones support each other. Described like that, *DUAL* is comparable to other analogy-making models – it takes a single target episode and searches for its best structural analog among its base episodes. However, it combines two essential properties: first, it has an activation spreading based retrieval, which allows the model to be simultaneously efficient (potentially, it could search any path) and effective (in fact, it deals with the most promising for the current context paths). Second, all mechanisms work locally and overlap in time, making the global behavior of analogy-making to emerge from countless local interactions.

The *RecMap* model for high-level [13], built on top of *DUAL*, could classify some inputs as predefined categories, and potentially, it can simultaneously create the hypotheses that Zara is a *dog*, a *mammal*, and an *animal*. However, the *RecMap's* problem is that its knowledge, including the links constructing the abstraction dimension, is predefined in advance. Various other categorization and category learning models, based on structural mapping [14]-[16], share this problem. Thus, our long-term project is to create a categorization and category learning model, that accounts for both intrinsically and extrinsically defined categories. The model should also account for one-shot learning and for context dependent categorization as well (for example, a *cat* should be recognized as a *mammal* in a certain context, and as a *predator* in another).

Following this goal, we extended the cognitive architecture *DUAL* with a new mechanism that transforms important mappings into concepts. That is a mechanism for automatic creation of new categories, if the activation of a given mapping exceeds a certain threshold. Thus, the conceptual system and the whole dimension of abstractness emerges as a result from the natural processes of reasoning, without it being a separate process, running in a special learning regime.

We also developed a simple *anticipatory mechanism* for categorization. If a certain target element is mapped to a base one, then the mechanism checks whether the target element is part of a categorized object. If it is not, the model creates an anticipation, by taking the category of the object that involves the base element. Simply saying, if, for example, a certain *tail* in the target is mapped to a *tail* that is part of a *cat* from the base episode, then the model creates an anticipation that the target's *tail* is also part of a *cat*. In contrast, the missing mappings inhibit the corresponding anticipations. In the other words, if there are *horns* on the target and these *horns* do not map to anything in the base description of the anticipated *cat*, then these *horns* will inhibit the *cat* anticipation, meaning that it is unlikely to categorize the target as a *cat*.

The model can deal with relation-based categories; however, in this paper we report only the initial simulations, all based on feature-based representations only.

A. Recognition at Different Levels of Abstraction

If the anticipation activation exceeds a certain threshold, the anticipation is transformed into a real instance of the anticipated category. Importantly, we added a second lower threshold. If an anticipation does not exceed the first threshold, but it exceeds the second one, then the anticipation forms a new category, but on a higher level of abstraction. In the other words, if something has much in common with a *cat*, but it is not similar enough to it, then it will be categorized as something more abstract than a *cat* (a *mammal*, for example).

The idea of using two different parameters could be employed in other categorization models as well. Let us take *SEQL*, for example [10]. If it uses a certain threshold for evaluating the mapping's quality, it will be able to categorize at the level of a *dog* or a *cat*. If the threshold is lower, it will categorize at the level of *mammal*. To be able to make those categorizations simultaneously, we could use two instances of *SEQL*, working at the same time with different parameters. The first instance of *SEQL* would categorize the target as a *mammal*, the second *SEQL* as a *cat*. Additional procedure could add an *is_a* link between the new categories.

The presented simulations with *DUAL* produce similar results. However, instead of using two models with different parameters, we use a single model. The model employs different pressures into a constraint satisfaction network, so categorization and category learning emerge from its local operations.

B. Constraint Satisfaction Network

As we already mentioned, the main operations of the model are creating of mappings and anticipations, which are interconnected with inhibitory and supporting links. The network comprising these mappings and anticipations is a constraint satisfaction network, reflecting various assumed pressures.

Every mapping creates an anticipation and supports it. Thus, the supporting links from the mappings to the anticipations reflect the pressure to categorize similar things as belonging to one and the same category. Even though the *DUAL'*s processes work locally, if the anticipations come from one and the same structure, the *anticipation mechanism* combines them into a single anticipation. Thus, for each reasonable category there is only one anticipation and all appropriate mappings support that anticipation. On the other side, the elements of the target that are not mapped to anything in the anticipated base inhibit this anticipation. This reflects the pressure not to recognize something as a member of a category, if it is too different from the category members.

In addition, there are inhibitory connections among the competing mappings, reflecting the famous one-to-one mapping pressure [17], i.e. one thing in the target cannot correspond to two different things from the base. There are also supporting links among the mappings from one and the same structure. For example, the mapping of a certain relation supports the mappings between its respective arguments and vice versa.

Finally, lower level anticipations support the anticipations from the higher abstraction level. For example, the anticipation to categorize something as a *cat* will support the anticipation to categorize the same thing as a *mammal*. In turn, the competing anticipations inhibit each other – one cannot be recognized both as a *cat* and as a *dog*.

It should be mentioned that all these restrictions are not necessary, they are simply pressures. Each one of them could be violated if the other pressures overbear it. Thus, all pressures form a constraint satisfaction network, from the relaxation of which the behavior of the model emerges. If after a fixed amount of time no mappings or anticipations exceed the respective thresholds, the lower threshold is considered.

There is something which may seem to be paradoxical on a first glance: on one side, every mapping strives to win and to be transformed into a concept. On the other side, every mapping supports a certain anticipation. These are two opposing tendencies. If the first one wins, the model will create a new concept. If the second one wins, the model will categorize the target as an existing category. Thus, the model's behavior depends on who will be the first to exceed its corresponding threshold. Importantly, there could be inconsistent mappings inhibiting each other, while together supporting one and the same anticipation. This would be the case when one target element is mapped to several base elements part of a single category. On the contrary, when there are target elements with no correspondences in the base, those elements can inhibit a given anticipation, while it will not influence the existing mappings. Therefore, more competing mappings will mean more similarity with elements part of the same category. Thus, the anticipations will prevail. More elements with no correspondences will inhibit the anticipations, meaning that the creation of a new category would gain strength.

C. Summary of the Model's Mechanisms

The presented model is part of a whole architecture that covers lots of cognitive abilities, exploring the hypothesis that few basic mechanisms (usually assumed as the analogy-making sub-processes) underlie broad range of cognitive functions. Most of the *DUAL* based model's behavior emerges from local interactions only. However, the model's ability to create categories and to categorize could be summarized in the following way.

When a description of a certain entity is presented on the target, its structure is compared with the structures of various memorized base entities. If the target is very similar to a single base and differs enough from the others, a new category is created on a very low level of abstractness – it combines only the target and its analogous base. However, if there are many exemplars that match the target and they are all members of the same category, the target is categorized as a member of that category.

The presence of a second threshold for transforming anticipations into categories allows the creation of more abstract categories. When the target is similar to the members of a certain category, but it is not similar enough, the model creates a new category on a higher level of abstractness and the target is categorized as a member of this category.

Finally, it may happen that the similarity between the target and the members' higher level categories is also not enough for the target to be categorized as something known. In that case, the model creates a category into an even higher level of abstractness.

A more detailed review of all these variants can be seen in the next section.

V. Simulating Various Categorizations and Creation of New Categories

To describe the model's mechanisms in work, we designed a small knowledge base. It allowed us to present various situations on the target, exploring different behaviors of the model (Fig. 1).

The bases *b1*, *b2*, and *b3* are encoded as consisting of five properties each, interrelated with specific relations (the relations are necessary to combine anticipations. If two features are not interconnected with relations, the model may assume that they belong to different objects members of the same category. However, this is not of importance for the current description of the model's work, thus, this specificity is omitted later in the text). Let us name these properties with mnemonic labels: *b1* consists of the features *barks*, *hunts rabbits*, *milks*, *breathes*, and *grows*; *b2 – barks*, *plays*, *milks*, *breathes*, and *grows*; *b3 – crows*, *lies on rocks*, *hatch eggs*, *breathes*, and *grows*. One may think of *b1* as a description of *beagle*; *b2* as another type of *dog*; *b3* as a *lizard*. We can assume that the model has already created two concepts: *c1* for a *dog*, and the more abstract one *c2* for an animal.

If the target object consisted of the properties *barks* and *hunts rabbits* (Fig. 1 (b)), then the mapping between *t* and *b1* became so strong that it won and by the mechanism of transforming mappings into concepts, a new concept was created on lower level of abstractness than *dog*. This was the *beagle* concept. In fact, the mappings themselves were between the properties *barks* and *hunts rabbits* from the target and the bases *b1* and *b2*. The model kept these mappings as permanent concepts and, in addition, created a binding node, thus, it bound the properties into a single object – *beagle*. The concept *beagle* was a sub-class of *dog* and its properties were *barks* and *hunts rabbits*. However, when the target represented properties such as *barks* and *milks* (Fig. 1 (c)), the mapping between *t* and *b1* could not win so easily because it was inhibited by the mapping between *t* and *b2*. However, nothing changed with the anticipation to categorize the target as a *dog*. All mappings between *t* and *b1* and between *t* and *b2* supported the anticipation in the same way like in the first case. Thus, the anticipation won before any of the mappings. This was not because the anticipation itself became stronger, but because the mappings became weaker. As a result, the model categorized the target as a *dog*, without creating any new concepts.

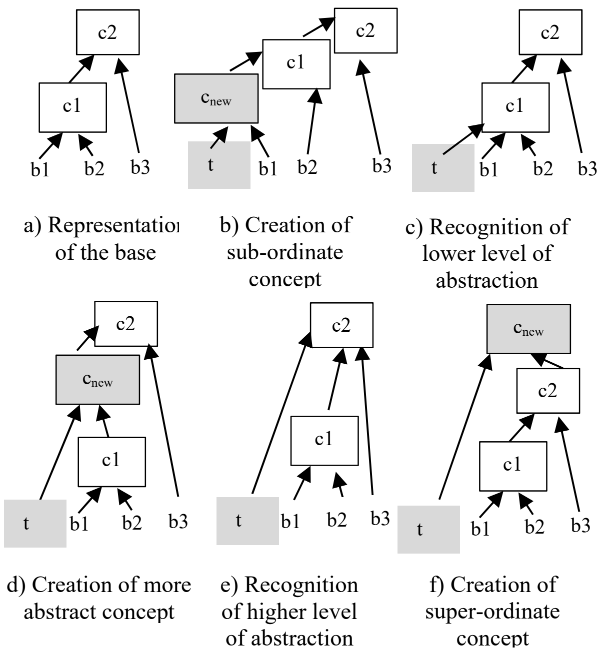

Fig. 1 The figure presents how different targets matched to the same base produce different results depending on the level of similarity between the target (t) and the exemplars in the memory (b1, b2, b3): a) the initial knowledge base; b) the target *t* is very similar to *b1,* less similar to *b2* and least similar to *b3*; c) the target *t* is moderately similar both to *b1* and *b2*, but not as much as to *b3*; d) *t* is similar to *b1* and *b2* but not enough for the anticipation to exceed the first categorization threshold; e) *t* is relatively equally similar to *b1*, *b2*, and *b3*; f) *t* is similar to *b1*, *b2*, and *b3* but not enough so the anticipations can exceed the first categorization threshold. C_{new} always denotes to a newly created concept

On Fig. 1 (d), a target was presented which had the features *milks* and *breathes*. In that case, the target created mappings with all *b1*, *b2*, and *b3*. These controversial mappings could not win over each other so easy. In this case, the model created two different anticipations – to categorize the target as a *dog* and to categorize it as an *animal*. The first one received as much support from the mappings with *b1* and *b2* as in the previous example. However, now the *dog* anticipation received an inhibition from its rival. The result was that this anticipation did not exceed the first threshold and the target could not be categorized as a *dog*. However, it exceeded the second threshold. Because of that, the model created a new category on a higher level of abstraction, which could be called *mammal*.

On the other hand, we also presented a target represented through properties like *breathes* and *grows* (Fig. 1 (e)). That means that it corresponded equally good to all three bases and two anticipations were created. In comparison with the previous example, however, the second anticipation (to recognize the target as an *animal*) was much stronger. It received support from the mappings between the target and *b3* but also from its rival anticipation. That is a mechanism according to which the anticipations from the lower-level of abstraction support the higher-level anticipations. In the other words, if something is a justification to categorize the target as a *dog*, it should be a justification to categorize it as an *animal* as well. As a result, the more abstract anticipation won the competition – the model categorized the target as an *animal*.

In the final example (Fig. 1 (f)), the target consisted of the properties *breathes* and *photosynthesizes*. Everything was almost the same as in the previous example – the mappings were equally good and inhibited each other; the higher-level anticipation received more support than the lower-level one. However, the difference was in the amount of the support the *animal* anticipation received. It came only from one of the properties (*breaths*) and was inhibited from the other one (*photosynthesizes*). As a result, this anticipation won but it did not exceed the first threshold. It only exceeded the smaller threshold and by the same mechanism as in Fig. 1 (d), a more abstract concept was created, for example – *living thing*.

It is important to stress that this is a very simplified description of the main idea behind the model. Because the model is implemented in the *DUAL* cognitive architecture, it shares its strengths. It can deal with structural descriptions and various context influences. The big strength of the model is that it can find deep relational structures and match analogous episodes without them being superficially similar. Because of that, the model can capture not only feature-based categories, but relational categories (defined by extrinsic relations with other categories) as well.

VI. Simulations

To test the stability of the model, we encoded several DUAL-based representations of various types of animals. We gave those representations as targets sequentially and in different order. For example, a representation of a *cat* was manually encoded. It consisted of about 10 agents like *head_1*, *tail_1*, as well as specific relations among them. The features themselves, as well as the relations among the features, pointed with *is_a* links to the specific concepts, also manually encoded in *DUAL*'s long-term memory.

During the first run of the model, we gave it a *cat* representation as target. There were no other bases in the memory and *DUAL* just memorized the target. Then, a second target was encoded. It was assumed to represent a *dog*. The target again consisted of about 10 agents (as the memorized *cat*). The *dog* and the *cat* shared about half of their features and the relations among them. Importantly, when presented to the model, those descriptions are not combined through a single node.

The model retrieved most of the *cat*'s description relatively easy and created some mappings between the target and the base. These mappings did not have any competitors and were transformed into concepts (for example, a concept that involved the *cat*'s and the *dog*'s *heads* was created, pointing to the superior concept *head*). In addition, few new agents were created – binding nodes for the *cat* and for the *dog*'s descriptions, and a concept (let us call it *mammal*) involving both. Now, the model already had some concepts in its long-term memory as well as representation of a *cat*, and representation of a *dog*, On the next simulation, we gave as target another description of a *cat*, very similar to the first one – it again consisted of about 10 agents like *head_2*, *tail_2*, etc.

The retrieval process activated various concepts as well as agents from the memorized descriptions of the *dog* and the *cat*. Again, many mappings emerged but this time there were some inhibitions as well. For example, the mappings between the target *head_2* and the *head* of the memorized *cat* and the mapping between *head_2* and *head_1* (coming from the memorized *dog*) competed with each other. However, all mappings supported the just created anticipation to categorize the target as a *mammal*. Several pressures competed influencing the constraint satisfaction network: from one side, almost all mappings supported the *mammal* anticipation (*mammal* was a binding node, combining all target elements). From another side, the mappings between the target and the *cat* base supported each other, as well as the mappings between the target and the *dog* base. From third side, these two sets of mappings formed two coalitions that inhibited each other. Finally, the constraint satisfaction network was also influenced by the relevance of the respective elements (features, relations, etc.) that were involved in these mappings. However, the final pressure is not important for the current simulation, we kept the relevance of the target elements constant. This pressure will be more important for simulating various context effects.

The *mammal* anticipation needed time to win and to be transformed into a real instance-agent (i.e. the target to be recognized as a *mammal*). If other processes did not interrupt, the model would have already categorized the target *cat* as a *mammal*. However, the coalition of mappings between the target and the base *cat* was so strong that succeeded to inhibit their competitors (coming from the base *dog*). Thus, the mechanism transforming mappings into concepts won before the anticipation. The *cats' heads, tails*, etc. were combined in new concepts, all they bounded by another new concept – *cat*. After the new lower-level category of *cat* was created, the simulation stopped.

Note, in a certain context, this may not have happened. For example, if the base *dog* was more active (because of priming, recency, etc.), the target would have been categorized as a *mammal* and the *cat* category would not have been created.

Finally, we presented a *lizard* description on the input and the model created a concept on higher-level of abstraction – we called that concept an *animal*.

After resetting the model, we sequentially introduced the same targets but in different order – *cat*, *cat*, *dog*, *lizard*. Then – *cat*, *lizard*, *cat*, *dog*, and finally – *cat*, *lizard*, *canary*, *dog*. In all cases, the model created one, and the same tree of hierarchically structured concepts and their instances.

VII. Conclusion

A structure-based model for category learning and categorization on different levels of abstraction was presented. The model is based on the cognitive architecture *DUAL* and works with structural descriptions of entities.

One of the novelties of the model is that it uses two threshold parameters to evaluate the structures' similarity. The very simple simulations, that were presented, highlight the basic idea behind the model – it compares the similarity between structures and based on this similarity, it either categorizes the target as something known, or it creates a new category.

The model has more potential than the presented one. It challenges the modern deep neural networks with its capabilities for single-shot learning. Potentially, it could deal with relational categories and various contextual influences.

Yet, it has its limitations. We have several important goals ahead. To test its capabilities statistically, we plan to run the model with larger knowledge base consisting of much more entities. In many runs, we plan to vary randomly the order of their presentation; as well the weight of the links.

In addition, we plan a simulation to test the model's work with structures involving higher-order relations. This would demonstrate its ability to learn and use relational categories.

One of the strengths of all *DUAL* based models is their ability to combine bottom-up with top-down pressures. However, at the current development of the *RoleMap* model, the top-down pressures are underestimated. This is drawback of the model – it can find a similarity between a "bat" and "bird" for example, yet people have the knowledge that bats are not birds. Additional mechanisms for implementing top-down knowledge and kind of supervised learning are planned.

Finally, we are aware of the limitations that the hand-encoded knowledge base imposes, but we consider it as a necessary first step for evaluating the model.

Even though the model is still in its embryo phase, it proposes a powerful approach for learning hierarchically structured knowledge. This is done mainly by introducing two threshold parameters that determine whether a new exemplar will be classified as part of an existing category, a new superordinate category or a new subordinate category. This solution based on two thresholds is powerful and more general than just a mechanism in a concrete model. The same solution could be applied to other models of category learning and relational learning.

ACKNOWLEDGMENT

This research was financially supported by the European Office for Aerospace Research and Development under grant FA9550-15-1-0510 (Anticipating Future by Analogy-Making).

REFERENCES

[1] Collins, Allan M., and M. Ross Quillian. "Retrieval time from semantic memory." *Journal of verbal learning and verbal behavior* 8.2 (1969): 240-247.
[2] Collins, Allan M., and Elizabeth F. Loftus. "A spreading-activation theory of semantic processing." *Psychological review* 82.6 (1975): 407.
[3] Tenenbaum, Joshua B., et al. "How to grow a mind: Statistics, structure, and abstraction." *Science* 331.6022 (2011): 1279-1285.
[4] Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. *Learning internal representations by error propagation*. No. ICS-8506. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
[5] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.
[6] Amos, Brandon, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. "Openface: A general-purpose face recognition library with mobile applications." *CMU School of Computer Science* (2016).
[7] Graves, Alex, Abdel-rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." Acoustics, speech and signal processing (icassp), 2013 *ieee international conference on. IEEE*, 2013.
[8] Zhou, Bolei, Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. "Learning deep features for scene recognition using places database." *Advances in neural information processing systems*. 2014.
[9] Ramanathan, Vignesh, et al. "Learning semantic relationships for better action retrieval in images." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
[10] Kuehne, Sven, et al. "SEQL: Category learning as progressive abstraction using structure mapping." *Proceedings of the 22nd annual meeting of the cognitive science society*. Vol. 4. 2000.
[11] Falkenhainer, Brian, Kenneth D. Forbus, and Dedre Gentner. "The structure-mapping engine: Algorithm and examples." *Artificial intelligence* 41.1 (1989): 1-63
[12] Kokinov, B. "The context-sensitive cognitive architecture DUAL." *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*. 1994.
[13] Petkov, Georgi, and Luiza Shahbazyan. "The RecMap model of active recognition based on analogical mapping." *Proceedings of the 2007 International Conference on Cognitive Modeling*. 2007.
[14] Biederman, Irving. "Recognition-by-components: a theory of human image understanding." Psychological review 94.2 (1987): 115.
[15] Hummel, John E. "Where view-based theories break down: The role of structure in shape perception and object recognition." *Cognitive dynamics: Conceptual change in humans and machines* (2000): 157-185.
[16] Hummel, John E., and Brian J. Stankiewicz. "Two roles for attention in shape perception: A structural description model of visual scrutiny." *Visual Cognition* 5.1-2 (1998): 49-79.
[17] Gentner, Dedre. "Structure-mapping: A theoretical framework for analogy." *Cognitive science* 7.2 (1983): 155-170.