Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Solving and Explaining Analogy Questions Using Semantic Networks

Adrian Boteanu, Sonia Chernova
Worcester Polytechnic Institute
100 Institute Road Worcester, MA 01609
aboteanu@wpi.edu, soniac@wpi.edu

Abstract

Analogies are a fundamental human reasoning pattern that relies on relational similarity. Understanding how analogies are formed facilitates the transfer of knowledge between contexts. The approach presented in this work focuses on obtaining precise interpretations of analogies. We leverage noisy semantic networks to answer and explain a wide spectrum of analogy questions. The core of our contribution, the Semantic Similarity Engine, consists of methods for extracting and comparing graph-contexts that reveal the relational parallelism that analogies are based on, while mitigating uncertainty in the semantic network. We demonstrate these methods in two tasks: answering multiple choice analogy questions and generating human readable analogy explanations. We evaluate our approach on two datasets totaling 600 analogy questions. Our results show reliable performance and low false-positive rate in question answering; human evaluators agreed with 96% of our analogy explanations.

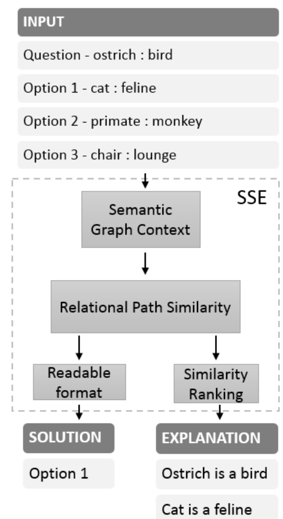


Figure 1: Overview of the Semantic Similarity Engine and its application to answering analogy questions and explaining the analogy relationship.

Introduction

Analogy is a powerful cognitive mechanism that enables people to transfer knowledge from one situation or context to another. By identifying similarities between situations, reasoning by analogy facilitates understanding, inference making, learning new abstractions and creating conceptual change (Schiff, Bauminger, and Toledo 2009). At the core of analogical reasoning lies the concept of similarity, which can be modeled as *featural* or *alignment-based* (Goldstone and Son 2005). The former relies on comparing observable attributes, while the latter emphasizes structural correspondences. Analogical reasoning is founded on the alignment-based model of similarity – the process of understanding an analogy requires reasoning from a relational perspective.

For AI, analogy solving presents an interesting and important problem because it offers the potential for deep problem understanding and automated generalization of learned tasks. In human learning, analogies have long been applied as a measure of verbal intelligence. Of particular interest to this work are verbal analogy questions commonly used on human standardized tests designed to evaluate understanding of relationships between a broad vocabulary of words.

A verbal analogy has the form A:B::C:D, meaning "A is to B, as C is to D". A question is formed by the first pair of words (A:B), followed by a number of possible answer pairs (C:D, E:F, etc.); the task is to select the answer pair for which the relation between the pair of words is the same as for the question pair. For example, given the initial pair *ostrich:bird*, and the options (a) *cat:feline*, (b) *primate:monkey* and (c) *chair:lounge*, the correct answer is (a), because the same *is a* relation connects the first word to the second word on both sides of the analogy.

Prior work in this area of AI includes several techniques for solving analogy questions designed for humans. Automated methods rely on latent analysis (Turney 2006). Crowdsourcing answers has been attempted, with no performance gains over statistical methods (Lofi 2013). None of these approaches produce interpretable justifications, focusing only on providing correct answer choices.

In this paper, we argue that complete analogical reasoning requires more than just the ability to select the correct answer choice. Equally importantly, we believe an analogical reasoning system must be able to effectively *model* and *explain* the mutual relationship that connects the pairs of words. Toward this end, we contribute the Semantic Similarity Engine (SSE), a framework that leverages noisy semantic networks to answer and interpret analogy questions.

Semantic networks are graphs in which words (or concepts) are represented in the nodes and the edges signify relations between them. Some networks are hand-crafted, for example WordNet (Pedersen, Patwardhan, and Michelizzi 2004), while others are generated automatically from mining documents. The key difference is that hand-crafted graphs are more accurate, but represent a smaller number of concepts and types of relations. Conversely, mining produces noisy graphs which express a broader set of concepts and relations. For evaluating analogies as diverse as those designed for humans, we use ConceptNet (Speer and Havasi 2012), a project which aggregates data from DBPedia, WordNet, VerbNet and other sources. ConceptNet represents relations of multiple types: lexical, functional, taxonomical, etc. In this work we use all fixed form relations from ConceptNet, 46 in total. Relations are expressed as weighted directed edges. Since it includes automatically generated data, ConceptNet has noise both in the nodes and in the edges.

Leveraging ConceptNet, we introduce techniques for reducing the concept-relation search space by extracting the graph context, evaluating relational sequence similarity within word pairs, answering questions using similarity ranking across word pairs and generating human-readable explanations for analogies. Figure 1 shows an overview of our system. The input and output of our system are closer to human-readable text than to structured representations, and 96% of human evaluators agreed with our analogy explanations.

Related Work

Prior work in this area has focused on answering multiple choice analogy questions via unsupervised latent analysis (Turney and Littman 2005; Turney 2006). Similar to LSA (Hofmann 1999), the authors introduce Latent Relational Analysis (LRA), in which the relation formed by each side of the analogy is modeled as a latent variable. Answers are selected based on likelihood ranking. The peak performance of LRA on the SAT dataset (which we also use in our work) is 56%. More recent approaches incorporate supervised learning, building on previous statistical methods (Turney 2013). We consider our work complementary to LRA, in that we focus on providing explanations to analogies while LRA is designed to answer multiple-choice questions and does not offer interpretable answers.

The core of our work relates to evaluating similarity at a relational level. The most closely related prior work is that on the Structure Mapping Engine (SME), which enables matching of relational characteristics between two semantic frames (Gentner 1983; Gentner et al. 1997). SME has been applied in multiple contexts, including sketch classification (Chang and Forbus 2012) and games (Hinrichs and Forbus 2007). Our approach has commonalities with SME in that it relies on one-to-one concept associations to form analogies. However, SME relies on a hand-crafted ontology (Cyc), while our methods are designed to be noise resilient, allowing us to benefit from a broader, automatically generated data source (ConceptNet). Another key difference is that both the input and output of our system are close to a human readable form (Figure 1), making it easy to integrate with natural language systems.

Additionally, previous approaches exist for uncovering unknown relations in semantic graphs. SPARQL is a common query language for accessing RDF triplet stores expressing semantic information (Quilitz and Leser 2008). SPARQL has been extended with queries that return the most direct connection between two nodes (Kochut and Janik 2007). Our context extraction method differs from this in that it returns a set of concept-relation sequences of different length connecting the pair of words instead of the shortest path.

Semantic Similarity Engine

Figure 1 shows the block diagram for SSE and how it processes analogy questions. The system has two common steps: extracting semantic contexts represented as graphs for each pair of words, and computing sequence similarity. This common core is then used for the tasks of explaining analogies and answering multiple choice questions.

Semantic Context Subgraph Extraction

The first stage of our pipeline is to extract the context defined by a pair of words, which we refer to as the *start words*. The goal of this stage is to model the relationship between the start words by identifying multiple semantic paths between them. We refer to chains of nodes and the relations connecting them as *sequences*, and define the *context* of a pair of start words as a graph containing the start words and sequences of nodes and relations by which they are connected.

It may not be immediately clear why we are seeking to identify multiple paths within the semantic network. In fact, many word pairs in our analogy dataset are directly linked by one (or more) of the 46 relationships within ConceptNet. However, indirect paths through other nodes may provide greater insight into the relationship of the start words themselves. Figure 2 presents an example of such a case, visualizing the graph extracted from ConceptNet for the word pair *goose*:*flock*. The correct relation implied by the analogy is *part of*, which is represented in the graph, but only for the superclass of *goose*, i.e. *bird*. The start words *goose* and *flock* are directly connected, but only through a less informative *related to* edge, while both have stronger connections with *bird* through *is a* and *part of* edges, respectively. It is therefore necessary to explore multiple paths of different lengths in order to reason effectively about the relationship between these words and find a good analogy explanation.

We generate the context graph for a given pair of start words in two steps. First, we extract the **unpruned semantic context**. This is performed by recursively expanding concepts in breadth-first order starting from the start words, caching a subgraph from the full semantic graph (i.e. Con-

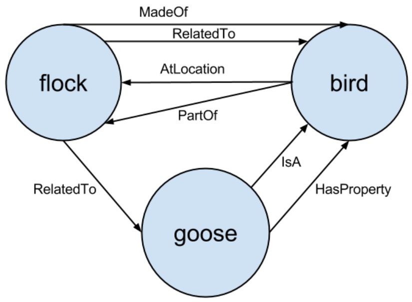
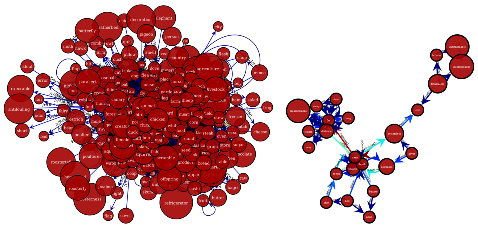


Figure 2: Example context surrounding *goose* and *flock*. The most meaningful sequece of relations is through an intermediate node, *bird*.



(a) (b)

Figure 3: Unpruned (a) and pruned (b) context graphs.

ceptNet). The entire graph is too large (tens of GB) to be accessed efficiently without caching. The stopping condition is a limit on the number of explored concepts. At each node addition in the breadth-first exploration, we test if there are edges to or from the rest of the nodes in the context-graph, and add them if so. This ensures that all existing relations between the context's words are captured. Figure 3(a) shows an unpruned graph example.

If the search fails to find a sequence between the start words, then analogical comparisons with another graph are not possible. This occurs for 14% of the word pairs within our dataset when using a 500 word limit for expansion. Using a larger limit did not impact results. Additionally, attempting to identify long sequences connecting the start words does little to aid the analogy solving or explanation process, since long sequences become difficult to interpret.

The context graph contains many leaf nodes that are irrelevant to the analogy. In the second step, we **prune** the graph by removing any nodes that are not part of a sequence between the start nodes. Edge direction and weight are ignored at this step. The result is a much smaller graph, typically consisting of tens of nodes, as illustrated in Figure 3(b).

Sequence Similarity

Now that we have a method for extracting the pruned context subgraph for any single pair of start words, we describe how two such contexts can be compared to determine the similarity in the relation between them. Specifically, we present an algorithm for identifying the *highest similarity sequence pair (HSSP)*, the sequence of nodes and edges that has the greatest number of common edges between two contexts.

Algorithm 1 Sequence similarity. s_1 and s_2 have the same length, each connecting a pair of concepts in different contexts.

1: $sim \leftarrow 1.0$
2: **for** $k\ in\ range(1, length(s_1) - 1)$ **do**
3: $\quad s_{1,k} \leftarrow s_1[k : k+1]$
4: $\quad s_{2,k} \leftarrow s_2[k : k+1])$
5: $\quad edges_1 \leftarrow context_1.get\_rel\_types(s_{1,k}[0], s_{1,k}[1])$
6: $\quad edges_2 \leftarrow context_2.get\_rel\_types(s_{2,k}[0], s_{2,k}[1])$
7: $\quad sim_k \leftarrow CommonRelProp(edges_1, edges_2)$
8: $\quad sim \leftarrow sim * sim_k$
9: **end for**
10: **return** sim

For a pair of context-graphs, we identify the HSSP by iterating through all possible pairs of sequences of the same length, one from each graph, and selecting the one with the highest similarity score. Algorithm 1 presents the algorithm for calculating the similarity score, which has a value between 0 and 1. For each sequence pair, s_1 and s_2, the algorithm iterates over the length of the sequence (line 2). For each segment, we compute the size of the set of common relations relative to the total set of relations present on that segment (lines 5-8). For example, comparing segments *A–B* and *C–D*, linked by relations $\{p,q,r\}$ and $\{r,p\}$ respectively, the similarity score becomes $2/3$, because there are two relations in common out of three total relations for this segment. At this stage, we do not yet take into account the weight or direction of edges, as this makes the algorithm more resilient to noisy edge additions or omissions within ConceptNet.

To generalize this algorithm for sequence pairs of arbitrary (but equal) length, we apply this metric to all segments of the sequence and multiply the similarity scores (lines 8). This ensures that if at any point the sequences are entirely dissimilar, the overall similarity is zero.

Modeling Analogies through SSE

The methods presented in the previous section allow us to find the best common relational link between two different pairs of words by searching a large semantic network. In this section, we describe two applications for understanding the relationship between word pairs: solving analogy questions and explaining analogies.

Answering Analogy Questions

Our approach to solving analogy questions stems directly from the similarity score obtained from the SSE. Our questions take the form presented in Figure 1. To select an answer, we first compute the HSSP between the question word pair and each possible answer. Then, we rank all answer options by their respective HSSP score and select the one with the highest score.

Options that have a similarity score of 0, or for which a context-graph connecting the pair of words can not be found, are discarded. We can then use the sequence pair that generated the similarity value (i.e. HSSP) to explain the analogy, as discussed in the following section.

In the results presented in this paper, we do not attempt to answer the question if there are no answers with a similarity score greater than 0. It is trivial to extend our technique to allow the algorithm to simply guess one of the multiple choice options. We do not utilize random guessing in this paper both to more accurately reflect the performance of the algorithm, and to facilitate our main goal of studying how the relationship behind the analogy can be explained. An answer obtained through guessing would make our algorithm, just as a human student, unable to explain the similarity between the two word pairs.

Explaining Analogies

Established practices for teaching human students to solve analogies instruct them to do so by forming a full sentence that clearly shows the relationship between the two question words, and then forming a second sentence that shows a similar relationship for their chosen answer word pair. The aim of our work is to generate the sentences that describe these relationships automatically.

If two pairs of words are stated to be an analogy, we can produce an interpretation from the corresponding HSSP. In order to obtain output that is easily readable, we need to reduce the HSSP from having multiple relations per segment to a single chain of relations (Algorithm 2). Therefore, we iterate through each segment along the sequences (line 4), and choose the salient common edge between the two sides of the analogy (lines 6-10), appending it to the explanation pair along with the corresponding concepts (lines 5, 10). Note that edge direction is taken into account in these steps.

We select the common relation with the highest minimum-weight, preventing imbalances in which only one side of the analogy is strongly related while the other is relatively weak. Figure 4 shows and example, in which the bolded relations are selected according to Algorithm 2.

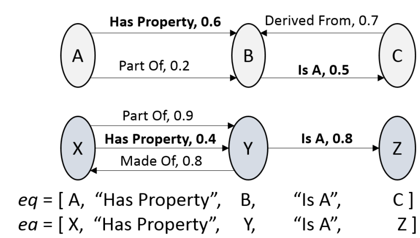


Figure 4: Starting from HSSP, we select the common salient edge on each segment to produce human-readable explanations. *eq* and *ea* are then converted to English.

Once a relation is selected for each segment, we convert the resulting list of nodes and relations into English using a dictionary which maps coded relations names to more readable versions (line 13). For example, a *PartOf* edge translates to *"is a part of."* While more sophisticated methods can be used to generate explanation phrases, grammatical correctness it is not the focus of our work; human evaluators were asked not to assess grammatical correctness.

Algorithm 2 Generating a human-readable explanation from the best similarity sequence pair, which have the same length.

1: $sq \leftarrow question\_seq;\ sa \leftarrow answer\_seq$
2: $eq \leftarrow [\,];\ ea \leftarrow [\,]$
3: $n \leftarrow length(question_s eq)$
4: **for** $k\ in\ 0...(n-1)$ **do**
5: $\quad eq.append(sq[k]);\ ea.append(sa[k])$
6: $\quad$ **for** $rel\ in\ \cap(sq[k:k{+}1].edges, sa[k:k{+}1].edges)$ **do**
7: $\quad\quad support[rel] = \min(sq[k,k+1].rel.weight, sa[k,k+1].rel.weight))$
8: $\quad$ **end for**
9: $\quad rel\_max \leftarrow rel\ for\ which\ \max(support[:])$
10: $\quad eq.append(rel\_max);\ ea.append(rel\_max)$
11: **end for**
12: $eq.append(sq[n]);\ ea.append(sa[n])$
13: **return** $convert\_to\_english(eq, ea)$

Results

We evaluate the SSE's ability to correctly answer and explain analogies using two datasets:

- 373 questions used in SAT US college admittance tests. This dataset was also used in previous work on answering analogies (Turney and Littman 2005); Table 1 shows question examples;
- 227 questions from a public domain website[1] targeted for grades 1-12. We combine these into four groups: elementary school (grades 1-4), middle school (grades 5-8) and high school (grades 9-12), containing 120, 60, and 47 questions, respectively.

In both datasets, each multiple choice question contains five possible answers. Combined, these questions form a progression of increasingly difficult analogy problems.

Question Answering Performance

In this section, we evaluate the SSE's performance in answering questions. We track two performance metrics:

1. *answer attempt proportion*, which represents the number of questions for which our algorithm selected an answer. As discussed earlier, our system attempts to answer a question only if there is at least one answer for which the HSSP has a non-zero similarity score, thus preventing random guessing.
2. *answer correctness proportion*, which represents how many of the attempted questions were answered correctly.

In our analysis, we limited the semantic context subgraphs to have a maximum geodesic distance of two. The geodesic distance is the shortest distance between two vertices in the graph, measured in number of edges. Longer paths, while feasible, proved to result in few answer attempts. In the results, we separately report performance for solutions with

[1] Section "Unit 2: Read Theory Word Pair Analogies" from http://www.englishforeveryone.org/Topics/Analogies.htm

Table 1: Question examples from the SAT dataset and the answer results of our approach (correct answers shown in bold).

QUESTION	OPTION A	OPTION B	OPTION C	OPTION D	OPTION E	ATTEMPTED	CORRECT
custom:society	hypothesis:evidence	testimony:trial	ballot:election	**rule:game**	contest:debate	yes	yes
seed:plant	pouch:kangaroo	root:soil	drop:water	bark:tree	**egg:bird**	yes	yes
lull:trust	balk:fortitude	betray:loyalty	**cajole:compliance**	hinder:destination	soothe:passion	yes	no
virtuoso:music	**bard:poetry**	crescendo:scale	lyricist:melody	portrait:photography	critic:performance	yes	no
querulous:complain	silent:talk	humorous:laugh	dangerous:risk	**deceitful:cheat**	gracious:accept	no	-
audacious:boldness	anonymous:identity	remorseful:misdeed	deleterious:result	impressionable:temptation	**sanctimonious:hypocrisy**	no	-

a distance of 1, which we call *direct*, and those with a distance of two, which we call *one-hop*, to demonstrate the frequency of occurrence and reliability of both cases. We show results separately because an answer may be available for each path length. These answers could be combined via ensemble methods to increase the answer attempt proportion, but that extension is outside the scope of this paper.

Additionally, to establish how well the SSE measures similarity, we compare against a baseline approach that relies on ConceptNet's own metric for similarity provided by the Divisi toolkit (Speer, Arnold, and Havasi 2010). This metric is a real number between 0 and 1, calculated using the SVD within ConceptNet. In this experimental condition we performed question answering as follows: for each pair of start words, we computed the Divisi similarity value; we selected the answer for which the similarity was numerically closest to the similarity of the question pair, attempting answers only if the question and at least one answer had non-zero similarity.

Figure 5(a) presents answering performance for direct, one-hop and baseline methods over the analogy data set. The solid line shows the proportion of attempted questions, while the histogram presents the proportion of correct answers of each method. Across all four question levels, the baseline technique attempts to answer a large percentage of questions, but has low accuracy. Its best performance is on the elementary school data set, where it achieves 28% accuracy. By comparison, both SSE conditions (direct and one-hop), are less likely to attempt an answer, but have a significantly higher accuracy.

In the elementary grade data set, the direct solver achieves an accuracy of 85%. While performance declines as question difficulty increases, both solvers answer correctly 83% of attempts on average in the 1-12 grade dataset. As questions become more difficult, especially for the SAT dataset, knowledge of the words' meanings becomes key. SAT questions often focus on rarely encountered words, so it is unsurprising that the attempt ratio decreases due to lack of connections between the start words. Despite this, the SSE methods achieve answer correctness of 40% on the SAT dataset.

The experimental results presented in Figure 5(a) were obtained by utilizing all 46 relations found in ConceptNet. In Figure 5(b) we present similar results for a second condition in which we ignore two relation types, *RelatedTo* and *ConceptuallyRelatedTo*, which are unique within ConceptNet because they are derived statistically from document co-occurrence and thus are far more noisy. Moreover, they are not useful for generating explanations. In this condition we note that the attempt proportion is lower, since many edges within the context are ignored. However, the accuracy for attempted question is higher than in the original condition, providing a means of regulating the algorithm's behavior in focusing more or less on accuracy vs. attempts.

In summary, the results demonstrate that the SSE-based methods for analogy solving achieve high accuracy in this difficult domain, but sacrifice coverage by not attempting to answer questions which would require the algorithm to guess. We made this design choice due to our focus on explaining analogy questions, which requires the ability to accurately model the relationship between word pairs. We discuss our results for generating explanations in the following section.

Explanation Performance

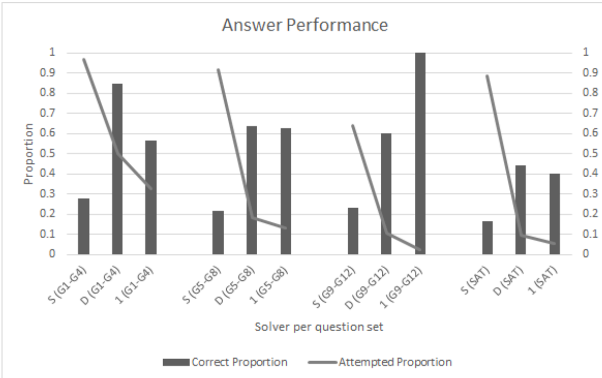
The ultimate goal of our system is to generate full sentence, human readable explanations of analogy question solutions. To evaluate our success, we surveyed people to test whether the explanations generated by our algorithm were meaningful to them. We conducted our survey through the CrowdFlower crowdsourcing market using the 74 explanations (60 direct, 14 one-hop) produced by SSE from the correct answers selected when ignoring *RelatedTo* and *ConceptuallyRelatedTo* edges.

For each explanation, the survey first presented the corresponding analogy question in its full form, including the question statement and all possible answers (as in Figure 1). Then we told readers the true answer, followed by the explanation. Participants were asked to choose whether the explanation correctly justified the answer.

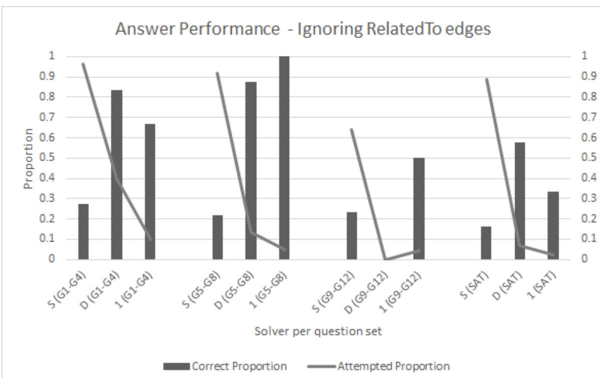
To evaluate the quality of our explanations, and to ensure that human readers were paying attention, we ran this study by randomly interleaving examples from three conditions:

1. *SSE-generated explanations* - an explanation generated from a HSSP selected by SSE;
2. *randomized explanations* - we substituted the relations in the SSE-generated set with random selections from the set of 44 relations;
3. *edited explanations* - we substituted the relations in the SSE-generated set with manually edited relations that were contradictory to the true relation.

Table 2 presents examples of all three explanation types. We included the randomized and edited conditions in our analysis because many relations within ConceptNet are similar, and thus selecting one at random may result in a relation that was very close, but not identical to, the SSE-selected one. Our goal was to verify that the SSE-selected relations, and the explanations derived from them, were clearly correct and differentiable from both random noise and wrong answers.



(a)



(b)

Figure 5: Answer performance on four datasets, comparing the direct (D) and one-hop (1H) methods with the Divisi similarity baseline (S) both when using (top) and when ignoring (bottom) statistically derived relations. Questions are grouped by elementary grades (G1-G4), middle school grades (G5-G8), high school (G9-G12) grades and SAT.

In total, the surveys tested 222 explanations (three study conditions applied to 74 questions), and each question received at least 5 judgements (5.79 on average). Figure 6 reports the proportion of analogy explanations that participants considered to be valid for each condition, reported separately for direct and one-hop solvers. We observe that human evaluators agreed with 96% of the explanations produced by our method – all but a single one-hop explanation were accepted. Approximately half of the randomly selected explanations were considered valid, and we observed higher disagreement between participants in this dataset (study-wide inter-user agreement was high, 87% on average, but only 78% for randomized condition). When analyzing these instances case by case, we found many of the randomly selected explanations to be reasonable, if not entirely sound. However, in the edited dataset, which contained intentionally illogical relations, very few were considered valid. This result strongly supports the validity of the SSE-generated similarity relationships and the analogy explanations founded upon them.

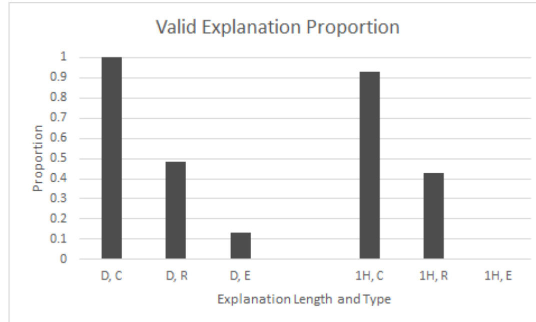


Figure 6: Evaluation of explanation quality for direct (D) and one-hop paths (1H), comparing our system's output (C) with randomized (R) and edited (E) conditions.

Table 2: Question examples from the SAT dataset and the answer results of our approach (correct answers in bold).

Solver	Dataset	Explanation Pair
Direct	SSE-selected	**fire** has property of **hot** **ice** has property of **cold**
Direct	Randomized	**fire** is a member of **hot** **ice** is a member of **cold**
Direct	Edited	**fire** is not **hot** **ice** is not **cold**
One-hop	SSE-selected	**tub** is used for **bath**, which is at location **bathroom** **stove** is used for **cook**, which is at location **kitchen**
One-hop	Randomized	**tub** is located near **bath**, which is attribute of **bathroom** **stove** is located near **cook**, which is attribute of **kitchen**
One-hop	Edited	**tub** is participle of **bath**, which inherits from **bathroom** **stove** is the participle of **cook**, which inherits from **kitchen**

Finally, we note that the relatively strong performance of the randomized dataset suggests that humans were less sensitive to the exact wording of the analogy explanation and accepted relatively close relation substitutes, as long as the substituted relation was sufficiently similar to the one intended by the analogy. This result has broader implications, as it suggests that correctly identifying the *ideal* relation represented by the analogy may not be necessary. Computationally, the set of all possible analogy relations is potentially very large. However, if we allow for approximations, multiple analogy relationships can be collapsed, a hypothesis that is supported by these results.

Conclusion

Analogies are an essential reasoning pattern that enables learning through similarity and skill transfer. In this work, we presented a method for evaluating analogical similarity by comparing paths within semantic context graphs derived from large scale noisy semantic networks. We demonstrated the effectiveness of our approach on two applications, solving multiple choice analogy questions and generating human-readable explanations for the analogies. Our results demonstrate that our methods achieve high accuracy in correctly answering attempted questions, and surveyed study participants agreed with the analogy explanations generated by our algorithm in 96% of cases. In future work, we plan to explore the use of semantic similarity in other domains, including scene understanding and skill transfer.

References

Chang, M. D., and Forbus, K. D. 2012. Using quantitative information to improve analogical matching between sketches. *Innovative Applications of Artificial Intelligence (IAAI). Toronto, Canada*.

Gentner, D.; Brem, S.; Ferguson, R. W.; Markman, A. B.; Levidow, B. B.; Wolff, P.; and Forbus, K. D. 1997. Analogical reasoning and conceptual change: A case study of johannes kepler. *The journal of the learning sciences* 6(1):3–40.

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

Goldstone, R. L., and Son, J. Y. 2005. *Similarity.* Cambridge University Press.

Hinrichs, T. R., and Forbus, K. D. 2007. Analogical learning in a turn-based strategy game. In *IJCAI*, 853–858.

Hofmann, T. 1999. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, 289–296. Morgan Kaufmann Publishers Inc.

Kochut, K. J., and Janik, M. 2007. Sparqler: Extended sparql for semantic association discovery. In *The Semantic Web: Research and Applications*. Springer. 145–159.

Lofi, C. 2013. Just ask a human?–controlling quality in relational similarity and analogy processing using the crowd. In *BTW Workshops*, 197–210.

Pedersen, T.; Patwardhan, S.; and Michelizzi, J. 2004. Wordnet:: Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, 38–41. Association for Computational Linguistics.

Quilitz, B., and Leser, U. 2008. *Querying distributed RDF data sources with SPARQL.* Springer.

Schiff, R.; Bauminger, N.; and Toledo, I. 2009. Analogical problem solving in children with verbal and nonverbal learning disabilities. *Journal of Learning Disabilities* 42(1):3–13.

Speer, R., and Havasi, C. 2012. Representing general relational knowledge in conceptnet 5. In *LREC*, 3679–3686.

Speer, R.; Arnold, K.; and Havasi, C. 2010. Divisi: Learning from semantic networks and sparse svd. In *Proc. 9th Python in Science Conf.(SCIPY 2010)*.

Turney, P. D., and Littman, M. L. 2005. Corpus-based learning of analogies and semantic relations. *Machine Learning* 60(1-3):251–278.

Turney, P. D. 2006. Similarity of semantic relations. *Computational Linguistics* 32(3):379–416.

Turney, P. D. 2013. Distributional semantics beyond words: Supervised learning of analogy and paraphrase. *arXiv preprint arXiv:1310.5042*.