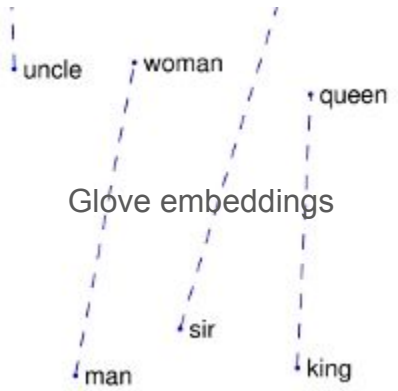
Word & System Analogies Retrieval

Akshara Prabhakar and Margarita Belova

https://github.com/across-stars/pinecone/tree/master

Methodology overview:



- Take datasets showing word, relation and system analogies
- Index word and relation embeddings in Pinecone
  - GloVe (glove42) – word embeddings (“kant” → vector)
  - RelBERT – word pair, or relation embeddings (“kant__philosopher” → vector)
- Use KNN to retrieve analogical words / pairs

Results:

- Compared knn analogical retrieval performance: *Glove* vs *RelBERT* embeddings
- Extended the method for complex system analogies retrieval

Word Analogy

$$w_1 : w_2 :: w_3 : w_4$$

For Glove embeddings check:

$$w_4 = w_3 + (w_2 - w_1) + \delta,$$

Algorithm:

- Retrieve w1, w2, w3 from Pinecone db;
- Retrieve knn(w3 + w2 - w1)
- label(w4) – ground truth
- Check: label(w4) ∈ knn(w3 + w2 - w1)

Glove database:

- 1_894_411 embeddings;
- dim = 300;
- Metric: cosine

source:
https://nlp.stanford.edu/projects/glove/
Common Crawl (42B tokens, 1.9M vocab, uncased)

w/o non-ASCII symbols

```
base,target,top0,top1,top2,top3,top4,k_order,k_score
belgrade,serbia,belgrade,serbia,lebanon,macedonia,kosovo,1,0.678768814
manila,philippines,manila,philippines,philippine,singapore,cebu,1,0.738674104
paris,france,paris,france,french,belgium,pierre,1,0.692115188
lilongwe,malawi,malawi,lilongwe,gabon,botswana,zambia,0,0.703917921
```

Relation Analogy w1_w2 :: w3_w4

For Relbert embeddings check:
w3_w4 ∈ knn(w1_w2)

Algorithm:

- Retrieve w1_w2 from Pinecone db;
- Retrieve knn(w1_w2)
- label(w3_w4) – ground truth
- Check: label(w3_w4) ∈ knn(w1_w2)

RelBERT database:

- 104_973 embeddings;
- dim = 1024;
- Metric: cosine

```
base,target,top0,top1,top2,top3,top4,k_order,k_score
hitler__dictator,strauss__composer,hitler__dictator,napoleon__emperor,charlatan__impostor,superman__emperor,truman__president,,
rousseau__writer,hegel__philosopher,rousseau__writer,tolstoi__novelist,andersen__writer,raphael__painter,wagner__composer,,
kant__philosopher,lincoln__president,kant__philosopher,mencius__philosopher,locke__philosopher,raphael__painter,tolstoi__novelist,,
stalin__dictator,hawking__physicist,stalin__dictator,newton__scientist,strauss__composer,stalin__napoleon,kepler__mathematician,,
```

System Analogy: definition

$$< concept_1 \rightarrow relation_1 \rightarrow concept_2 >,$$
$$\{< u_1 \rightarrow r_1 \rightarrow u_2 >, < u_1 \rightarrow r_2 \rightarrow u_3 >, \ldots\} :: \{< v_1 \rightarrow r_1 \rightarrow v_2 >, < v_1 \rightarrow r_3 \rightarrow v_3 >, \ldots\}$$

Examples (from SCAN_dataset):

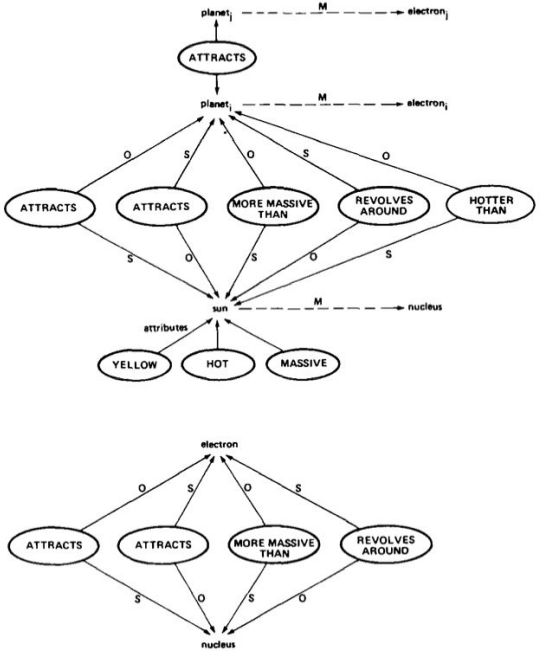
Solar system -- Atom:

Solar system:

- Sun
- Planet
- Mass
- Attracts
- Revolves
- Gravity

Atom:

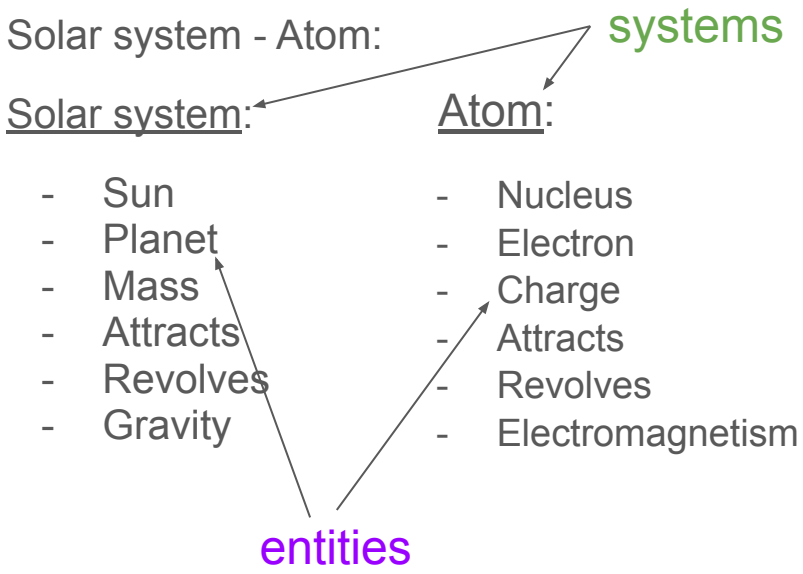
- Nucleus
- Electron
- Charge
- Attracts
- Revolves
- Electromagnetism



Genter et. al. (1983)

System Analogy: task

$$< concept_1 \rightarrow relation_1 \rightarrow concept_2 >,$$
$$\{< u_1 \rightarrow r_1 \rightarrow u_2 >, < u_1 \rightarrow r_2 \rightarrow u_3 >, \ldots\} :: \{< v_1 \rightarrow r_1 \rightarrow v_2 >, < v_1 \rightarrow r_3 \rightarrow v_3 >, \ldots\}$$



Task:
Given a system and its entities, retrieve the most analogous system

https://github.com/taczin/SCAN_analogies/tree/main

System analogy: methodology

	target	source	targ_word	src_word
2	atom	solar system	nucleus	sun
3	atom	solar system	electron	planet
4	atom	solar system	charge	mass
5	atom	solar system	attracts	attracts
6	atom	solar system	revolves	revolves
7	atom	solar system	electromagnetism	gravity

GloVe + RelBERT for relations = System analogy

- Retrieve RelBERT for every pair source__source_word

  Relbert(atom__nucleus), Relbert(atom__electron), Relbert(atom__charge)... → [w1, w2, w3…]

- Retrieve **labels** for knn(w1), knn(w2), knn(w3)... → <{}, {}, {}...> – sets of labels pairs

  E.g. knn(atom__charge) = {object__light, blossom__water, vital__energy…}

- Parse labels: pick the most frequent label across entities – this is the analogical system (*target*)

We have source (***s***) and target (***t***) labels that are words.

We can retrieve from Glove ***r = t - s*** – relation and save it to our *library of relations*.

Evaluation Setup

Datasets

- BATS - 1799 example
- GOOGLE - 500 examples
- SAT - 337 examples
- SCAN (system analogy) - 45 examples

Metric

- Accuracy @ k

RelBERT

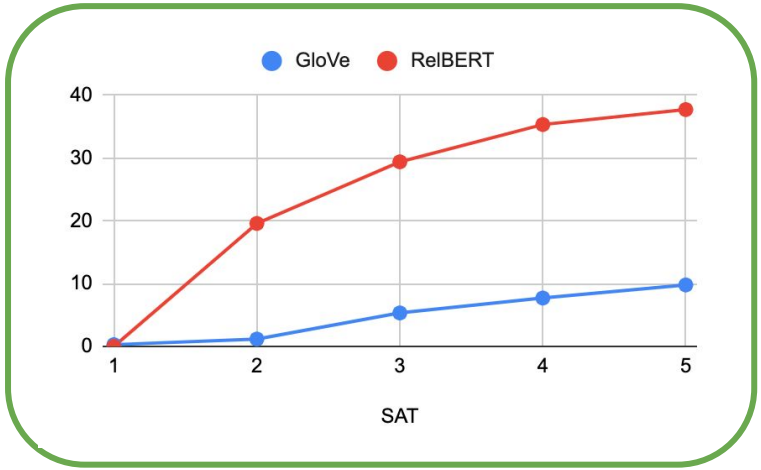
- Distilling relation embeddings from RoBERTa (roberta-large)
- Manual Prompting
- Triplet loss to finetune
  - x_a (anchor), x_p (positive), x_n (negative)

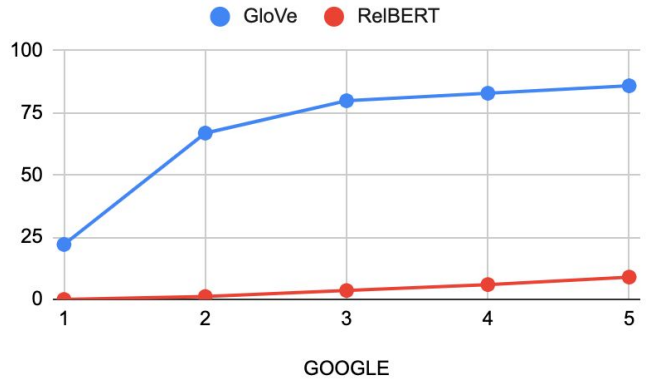
$$L_t = \max\left(0, \|\boldsymbol{x}_a - \boldsymbol{x}_p\| - \|\boldsymbol{x}_a - \boldsymbol{x}_n\| + \varepsilon\right)$$

1. Today, I finally discovered the relation between **[h]** and **[t]** : **[h]** is the <mask> of **[t]**
2. Today, I finally discovered the relation between **[h]** and **[t]** : **[t]** is **[h]**'s <mask>
3. Today, I finally discovered the relation between **[h]** and **[t]** : <mask>
4. I wasn't aware of this relationship, but I just read in the encyclopedia that **[h]** is the <mask> of **[t]**
5. I wasn't aware of this relationship, but I just read in the encyclopedia that **[t]** is **[h]**'s <mask>

https://arxiv.org/pdf/2110.15705.pdf

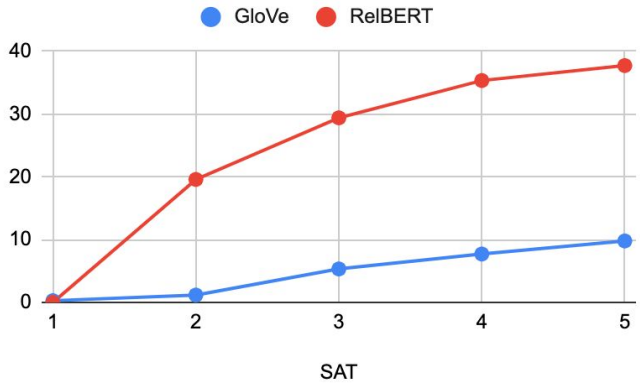
Word Analogy Results

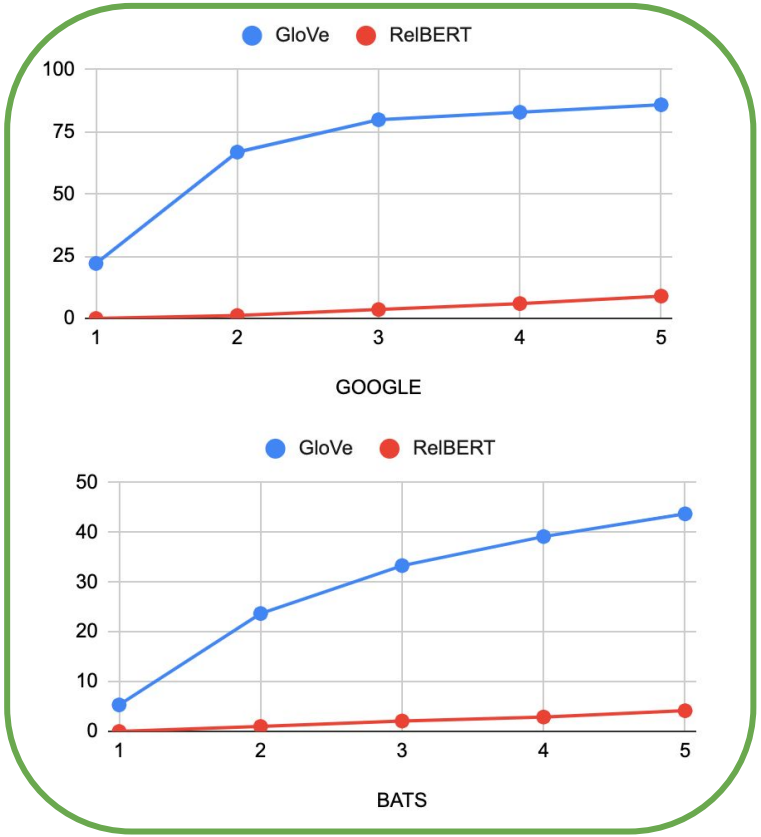




- Questions from real SAT exams

Word Analogy Results





- GOOGLE dataset has been shown be biased towards word similarity

System Analogy Results

- Hard accuracy: 15.56%
  - Considering exact match only

Some predictions

reasons_for_a_theory ⟷ grounds_for_a_building

gas_molecules ⟷ billiard_balls

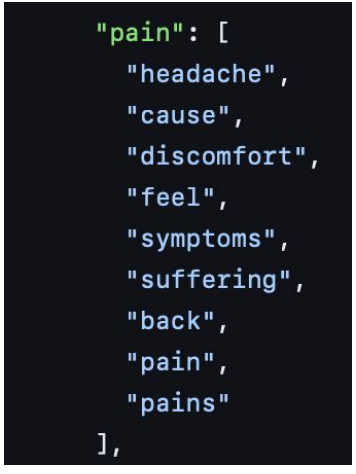
respiration ⟷ combustion

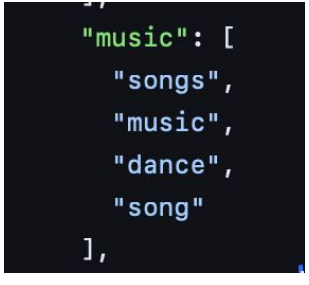
bacterial_mutation ⟷ distortion

NEW System Analogy: Library of relations (created early)

- 443 relations
- SVD of them to remove noise
- Take a random word ***w***, add a relation ***r*** from library and retrieve knn(***w+r***) from Glove
- Hopefully, it gives some analogy…

NEW System Analogy: Library of relations (created early)





SVD of relation matrix rank 10

SVD of relation matrix rank 80