# Final Project Submission *Relations as Affine Transformations in Word Embeddings*

**Submitted by:** *Eduardo Salvador Hidalgo Vargas*

## 1 Abstract

Word embeddings capture semantic relations as *translation* between vectors. This project proposes and evaluates a method that given a relation and an embedding vector space, it learns a representation of such relation as an affine transformation. The learned transformation performs 5-18 points better than a simple translation which is what the analogy task usually performs.

## 2 Introduction

Extensive work shows that word embeddings can capture semantic and syntactic relations [4] [5] [3] as *vector offsets*. Specifically, such embeddings are usually evaluated in similarity tasks (distance between related words) and analogy tasks (complete the sentence: "$\boldsymbol{x}$ is to $\boldsymbol{y}$, as $\boldsymbol{z}$ is to $\boldsymbol{w}$") where the type of relation is not explicitly provided. For a human to answer, the relation needs to be infered and then applied it to the third argument [3]. In [6] they propose a more specific task of predicting the hypernym of a given word. It can be seen as the task of completing the sentence "*x is a y*".

We propose a variation of this task that is generic over the type of relation. For example: "*a* $\boldsymbol{x}$ *can* $\boldsymbol{y}$" or "*a is* $\boldsymbol{x}$ *cause of* $\boldsymbol{y}$". This task is challenging in at least two different ways with respect to the analogy task: 1) word relations need not be *functions*: there might be two such instances of $b$ that hold. e.g. *a bee can sting* and *a bee can fly*; and 2) polysemy in an argument e.g. *a dog can bark*, bark(make a loud noise) and bark(part of plant). This are pitfalls of word embeddings in general, but the analogy task alleviates it by possibly dissambiguating with the third argument $c$.

A dataset extracted from `ConceptNet` is used to find answers to such task with vector algebra, akin to work in the analogy task. This can be described as a way to represent the symbolic knowledge of concepts and edges into the word embeddings, where there's a correspondence between concept $\sim$ embedding, and edge $\sim$ affine transformation.

### 2.1 Novel Aspects

This new formulation of the task as searching a single affine transformation for all triplets of a relation over an embedding space is,to the best of my knowledge, novel.

Both GloVe and word2vec approaches to this is through analogies that do not make the specific relation between terms explicit. This is a tradeoff between generality and interpretability.

## 3 Problem Definition

For concepts $c_h, c_t$ with corresponding word representations $w_h, w_t \in V : \mathbb{R}^d$ and a relationship between them $r$, we want to find a function $\boldsymbol{f_r} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ such that:

$$\boldsymbol{f_r}(w_h) \approx w_t \tag{1}$$

This approximation has an underlying error because it is trying to model a relation as a function. Properties of $r$ such as symmetry/asymmetry, transitivity and reflexivity make this task hard. Additionally if $r$ is non-injective (such as is-a relationship) it can be harder to learn and possibly it's inverse relation might be easier, e.g. *"x is a hypernym of y"* is harder to learn than its inverse *"x is a hyponym of y*.

ConceptNet can be seen as a Set of triplets of the form $(r, c_h, c_t)$ that is a relation $r$ with a concept that is the head $c_h$ followed by a tail concept $c_t$.

# 4 Technical Approach

With Equation (1) as an objective, we now choose $\boldsymbol{f_r}$, since we want to capture *geometric properties* of the underlying vector space of the embeddings, We will limit $\boldsymbol{f_r}$ to be either a translation (2), or an affine transformation (3):

$$w_h + x \approx w_t \tag{2}$$

$$Ww_h + b \approx w_t \tag{3}$$

In the actual implementation, an affine transformation is equivalent to a single linear layer + a bias. During development I tried different architectures such as 2 tanh layers, and while it increased accuracy in some cases, it did not justified losing the interpretability of the affine transformations.

This method is independent of the embedding except for the dimensions of the Affine transformation matrix and translation vector, therefore it is another way to intrisically measure the semantic information captured in such embeddings. Work [2] has been done in exploring how to *retrofit* simbolic knowledge of relations back into the word embeddings, one such example being ConceptNet NumberBatch [9] that consolidates the word embeddings from multiple sources into one and injects knowledge of the ConceptNet graph to increase performance.

Other influential approaches like `TransE`, [1] and `TransH` [10] Search to minimize equations similar to (2) defined as a loss, but they do it directly in the embeddings. In contrast to our approach that takes existing embeddings and searches for such a transformation.

During training, we want to minimize the mean square loss of equation (1). And during test, we query our model by asking for the most similar vector $\widehat{w_t}$ in the embedding space $V$, $v$ that corresponds to our predicted concept $\widehat{c_t}$.

$$\underset{w_t \in V}{\operatorname{argmax}} \left( f_r(w_h) \right) \tag{4}$$

There is an alternate formulation as a multi-classification problem not explored in this report where given a pair of concepts the model assigns one relation to it.

# 5 Evaluation

## 5.1 Rationale

Our main question is if there exists such transformation that satisfies (1) reasonably well. By reasonably well we expect the transformation to be able to predict the corresponding $c_t$ for a concept that it has not seen, and that it performs better than simple translation (the baseline).

Table 1: Relational properties of `ConceptNet`

| Relation $A \rightarrow B$ | triplets | unique A | unique B | avg A | avg B |
|---|---|---|---|---|---|
| `IsA` | 63,184 | 43,548 | 10,812 | 1.45 | 5.84 |
| `CapableOf` | 874 | 514 | 550 | 1.70 | 1.59 |
| `Causes` | 1,272 | 456 | 768 | 2.79 | 1.65 |
| `UsedFor` | 4,208 | 1,192 | 2,009 | 3.53 | 2.09 |
| `Antonym` | 14,388 | 9,006 | 10,030 | 1.60 | 1.43 |

Some other interesting aspects are how the *relational properties* of the dataset e.g. symmetry and arity of the relation affects the performance of the corresponding transformation.

### 5.2 Experimental Settings

The data for the triplets was extracted from `ConceptNet 5.7` [8]. From the whole dataset of 3.1 million assertions, we filtered tuples with the following aspects: 1) both concepts are in english, 2) both concepts are single words, and 3) the relation is not deprecated and is not `/r/ExternalURL`.

Five relationships (in Table 1) were selected for their different properties and because they were larger than 7000 triplets after the initial filtering. For example, Antonym relation is symmetric; while IsA relation is transitive, it also shows its "tree-like" structure since the B nodes (parents in this case) have more connections on average than children (A nodes) to parents.

The baseline is a simple translation that is learned in the same way as the affine transformation: an implementation in pytorch with optimizer `Adam` (default paramenters: $\alpha = 0.001$, $\beta = (0.9, 0.999)$ and $\epsilon = 1^{-8}$). It was trained for 2000 epochs of batch gradient descent, and a loss function of mean squared error. since the transformation can also rotate/skew/scale the original vector it is expected to perform better. No hyper-parameter was tuned. The size of the translation vector is the same as the embedding dimension $d$, and the shape of transformation matrix is $(d, d)$. All tests were done against *gensim's* `glove-wiki-gigaword-200` word embeddings. For each relation, deterministically 15% of the dataset was for testing and the rest was for training. Only

### 5.3 Results

Table 2 presents the results of testing the model against every relationship, with only translation (model of equation 2) and with an affine transformation (model of equation 3). model 3 consistently performs better than model 2 (17%-43% relative improvement), while still being pretty simple (no activation nor hidden layers). This hints that the word embedding do capture to some extent this complex(not 1-to-1) semantic relationships almost linearly, it also suggests that preffering a matrix multiplication over a simple vector difference might yield better results for other tasks with embeddings, as a tradeoff, now that matrix needs to be searched.

## 6 Summary

This project proposes a new method to represent semantic relations as an affine transformation over word embeddings. I learned about different word embeddings, their nuances and properties, how they are trained. I also got familiar with ConceptNet and how to manipulate for preprocessing using unix and python scripts. If time were aplently, I would extend this work to also cover different embeddings beyond GloVe 200. Other directions of work is multi-word concepts

Table 2: F1 Scores of the prediction task, for model with only translation and with affine transformation. `glove-wiki-gigaword-200`

| Relation | model (2) | model (3) | absolute / relative difference |
|---|---|---|---|
| IsA | 0.333 | **0.403** | 0.07 / 17.37% |
| CapableOf | 0.236 | **0.338** | 0.102 / 30.18% |
| Causes | 0.313 | **0.368** | 0.055 / 14.95% |
| UsedFor | 0.245 | **0.434** | 0.189 / 43.55% |
| Antonym | 0.421 | **0.507** | 0.086 / 16.96% |

that are abundant in ConceptNet which represent the challenge of extending the embeddings to also handle those, finally it would be interesting to model relations of multiple arguments, e.g. $and(“water, “cold'') \rightarrow “ice''$ and other types of reasoning beyond single aasertions. As a possible application, this lightweight transformations can be use as an inference step of much larger models.

### 6.1 Difference from proposal

Because of the unconvential approach there was high uncertainty on whether there was in fact affine transformations in the embeddings or not and if they were learnable, the loss function changed but the core formulation of the task stayed the same. Learning a representation for each relation turned out to be a field with a lot or possible experiments so the focus was turned exclusively on that and dropped the secondary proposed subtask of infering chains of relations for `CommonSenseQA`. The proposal also included a section of running the relation against unincluded concepts to propose new edges, while this was observed informally during development there was no formal experimentation in that direction.

## 7 Team member contribution

Sadly my original team member Dan Luo had to drop the class for personal reasons so I did the whole report, algorithm implementation, model testing and output evaluation by myself. She was really important during the development of the idea and did half of the job when writing the project proposal.

## References

[1] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2787–2795. Curran Associates, Inc., 2013.

[2] M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. H. Hovy, and N. A. Smith. Retrofitting word vectors to semantic lexicons. *CoRR*, abs/1411.4166, 2014.

[3] O. Levy and Y. Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180, 2014.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[5] T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 746–751, 2013.

[6] N. Nayak. In learning hyperonyms over word embeddings.

[7] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[8] R. Speer, J. Chin, and C. Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. *CoRR*, abs/1612.03975, 2016.

[9] R. Speer and J. Lowry-Duda. Conceptnet at semeval-2017 task 2: Extending word embeddings with multilingual relational knowledge. *CoRR*, abs/1704.03560, 2017.

[10] Z. Wang, J. Zhang, J. Feng, and Z. Chen. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119, 2014.