Project Proposal

- 2020 SPRING CS577 NLP -

Dan Luo
luo227@purdue.edu

Eduardo Hidalgo
hidalgov@purdue.edu

1. INTRODUCTION

Since word embeddings are vectors in $\mathbb{R}^k$ We can think of a relationship between them as a geometric transformation. `TransE` models such relation as a translation, other more complicated models like `TransH` models this as a translation on a hyperplane. This approach is inspired by word arithmetic performed on word2vec embeddings like the classic example $king - man + woman = queen$ and from work both in GloVe and the papers related to word2vec [1] where they show that this embeddings capture some relations like capital-of as translations. We propose to generalize this intuition of how geometric properties capture relations between words by making modelling them as affine transformations, since all translations is a subset of affine transformations.

To evaluate this approach in a real task, we will apply it to the task of question answering in CommonsenseQA, by infering a possible sequence of relations between the entities in the question. Figure 1 is an example of such a chain of relations that we should be able to infer.

2. LITERATURE REVIEW

For our literature survey, we have picked three papers that tackle related issues. This should provide some foundation for us to understand the problem in a better light and identify where we can improve from these past works.

2.1 Translating Embeddings for Modeling Multi-relational Data

A method[2] that models relationships as the translations operating on the embeddings of the entities. Formally, given a set S of triplets (h, r, t) composed of two entities $h, t \in E$ (the set of entities) and $r \in R$ (set of relationships) Their objective is to learn embeddings in $\mathbb{R}^k$ (k is a hyperparameter) for every h, r, t such that:

$$\boldsymbol{h} + \boldsymbol{r} \approx \boldsymbol{t}$$

In contrast with our approach:

1. We will not learn the entity embeddings, only learn relations as *transformations* on already pretrained embeddings like GloVe or Word2Vec.
2. We will explore transformations beyond a simple translation, like *linear transformations* and *affine transformations.*

2.2 Knowledge Graph Embedding by Translating on Hyperplanes

`TransH` is an extension of `TransE` That increases it's capabilities and improve it's performance while trying to still be efficient. [3] The main issue they tackle is that `TransE` does not deal well with relations that are: reflexive, 1-N, N-1, and N-N. This has a mathematical intuition, because for that model:

1. **reflexivity:** for triplets (h, r, t) and (t, r, h) $r = 0$ and $h = t$ must hold
2. **1-N, N-1, and N-N:** $\forall i \in \{0, ..., m\}$ and (h_i, r, t), if r is a 1-N, then all h_i must be equal. Similar reasoning in the case of N-1 for all t_i and in the case of N-N both h_i and t_j are collapsed into one.

The key idea to overcome this limitation is to first project h and t to a hyperplane of that relation, then learn a translation on that hyperplane:

$$(\boldsymbol{h} - \boldsymbol{w}_r^T \boldsymbol{h} \boldsymbol{w}_r) + \boldsymbol{d}_r \approx (\boldsymbol{t} - \boldsymbol{w}_r^T \boldsymbol{t} \boldsymbol{w}_r)$$

There is one: $\boldsymbol{w}_r$ and $\boldsymbol{d}_r$ for each relation r. This is the `TransH` model. Just like `TransE`, Our approach is only interested in learning the relations and not the embeddings. TransH model seems like an interesting approach that could be integrated to our method and improve it's performance, since the affine transformations proposed has the same drawbacks as TransE with respect to multi-arity relations.

2.3 Characterizing the impact of geometric properties of word embeddings on task performance

The *Characterizing the impact of geometric properties of word embeddings on task performance*[4] rose a question that geometric properties of the continuous feature space contribute directly to the use of embedding features in downstream models, and are largely unexplored. So they set up some experiment define a sequence of transformations to generate new embeddings that expose subsets of these properties to downstream models and evaluate change in task performance to understand the contribution of each

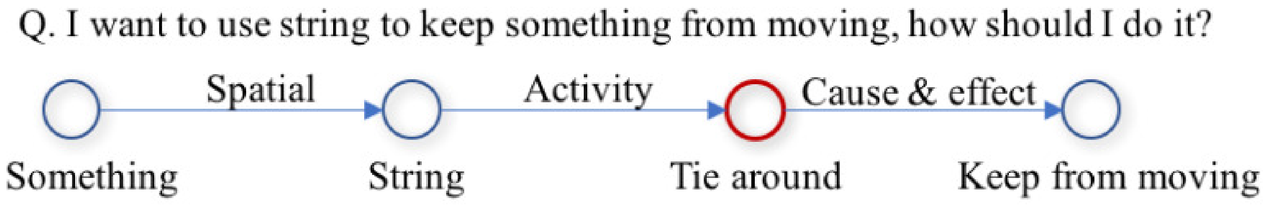
Q. I want to use string to keep something from moving, how should I do it?
Spatial
Activity
Cause & effect
Something
String
Tie around
Keep from moving

Figure 1: An example of the chain relation we would learn to explain the CommonSenseQA.

property to NLP models. To compare how different geometric properties of word embeddings contribute to model performance on intrinsic and extrinsic evaluations, the team considered 4 attributes of the word embedding geometry: position relative to the origin, distribution of feature values in d dimensions, global pairwise variables, and local pairwise variables.

The team analyzed the performance of affine transformation against other transformations applied to word embedding, including: cosine distance encoding, nearest neighbor encoding, and random encoding. The overall result of affine transformation is better than other transformations. And comparing within the set of affine transformations, the innocuous effect of rotations, dilations, and reflections on both intrinsic and extrinsic tasks suggests that the models used are robust to simple linear transformations.

3. PROBLEM DEFINITION

We define two subproblems:

1. Given a set S of golden triplets (h, r, t) composed of two entities $h, t \in E$ (the set of concepts) and $r \in R$ (set of relationships), and the word embeddings of h and t $w_h, w_t \in \mathbb{R}^k$, we want to find an affine transformation matrix $\boldsymbol{T}$ such that:

   $$\boldsymbol{T}_r w_h \approx w_t$$

   for every r, h, t. We want to minimize the error, in this case how dissimilar is the result of $\boldsymbol{T}_r w_h$ versus w_t. The dissimilarity function can be L_1 or L_2 norm like `TransE` model.

2. Model the task of answering a CommonsenseQA question as a sequence prediction of the relations between the entities of the question and each of the answers. The correct answer is then the concept contained in the most probable sequence. Formally, given the questions' text q, and the entities $e_1, ...e_n$ in it, plus the answer concept a, what is the most probable sequence of affine transformations relating them? i.e.

   $$\boldsymbol{T_m} * ... * \boldsymbol{T_2} * \boldsymbol{T_1} * e_1 = a$$

   such that the i^{th} intermediate step is equivalent to e_{i+0} For simplicity to illustrate this, using the example of Figure 1, the entities would be: ”Something”, ”String”, ”Tie around”, and ”Keep from moving”, and the relationships would be: ”Near”, ”capable of”, ”causes”. This adds explainability to the model in that the *reasoning* the model did is inspectionable.

4. TECHNICAL APPROACH

First we will train one model for each relation. The model will take as input both the head and tail entities' word embeddings and will output the affine transformation matrix, the loss will be calculated as the difference between the transformed head embedding and the tail embedding. The triplets will be extracted from ConceptNet. The perfomance of this encoding can be validated by cross-folding over ConceptNet as the ground truth.We suspect that we can also use this affine transformations as a way to discover new/missing triplets.

In order the answer the CommonsenseQA question in a more efficient and effective way, we want to learn the relation between each entities in the input question. We will adopt *ConceptNet*, a semantic knowledge graph, to create the word embedding vectors which can be further used to learn the meaning of words. We will apply affine geometry transformation such as `TransE` and `TransH` on the word vectors.

5. EXPERIMENTAL EVALUATION

We will present our results on the CommonSenseQA dataset using our proposed training model. We will compare our learning performance with the BERT model [5], which is widely popular for training and solving the CommonSenseQA tasks. We would like to see the difference between their performance and accuracy to see if our proposed model actually make an improvement on solving the CommonSenseQA questions.

We will also experiment regarding some of the initial design choices, for example using or not a hyperplane, using only translation instead of a complete affine transformation, and with different embeddings (like orGloVe, Word2Vec).

6. CONTRIBUTION

Team members include: Eduardo Hidalgo, and Dan Luo.

Task	Name
Algorithm implementation	Dan, Eduardo
Fine-tuning	Dan, Eduardo
Model testing	Dan, Eduardo
Output Evaluation	Dan, Eduardo

7. REFERENCES

[1] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *CoRR*, vol. abs/1310.4546, 2013. [Online]. Available: http://arxiv.org/abs/1310.4546

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 2787–2795. [Online]. Available: http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf

[3] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes." in *AAAI*, 2014, pp. 1112–1119.

[4] B. Whitaker, D. Newman-Griffis, A. Haldar, H. Ferhatosmanoglu, and E. Fosler-Lussier, "Characterizing the impact of geometric properties of word embeddings on task performance," *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for*, Apr 2019.

[5] N. F. Rajani, B. McCann, C. Xiong, and R. Socher, "Explain yourself! leveraging language models for commonsense reasoning," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. [Online]. Available: http://dx.doi.org/10.18653/v1/p19-1487