Analogical Reasoning with Knowledge-based Embeddings

Douglas Summers-Stay, Dandan Li
U.S. Army Research Laboratory
douglas.a.summers-stay.civ@mail.mil, happydandan2016@gmail.com

Abstract

For robots to interact with natural language and handle real-world situations, some ability to perform analogical and associational reasoning is desirable. Consider commands like "Fetch the ball" vs. "Fetch the wagon", the robot needs to know that carrying a ball is (in the appropriate sense) analogous to dragging a wagon. Without the ability to perform analogical reasoning, robots are incapable of generalizing in the ways that true natural language understanding requires. Inspired by implicit Verlet integration methods for mass spring systems in physics simulations, we present a novel knowledge-based embedding method in this paper, where distributional word representations and semantic relations derived from knowledge bases are incorporated. We use some SAT-style analogy questions to demonstrate potential feasibility of our approach on the analogical reasoning framework.

Introduction

Attempts at analogical reasoning using knowledge bases have only been successful in very limited, carefully arranged scenarios. Semantic vector representations, in contrast, have shown a surprisingly sophisticated level of analogy formation. In a semantic vector space, concepts that are related to each other in meaning have vectors that are nearby (in an appropriate distance metric, such as cosine similarity). This property directly gives rise to their ability to form analogies. Consider the analogy *sculptor:chisel::painter:paintbrush*, *sculptor* is related to other types of *artist*, and it is also related to terms having to do with *stone*. The fact that it must be near both these terms forces the word to be found somewhere near the midpoint of a line connecting the vectors for *artist* and *stone*: *sculptor* $\approx$ *(artist + stone)/2*. In the cosine similarity metric, division by a constant can be neglected, so we can just write *sculptor* $\approx$ *artist + stone*. Performing a similar decomposition on the other terms and neglecting some noise-like variation, we get the new analogy *artist+stone : tool+stone :: artist+paint : tool+paint*. This new representation makes it easy to see that adding the middle two terms and subtracting the first term: *[tool+stone] + [artist+paint] - [artist+stone]* must equal the fourth term *[tool+paint]*. Note that this isn't the only decomposition we could have performed. A different analogy involving *sculptor* might depend on the ability to decompose *sculptor* into *stonecarving+worker*, or some other decomposition. Additionally, these analogical relations should hold from any vector representation where semantically similar entities have similar vectors. This means that we could potentially use vectors derived from weights in deep learning networks trained on vision, depth, or other modalities, as well as language.

Relational knowledge is commonly stored in knowledge bases such as ConceptNet as triples of the form head, relation, tail. Knowledge extraction tools are also capable of extracting such triples directly from natural language sources. The semantic arithmetic above can be restated as a geometric constraint: the vector connecting two terms that share a particular relation should be approximately equal to the vector connecting two other terms that share the same relation. This suggests a way of incorporating knowledge base triples into a semantic vector space– match the knowledge base entities to their corresponding vector representations, and ensure that the relation vector connecting the head entity to the tail is similar for all examples of the relation. The method we explore in this paper begins with a set of distributional semantic vectors and attempts to modify these vectors such that the constraint holds.

Related Work

Vector space models have a long, rich history in the field of natural language processing, where each word is represented as a real-valued vector in a continuous vector space and the relationships between words can be encoded by vector operations. There are mainly three families for learning word vectors: (1) global matrix factorization methods, such as latent semantic analysis, which generates embeddings from term-document matrices by singular value decomposition (Deerwester et al., 1990). (2) neural network models, such as the skip-gram and continuous bag of words models of (Mikolov et al., 2013a, 2013b), referred to as word2vec, which learn embeddings by training a network to predict neighboring words. Mikolov et al.(2013c) demonstrate that the embeddings created by a recursive neural network encode not only attributional similarities between words, but also similarities between pairs of words. (3) knowledge graph embeddings: there are three main types of models for knowledge graph embeddings. TransE (Bordes et al., 2013) and its various improvements (Wang et al., 2014; Lin et al., 2014; Ji et al., 2015; Jia et al., 2016; Nguyen et al., 2016) learn vector embeddings of the entities and the relationships directly based on semantic relations in knowledge bases. Other methods (Fried and Duh., 2014) use relational knowledge as additional objectives when creating the distributional semantic vectors in the first place. A third set of models (Faruqui et al., 2014; Mrkšić et al., 2016, and Speer et al, 2017) begins with a set of distributional semantic vectors and attempts to modify these vectors such that the relations in knowledge bases holds. The method we explore in this paper falls into the third camp.

Algorithm

Modifying the vectors should be done in such a way that existing relationships between vectors are preserved as much as possible, to take full advantage of the information already implicitly contained in the geometry. Our approach is inspired by implicit Verlet integration methods for mass spring systems in physics simulations (Baraff and Witkin, 1998). We treat the vector space as if it were a warped soft body, allowing it to gradually evolve towards a solution where the difference between our objective and the vectors in the space is minimized, while maintaining as much local structure as possible. The iterative relaxation is intended to allow multiple constraints on particular entities to balance out to a lowest energy solution.

Algorithm 1 Generating Knowledge-based Word Embeddings.

```
Input: normalized word vectors based on the word2vec model.
Ouput: knowledge-based word embeddings.
// The point is to gradually move the vectors closer to having the same displacement vector for
each example of a relation.
Procedure:
for n = 1 to 100 do
    // For each pair, let b and c be the head and tail vectors of a related pair, and d be the average
    of all vectors sharing the same relation as b has with c.
    f =1/(n+1);
    find the midpoint m of the vector connecting b and c;
    create a new point b' =b*(1-f) + (m-d/2)*(f) and a new point c' =c*(1-f)+(m+d/2)*(f)
      assign b' to b and c' to c.
      // Recalculate all of the averaged relation vectors (d)
end for
```

Analysis

Figure 1 shows the difference between word2vec-based vectors (W2V) and our knowledge-based vectors (KB) of 1000000 words. The x-axis of the figure denotes "word count" from word2vec model and the y-axis denotes "L2-norm of the vector difference between normalized W2V and KB". The bi-dimensional histogram of data gives an in-depth description of word distribution. We set bins=[100,100] and the returned histogram describes the word density. In our experiment, we compute the natural logarithm of the number of words (i.e., log(Density)). From this figure, we can see that in our collection the ranges [2000000, 3000000] * [0, 1e-6] and [2000000, 3000000] * [0.4, 0.7] have high word density.

We give two SAT questions in Table 1 and Table 2 to demonstrate how our knowledge-based embeddings work. For target words "petrified" and "doting" in the SAT questions, their L2-norms of vector difference between W2V and KB are relatively large (0.445 and 0.548 respectively). For each target pair, it has five option pairs. The pair with ** is annotated as correct answer. We compute the directional similarity measure of each option's relationship to the target's relationship and select the option with the highest score as a guess answer. For these two examples, our model can recognize the correct answer while word2vec-based model fails in its prediction.

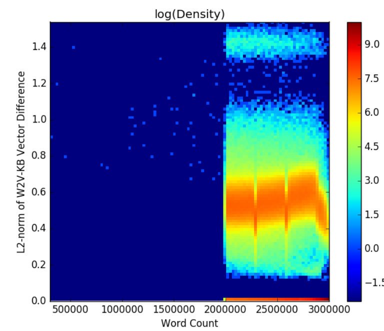

Figure 1: The difference between W2V and KB vectors

apprehensive:petrified	W2V Similarity	KB Similarity
happy:hostile	-0.0138	0.002
elated:deaden	**0.181**	0.136
sincere:satisfied	-0.063	-0.085
emotional:accord	-0.008	-0.039
cheerful:hilarious	0.171	**0.168**

Table 1: W2V-based Similarity and KB-based Similarity on SAT question *apprehensive:petrified*

fond:doting	W2V Similarity	KB Similarity
polite:servile	0.059	0.059
whisper:scream	-0.005	0.009
grasping:needy	0.032	**0.066**
abstinent:indulgent	**0.078**	0.057
no:choice	0.019	0.037

Table 2: W2V-based Similarity and KB-based Similarity on SAT question *fond:doting*

Conclusion and Future Work

In this work, we explore a novel knowledge-based embedding method based on Verlet integration methods for mass spring systems in physics simulation. Different from previous methods in the details of how the retrofitting is performed, we are working with over a thousand relations derived from ResearchCyc, ConceptNet, and WordNet combined. We plan to continue improving our algorithm and evaluating its performance on large datasets. Our eventual goal is to incorporate the semantically embedded knowledge base in analogical and associational reasoning of a robotic system.

References

Baraff, D., and Witkin, A. 1998. Large Steps in Cloth Simulation. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques,* 43–54.

Bordes, A., Usunier, A., Garcia-Duran, A., Weston, J., and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26 (NIPS),* 2787–2795.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. 1990. Indexing by Latent Semantics Analysis. *Journal of the Association for Information Science and Technology,* 41(6):391–407.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., and Smith, N. A. 2015. Retrofitting Word Vectors to Semantic Lexicons. In *The 2015 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*. Denver, Colorado, USA.

Fried, D., and Duh, K. 2014. Incorporating Both Distributional and Relational Semantics in Word Representations. In *arXiv preprint arXiv:1412.4369*.

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence,* 218–2187.

Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. 2015. Knowledge Graph Embedding via Dynamic Mapping Matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,* 687–696. Beijing, China.

Jia, Y., Wang, Y., Lin, H., Jin, X., and Cheng, X. 2016. Locally Adaptive Translation for Knowledge Graph Embedding. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence,* 992–998.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. 2013a. Efficient Estimation of Word Representations in Vector Space. In *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. 2013b. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems,* 3111–3119. Nevada, USA.

Mikolov, T., Yih, W.-T., and Zweig, G. 2013c. 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,* 746–751. Atlanta, Georgia, USA.

Mrkšić, N., Séaghdha, D. Ó., Thomson, B., Gašić, M., Rojas-Barahona, L., Su, P-H., Vandyke, D., Wen, T-H., Young, S. 2016. Counter-fitting Word Vectors to Linguistic Constraints. In *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies,* 142–148. San Diego, California.

Nguyen, D. Q., Sirts, K., Qu, L., and Johnson, M. 2016. STransE: A Novel Embedding Model of Entities and Relationships in Knowledge Bases. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 460–466.

Speer R, Chin J, and Havasi C. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. 2017. In *The Thirty-First AAAI Conference on Artificial Intelligence,* 4444–4451. San Francisco.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In *Association for the Advancement of Artificial Intelligence,* 1112–1119.