Improving Analogical Inference Using Vector Operations with Adaptive Weights

Tatsuhiko Kato (P)[1], and Shohei Hidaka[1]

[1] Japan Advanced Institute of Science and Technology
E-mail: skylark@jaist.ac.jp

Abstract— Recent studies have shown that a class of vector-space models of words has a potential to perform analogical inference at human-level. In these models, the analogical relationship is captured by some vector operations, and the performance of the inference depends on the choice of the operation. In this study, we extend the vector operations proposed in the past study by adaptively tuning the weghts of word vectors. With the proposed method, we obtained some substantial improvement in the analogical inferences.

Keywords— Vector-space models, Neural Network, Analogy

1 Introduction

In the field of natural language processing (NLP), it is important to represent words in a meaningful manner. In order to conduct any task such as sentiment analysis or machine translation, one needs some kind of similarity measure between linguistic entities. One effective way to define similarities between words is to represent each word to be a point in a high-dimensional (Euclidean) space, so that the distribution of words in the space reflects statistical property of words (e.g., co-occurrence probability) present in the corpus. In such models, typically distance between word-points in the space is supposed to reflect some kind of relationship among words. This class of models is called vector-space model (VSM) or word embedding.

The four-word analogical inference task has been used as a standard benchmark test for the word representation models. In the task, one is given a triplet of words in the form ($man : woman :: king : x$), and is asked to answer the missing fourth word x. In the VSM, the model answers the fourth word by applying some similarity operator to the given triplet of the three words for each question.

Mikolov et al. [6, 5] has proposed a simple method for solving the analogy task and showed that their models (Skip-gram and Continuous Bag of Words, in comibination known as word2vec) perform the task surprisingly well, beating previous models by large margin. Mikolov et al's method is to employ a cosine similarity between an additive operation on the vectors of three words and of the candidate word, selecting a vector which is most similar in the measure (more details in the next section). Improving on this method, Levy and Goldberg [3] has proposed a method based on a multipilicative operation, which achieves the state-of-the-art performance for the task [4].

Both methods are based only on the similarity of word vectors in the given space. We present a novel supervised-learning method for making analogical inference by adaptively weighting dimensions in favor of a given set of words in an analogical relationship. Our experiment showed significant increase in accuracy over both methods mentioned above.

2 Analogy and vector-space models

Skip-gram is one of well-studied models of word embedding. Since in this paper we test our methods specifically on the word vectors learned through this model, here we briefly introduce how Skip-gram learns the vector representation of words.

Skip-gram is a single-layer neural network model, which takes a word vector (represented as one-hot vector) as input and tries to maximize the prediction accuracy in output layer for the words around the input word in the corpus. The model optimizes vector representation v_j so that it maximizes the conditional probability of t^{th} word given the other words occurring within the time window of size c:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log P(v_{t+j}|v_t),$$

where the the probability function P is given by the soft-max function of cosine similarity of given words.

2.1 Analogical inference in VSM

A straightforward method for solving the analogical inference task is the one given by Mikolov et al. [6, 5].

$$f(v_a, v_b, v_c) = \underset{v_d\notin\{v_a,v_b,v_c\}}{\arg\max}\left(\cos(v_c - v_a + v_b, v_d)\right) \quad (1)$$

The method takes the offset of given word vectors and find the word vector which has the highest cosine similarity to the offset term. Levy and Goldberg [3] proposed another method to solve the analogy task:

$$f(v_a, v_b, v_c) = \underset{v_d\notin\{v_a,v_b,v_c\}}{\arg\max}\frac{\cos(v_b, v_d)\cos(v_c, v_d)}{\cos(v_a, v_d)} \quad (2)$$

The method has currently the best accuracy in analogy tasks[4]. Both methods are based on the idea that to get the desired result for analogy task, make the v_c vector as far from v_a and as near to the v_b. Using the same example as above, make the v_{king} vector as far from v_{man} and as near to v_{woman}.

3 Adaptive weighting of dimensions preserving the analogical relationship

Our method utilizes the supervised learning to select a subspace in which analogical inference is more accurate for a given test set. We define the analogical

inference function as follows:

$$g_{M_1,M_2}(v_a, v_b, v_c) = \underset{v_d \notin \{v_a, v_b, v_c\}}{\arg\max} (\cos(M_1 v_c - M_1 v_a + M_2 v_b, M_2 v_d)) \quad (3)$$

Here M_i is some linear transformation. The formulation can be viewed as a generalized version of method (1), since if we take $M_1 = M_2 = I$, where I is identity matrix, (3) corrresponds exactly to (1). The important part of our method is the selection process of M_i, since arbitrarily selecting M_i doesn't make analogical inference more accurate.

Before explaining our selection process, let us state some preriliminaries. First, a large number of word vectors learned through Skip-gram take values near zero, and fewer words take large values in the same dimension. As only one word is the "correct" answer in any analogy task, the vast majority of other words is a "noise". The empirical distribution of this "noise" words in a particular dimension follows an exponential distribution. Therefore, a rule of thumb derived from this observation is to choose some dimensions on which many words in the test set have larger absolute values.

Secondly, we choose some subspace in which the words in the test set form the "analogical relationship" which the analogy inference function (3) will identify as the answer. For D dimensional vector space of N words, let $\mathbf{V}_0 \in \mathbb{R}^{K \times D}$ be a matrix of K words, in which each row has a word vector of some category, such as man and king ("male" matrix), and let $\mathbf{V}_1 \in \mathbb{R}^{K \times D}$ be a matrix paired with $\mathbf{V}_0$, in which i^{th} row has a vector corresponding i^{th} row in $\mathbf{V}_0$, such as woman and queen ("female" matrix). Then, for the model applying the function (3), the dimension j which has smaller error ϵ_j defined as $\epsilon_j = \|\mathbf{V}_{0,j} - \mathbf{V}_{1,j} + \mathbf{1}_K c^\top\|$, is more preferable, where $c \in \mathbb{R}^D$ is some translation vector minimizing ϵ_j with respect to c, and $\mathbf{1}_K \in \mathbb{R}^K$ is the vector with its all elements being 1. Taking a subspace of dimension with $\epsilon_j = 0$, the analogy inference of the type (1) exactly identifies the correct answer for the given triplet.

Summarizing two general preferences to have a better vector space:

1. choose dimensions in which words in test set have larger absolute values
2. choose the dimension i in which words in test set have lower ϵ_i

Considering the two conditions, we derived a weight for dimension i below:

$$w_i = e^{\max(|\mathbf{V}_{0,i}|) + \max(|\mathbf{V}_{1,i}|) - \epsilon} \quad (4)$$

In (4), $\max(|\mathbf{V}_{0,i}|) + \max(|\mathbf{V}_{1,i}|)$ part reflects the condition 1 by taking the maximum of an absolute value of dimensions for each word, and subtracting ϵ reflects the condition 2. In the following experiment, we used the pretrained word vectors [1] which contain three millions of words with each having 300 dimensions. As analogy set, we utilized widely used Google test set[2], which contains 19544 pairs of analogy questions (8,869 semantic and 10,675 syntactic questions). We calculated weights by (4) and applied weights to 300 dimensions of 300 million word vectors, then obtained the model answer by (3) with weighted vectors.

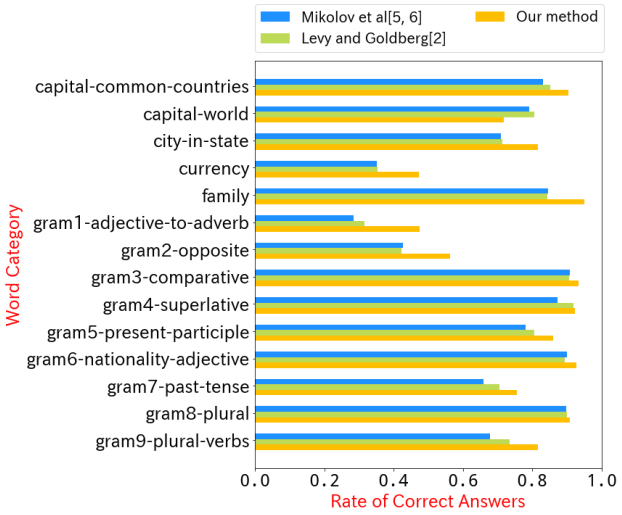

Figure 1: The inferential accuracy (x-axis) for each word catgory (y-axis) for each model.

4 Results and Discussion

The result of our experiment is shown in Figure 1. We compared our method with Mikolov et al's(1) and Levy and Goldbergs(2) methods by the number of correctly answered questions divided by number of questions in a category. Overall, our method improved analogy accuracy 10-20% compared to both methods. Accuracies in all questions are shown below.

Mikolov et al[6, 5]	Levy and Goldberg[3]	Our method
0.736	0.752	**0.783**

The result shows significant improvement of accuracy in both methods using our weight. This implies that taking a suitable subspace of word vectors is a promising way to improve analogy performance. Note that our weight is an approximation of (4), and likely not to be optimal result. More rigorous derivation of (4) will provide the better weight.

References

[1] https://code.google.com/archive/p/word2vec/.

[2] http://download.tensorflow.org/data/questions-words.txt.

[3] Omer Levy and Yoav Goldberg. Linguistic Regularities in Sparse and Explicit Word Representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180, April 2014.

[4] Tal Linzen. Issues in Evaluating Semantic Spaces Using Word Analogies. In *Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 13–18, 2016.

[5] T M Mikolov, K Chen, G Corrado, and J Dean. Efficient Estimation of Word Representations in Vector Space . *ICLR Workshop Papers*, pages 1–12, 2013.

[6] T M Mikolov, I Sutskever, K Chen, G Corrado, and J Dean. Distributed Representations of Words and Phrases and their Compositionality. *NIPS*, pages 1–9, 2013.