

Semantic Compositionality through Recursive Matrix-Vector Spaces

Richard Socher, Brody Huval, Christopher D. Manning, Andrew Y. Ng

Presented by Zhaoyang Guo

Semantic Word Vector Spaces

- Search query expansions
- Fact extraction for information retrieval
- Automatic annotation of text with disambiguated Wikipedia links

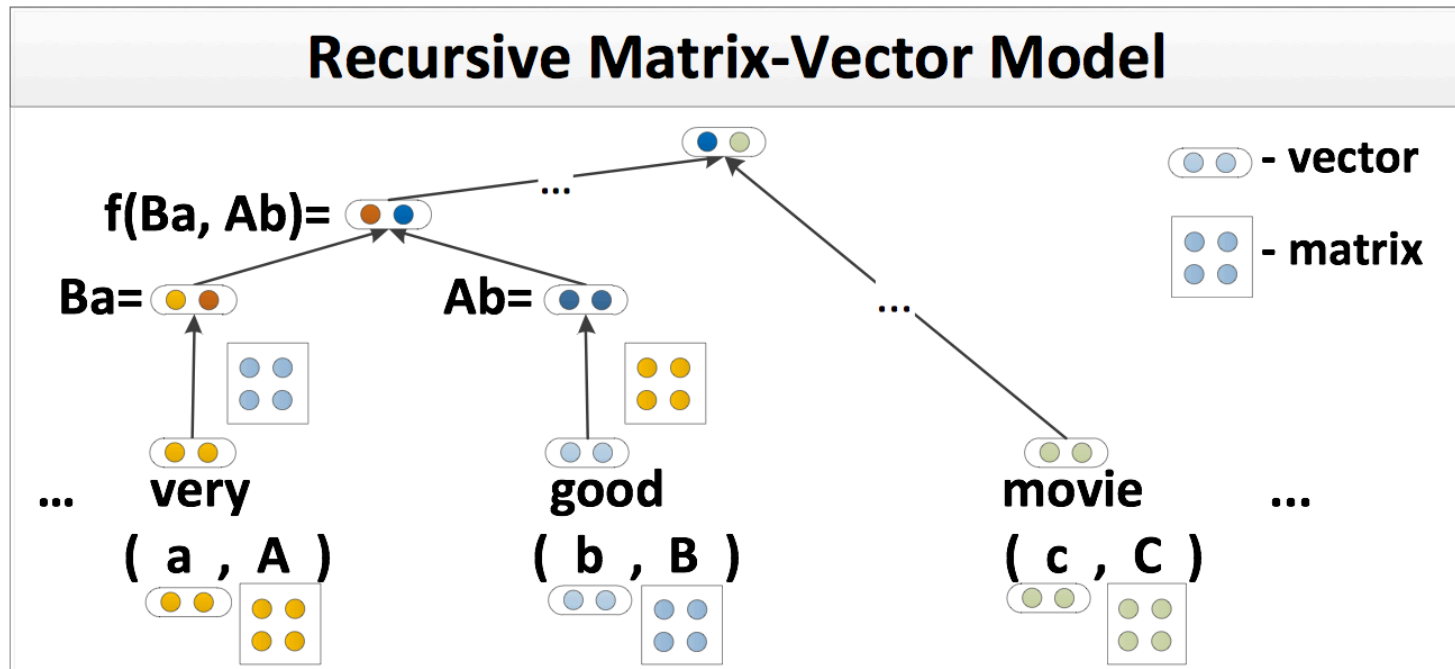
Compositionality

- Compositional meaning of longer phrases
- Deeper understanding of language

Learn Compositional Vector Representations

- Various types of phrase
- Sentences of arbitrary length

MV-RNN Model (Matrix-Vector Recursive Neural Network)



Existing Approaches

- Linear combination of single word representations
 - Sum
 - Weighted average
 - Multiplication
- Tensor product (outperformed by weighted addition and multiplication)
- Concatenation
- $p = Ab$ (Baroni and Zamparelli, 2010)
- Standard RNN (Socher et al., 2011c)
- Linear MVR (Mitchell and Lapata, 2010; Zanzotto et al, 2010)

Standard RNN (Recursive Neural Network)

$$p = g \left(W \begin{bmatrix} a \\ b \end{bmatrix} \right)$$

A **global matrix W** that multiplied the word vectors (a, b) , and a nonlinearity function g (such as a sigmoid or tanh)

Linear MVR (Linear Matrix-Vector Recursion model)

$$p = Ba + Ab$$

Linear combination

$$W = [I \ I]$$

$$g(x) = x$$

Existing Approaches

- Linear combination of single word representations
 - Sum
 - Weighted average
 - Multiplication
- Tensor product (outperformed by weighted addition and multiplication)
- Concatenation
- $p = Ab$ (Baroni and Zamparelli, 2010)
- **Standard RNN (Socher et al., 2011c)**
- **Linear MVR (Mitchell and Lapata, 2010; Zanzotto et al, 2010)**

MV-RNN

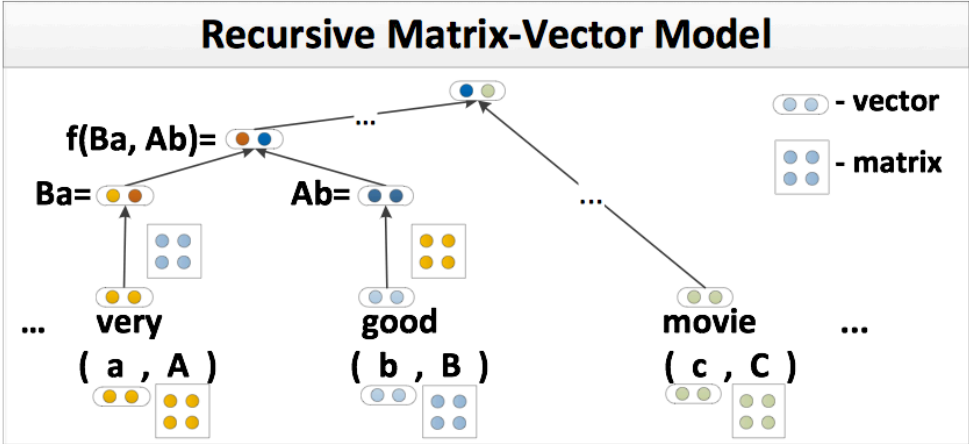
Mitchell and Lapata (2010) give as their most general function: $p = f(a, b, R, K)$, where R is the a-priori known syntactic relation and K is background knowledge

MV-RNN

Mitchell and Lapata (2010) give as their most general function: $\mathbf{p} = f(\mathbf{a}, \mathbf{b}, \mathbf{R}, \mathbf{K})$, where ~~\mathbf{R} is the a priori known syntactic relation and \mathbf{K} is background knowledge~~

- There is a constraint on \mathbf{p} which is that it has the same dimensionality as each of the input vectors
- Capture semantic/syntactic relation implicitly via the learned matrices

MV-RNN Dimensions



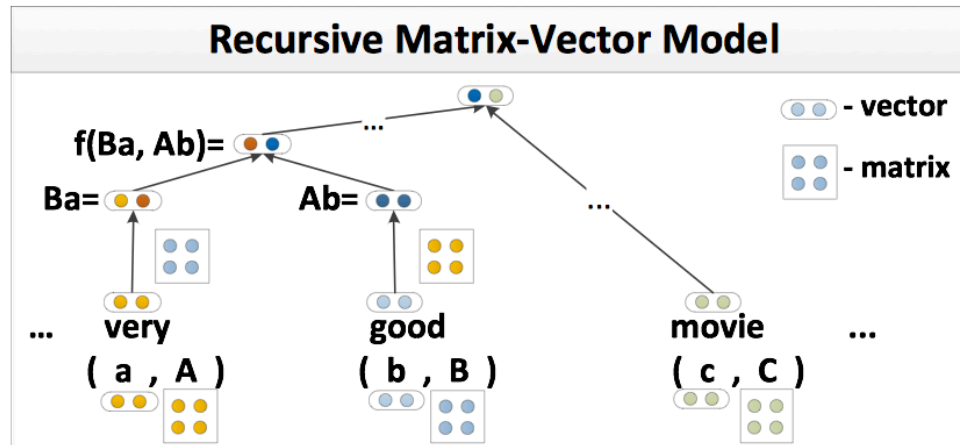
$$p = f_{A,B}(a, b) = f(Ba, Ab) = g \left(W \begin{bmatrix} Ba \\ Ab \end{bmatrix} \right)$$

$$x \in \mathbb{R}^n$$

$$X \in \mathbb{R}^{n \times n}$$

$$W \in \mathbb{R}^{n \times 2n}$$

MV-RNN Dimensions

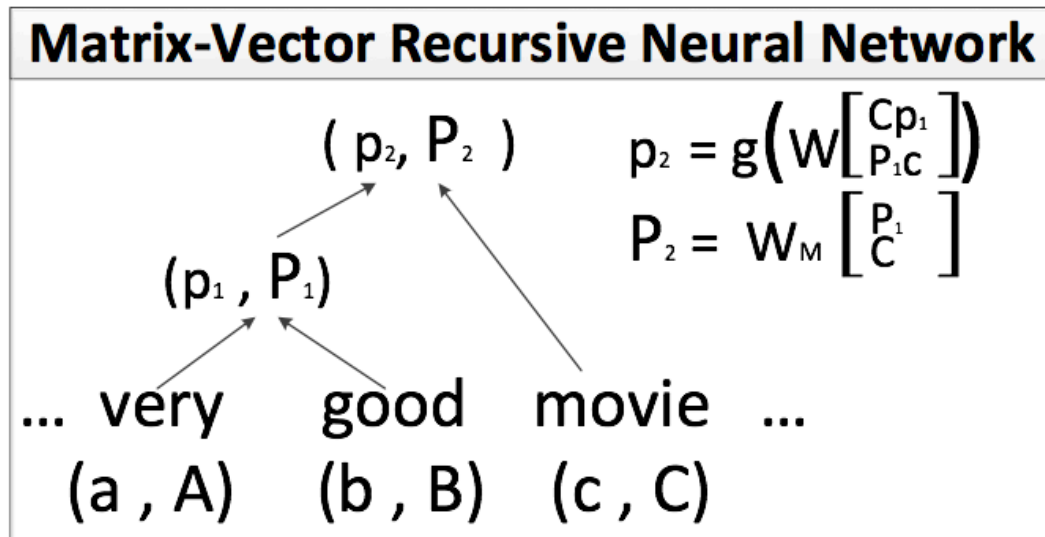


$$P = f_M(A, B) = W_M \begin{bmatrix} A \\ B \end{bmatrix}$$

$$W_M \in \mathbb{R}^{n \times 2n}$$

$$P \in \mathbb{R}^{n \times n}$$

MV-RNN Example



$$p_1 = f(Ba, Ab)$$

$$p_2 = f(Cp_1, P_1c)$$

$$P_1 = f_M(A, B)$$

$$P_2 = f_M(P_1, C)$$

MV-RNN Initialisation

- Initialize all word vectors $x \in \mathbb{R}^n$ with pre-trained 50-dimensional word vectors from the unsupervised model of Collobert and Weston (2008)
- Initialize matrices as $X = I + \varepsilon$, i.e., the identity plus a small amount of Gaussian noise
- Represent any phrase or sentence of length m as an ordered list of vector- matrix pairs $((a, A), \dots, (m, M))$, where each pair is retrieved based on the word at that position

MV-RNN Training

$$p = f_{A,B}(a, b) = f(Ba, Ab) = g \left(W \begin{bmatrix} Ba \\ Ab \end{bmatrix} \right)$$

- Rewriting the two transformed vectors as one vector z , we get $p = g(Wz)$ which is a single layer neural network
- Add on top of each parent node a simple softmax classifier to predict a class distribution over, e.g., sentiment or relationship classes: $d(p) = \text{softmax}(W^{label}p)$.
If there are K labels, then $d \in \mathbb{R}^K$ is a K -dimensional multinomial distribution

MV-RNN Training

(See Socher et al., 2010)

- Error function: $E(s, t, \theta)$
- The sum of cross-entropy errors at all nodes
- Where s : sentence, t : tree
- Parameters: $\theta = (W, W_M, W^{label}, L, L_M)$
- Learning function:

$$\frac{\partial J}{\partial \theta} = \frac{1}{N} \sum_{(x,t)} \frac{\partial E(x, t; \theta)}{\partial \theta} + \lambda \theta$$

- Low-rank matrix approximation

$$A = UV + \text{diag}(a)$$

where $U \in \mathbb{R}^{n \times r}$, $V \in \mathbb{R}^{r \times n}$, $a \in \mathbb{R}^n$ and we set the rank for all experiments to $r = 3$.

Evaluation and Generality

- Most related work compares similarity judgments of unsupervised models to those of human judgments and aims at high correlation
- The question remains how these models would perform on downstream NLP tasks such as sentiment detection

Evaluation and Generality

- Initializing the models with these general representations, did not improve the performance on the tasks we consider.
- For sentiment analysis, this is not surprising since antonyms often get similar vectors during unsupervised learning from co-occurrences due to high similarity of local syntactic contexts.
- In order to fairly compare to related work, we use only the supervised data of each task.

Predicting Sentiment Distributions of Adverb-Adjective Pairs

- IMDB dataset: extract adverb-adjective pairs from movie reviews
- The dataset provides the distribution over star ratings: Each consecutive word pair appears a certain number of times in reviews that have also associated with them an overall rating of the movie.
- Only word pairs that appear at least 50 times are kept .

Predicting Sentiment Distributions of Adverb-Adjective Pairs

- We never give the algorithm sentiment distributions for single words, and, while single words overlap between training and testing, the test set consists of never before seen word pairs.
- The softmax classifier is trained to minimize the cross entropy error

Predicting Sentiment Distributions of Adverb-Adjective Pairs

- Evaluation: KL-divergence

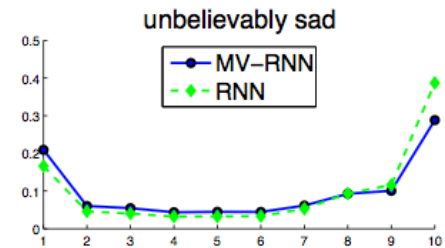
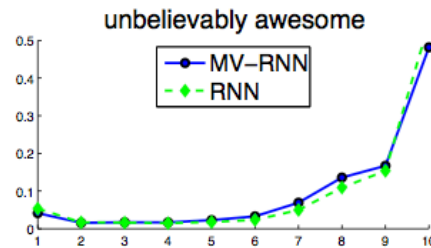
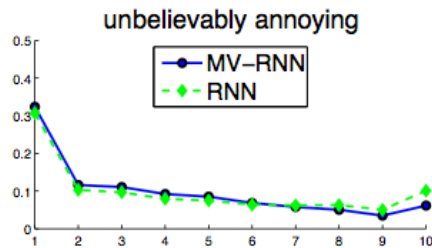
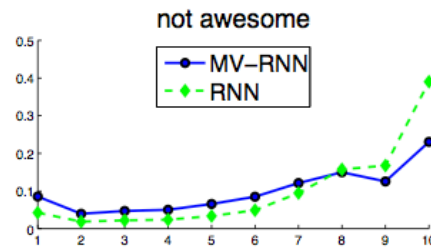
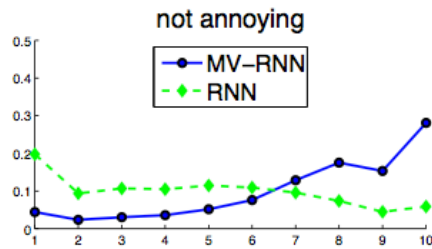
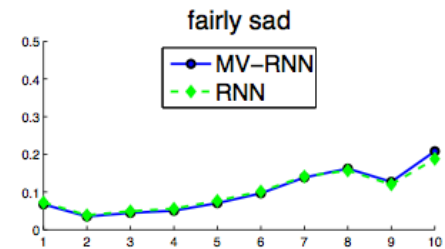
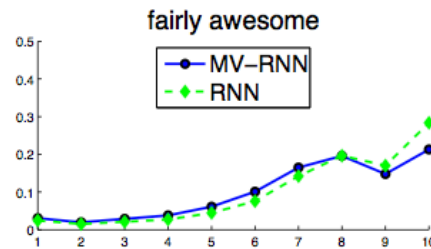
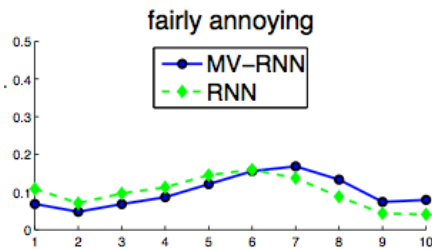
$$KL(g||p) = \sum_i g_i \log(g_i/p_i)$$

where g is the gold distribution and p is the predicted one

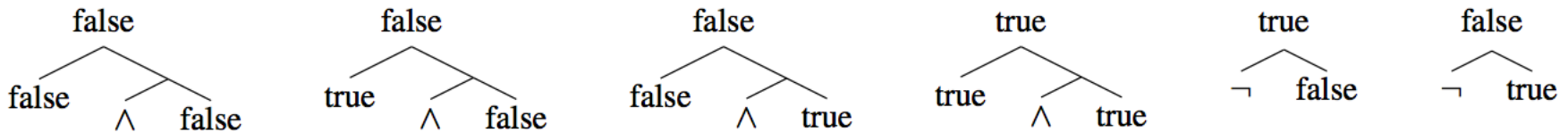
Predicting Sentiment Distributions of Adverb-Adjective Pairs

Method	Avg KL
Uniform	0.327
Mean train	0.193
$p = \frac{1}{2}(a + b)$	0.103
$p = a \otimes b$	0.103
$p = [a; b]$	0.101
$p = Ab$	0.103
RNN	0.093
Linear MVR	0.092
MV-RNN	0.091

Predicting Sentiment Distributions of Adverb-Adjective Pairs



Logic- and Vector-based Compositionality



$$\mathit{true} \quad (t = 1, T = 1)$$

$$\mathit{false} \quad (f = 0, F = 1)$$

$$\neg \mathit{false}: \min \left(\|p_{top} - t\|^2 + \|P_{top} - T\|^2 \right)$$

$$g(x) = \max(\min(x, 1), 0)$$

Predicting Movie Review Ratings

Method	Acc.
Tree-CRF (Nakagawa et al., 2010)	77.3
RAE (Socher et al., 2011c)	77.7
Linear MVR	77.1
MV-RNN	79.0

S.	C.	Review sentence
1	✓	The film is bright and flashy in all the right ways.
0	✓	Not always too whimsical for its own good this strange hybrid of crime thriller, quirky character study, third-rate romance and female empowerment fantasy never really finds the tonal or thematic glue it needs.
0	✓	Doesn't come close to justifying the hype that surrounded its debut at the Sundance film festival two years ago.
0	x	Director Hoffman, his writer and Kline's agent should serve detention.
1	x	A bodice-ripper for intellectuals.

Classification of Semantic Relationships

- The previous task considered global classification of an entire phrase or sentence
- MV-RNN can also learn how a syntactic context composes an aggregate meaning of the semantic relationships between words
- The task is finding semantic relationships between pairs of nominals.
- We use the dataset and evaluation framework of SemEval-2010 Task 8 (Hendrickx et al., 2010). There are 9 ordered relationships (with two directions) and an undirected *other* class, resulting in 19 classes.

Classification of Semantic Relationships

Relationship	Sentence with labeled nouns for which to predict relationships
Cause-Effect(e2,e1)	Avian [influenza] _{e1} is an infectious disease caused by type a strains of the influenza [virus] _{e2} .
Entity-Origin(e1,e2)	The [mother] _{e1} left her native [land] _{e2} about the same time and they were married in that city.
Message-Topic(e2,e1)	Roadside [attractions] _{e1} are frequently advertised with [billboards] _{e2} to attract tourists.
Product-Producer(e1,e2)	A child is told a [lie] _{e1} for several years by their [parents] _{e2} before he/she realizes that ...
Entity-Destination(e1,e2)	The accident has spread [oil] _{e1} into the [ocean] _{e2} .
Member-Collection(e2,e1)	The siege started, with a [regiment] _{e1} of lightly armored [swordsmen] _{e2} ramming down the gate.
Instrument-Agency(e2,e1)	The core of the [analyzer] _{e1} identifies the paths using the constraint propagation [method] _{e2} .
Component-Whole(e2,e1)	The size of a [tree] _{e1} [crown] _{e2} is strongly correlated with the growth of the tree.
Content-Container(e1,e2)	The hidden [camera] _{e1} , found by a security guard, was hidden in a business card-sized [leaflet box] _{e2} placed at an unmanned ATM in Tokyo's Minato ward in early September.

Classification of Semantic Relationships

- Many approaches use features for all words on the path between the two words of interest. We show that by building a single compositional semantics for the minimal constituent including both terms one can achieve a higher performance.
- MV-RNN only needs a parser for the tree structure and learns all semantics from unlabeled corpora and the training data.
- Only the SemEval training dataset is specific to this task, the remaining inputs and the training setup are the same as in previous sentiment experiments.

Classification of Semantic Relationships

Classifier	Feature Sets	F1
SVM	POS, stemming, syntactic patterns	60.1
SVM	word pair, words in between	72.5
SVM	POS, WordNet, stemming, syntactic patterns	74.8
SVM	POS, WordNet, morphological features, thesauri, Google <i>n</i> -grams	77.6
MaxEnt	POS, WordNet, morphological features, noun compound system, thesauri, Google <i>n</i> -grams	77.6
SVM	POS, WordNet, prefixes and other morphological features, POS, dependency parse features, Levin classes, PropBank, FrameNet, NomLex-Plus, Google <i>n</i> -grams, paraphrases, TextRunner	82.2
RNN	-	74.8
Lin.MVR	-	73.0
MV-RNN	-	79.1
RNN	POS, WordNet, NER	77.6
Lin.MVR	POS, WordNet, NER	78.7
MV-RNN	POS, WordNet, NER	82.4

Classification of Semantic Relationships

- In order to see whether our system can improve over this system, we added three features to the MV-RNN vector and trained another softmax classifier. The features and their performance increases were POS tags (+0.9); WordNet hypernyms (+1.3) and named entity tags (NER) of the two words (+0.6).

Conclusion

- Introduce a complete treatment of compositionality in word vector spaces
- Based on a syntactically plausible parse tree
- The combination of matrix-vector representations with a recursive neural network
- learn both the meaning vectors of a word and how that word modifies its neighbors (via its matrix)
- generalizes several models in the literature (propositional logic, sentiment and semantic relationships between nouns in a sentence)

Thanks for listening!