Polynomial Semantic Representation

Mincong Luo
State Key Lab of Information Security
Institute of Information and Engineering,Chinese Academy of Sciences
Peking,China
luominzhong@iie.ac.cn

Abstract

The current natural language processing methods from mainstream all rely on vector representation and neural network fitting, but their models have unexplainability and lack of advanced reasoning capabilities. In this paper, we propose a polynomial semantic representation based on the theory of ring,and the lack of current mainstream methods is reinforced. Detailed design principles is explained and the proposed model is evaluated on the Stanford Natural Language Inference(SNLI) dataset, and comparison with the baseline models is also given in the article.

1 introduction

In recent years, natural language processing has made impressive progress in machine translation, relation extraction, concept learning, and knowledge reasoning Binder & R. Yu et al. (2018) Mordatch (2018).It seems that end-to-end methods based on deep learning have become a must-have standard configuration, but more and more studies have revealed that learning methods based on numerical representations and neural networks have inherent defects Shalev-Shwartz et al. (2017) Vialatte & Leduc-Primeau (2018).In this paper, to address the shortcomings of existing representation learning methods, we propose a semantic representation method based on ring theory and polynomials, which has the following characteristics:

- **Distinguish between pronouns and predicates:**In fact, in linguistics, pronouns have "operability" Cook et al. (1988) dominated by predicates, whether they are subjects or passive objects. This can be analogized to the relationship between the preimage and mapping in mathematics; but in the mainstream method words are represented as vectors with the same dimension Pan et al. (2018) Zhang et al. (2019);
- **Word representation also owns a mapping function:**In the mainstream method, all word representations come from pre-trained corpus-based word vectors, which owns the advantages of transferability and continuous representation and easy calculation Gururangan et al. (2018) Kim et al. (2018),But it also has the disadvantages of unexplainability and lack of flexibility; while the polynomial-based word representation is also a mapping with arguments, so it has multiple functions of representation and mapping, and is more flexible in calculation.
- **Stronger generalization ability:**The mainstream methods that rely on vector representation and neural networks require a large amount of corpus to perform gradient descent to make the model converge, so the sample utilization is not efficient Shalev-Shwartz et al. (2017).The method proposed in this paper will combine the dividing mechanism in algebra and sentence semantic for reasoning, leads to better generalization;

A ring is one type of algebraic system that includes two types of operations(addition and multiplication).It is an abstraction of some actual operation systems such as integer operations, matrix

* indicates equal contribution.

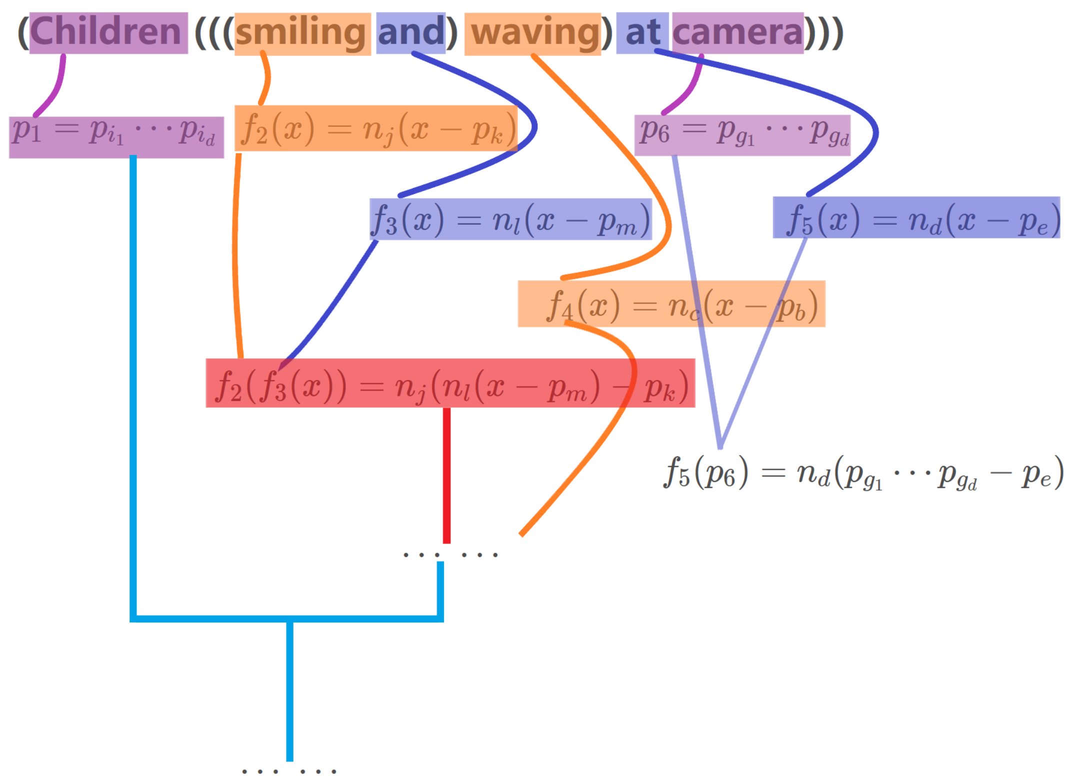

Figure 1: The process of a sentence represented by polynomial semantic method.

operations, and polynomial operations, and is also a very important researching section in modern algebra. The purpose of introduceing the ring in this paper is to shield the details related to numerical operations, and to replace a large number of floating calculations with pure algebra operations,which has advantages in the computational efficiency of the model and the realization of semantic reasoning.

The semantic representation in this article will distinguish pronouns and predicates.Pronouns refer to ordinary nouns and their special forms,like person names, physical names, place names, etc. Chomsky (1965), which is characterized by the only static functions of referencing and naming, but not the abstract ability to describe processes and relationships, and predicates are complements of pronouns, such as verbs, adjectives, adverbs ..., which have descriptions or dynamic function to manipulate the pronouns or other predicates. Consider introducing a ring $(R, +, \times)$, we use $\mathcal{W}^n, \mathcal{W}^p$ to refer to the pronoun space and predicate space,then the word representation method employed in this article is as follows:

$$w_i \in \mathcal{W}^n \mapsto p_{i_1} p_{i_2} ... p_{i_d} w_j^k \in \mathcal{W}^p \mapsto f_j^k(x) = n_k(x - p_j) \tag{1}$$

where the pronoun w_i is mapped into timed primes $p_{i_1} p_{i_2} ... p_{i_d} \in R$, whose length (i_d) is an adjustable hyperparameter, which is a set of algebraic elements combined.It differs from vector representations only in vector space using continuous numerical values but abstract algebraic symbols with operational capabilities;And the predicate word w_j^k refers to the j-th word of the k-type predicates,which is mapped into a first-order polynomial formula $f_j^k(x) = n_k(x - p_j)$, where the coefficient $n_k \in R$ and the parameter $p_j \in R$ is a prime, and x is an varible is assignable, which can be any pronoun $w_i \in (W)^n$, or another polynomial $f_m^l(x) = n_l(x - p_m)$, so there can be an operable function between words, which is the same as the algebraic feature of words in Montague grammar Morrill (1994).The detailed design can be seen in Chapter **??**.

After representing a word as an algebraic representation on the ring R, the representation of the sentence can be calculated using certain rules. Firstly we represent a sentence $S_i = \{w_1 \cdots w_n\}$ as a syntactic dependency tree T_i Culotta & Sorensen (2004),which can describe the dependency relationship between words, that is, pointing out the syntactic relationship between words. According to the syntactic dependency tree T_i,we can define the connection relationship and arithmetic relationships between words. For example, consider a path $t = w_1, w_2, w_3 \in T_i$ in a syntactic dependency tree T_i and w_1 is the product of the pronouns mapped to prime factors $P_{w_1} = p_1 ... p_d$, then this path can be represented as a polynomial $f_t(P_{w_1}) = n_{w_2}(n_{w_2}(P_{w_1} - p_{w_2}) - p_{w_3})$. Therefore, each path of the syntactic dependency tree T_i corresponds to a polynomial f_t, and according to the needs of the semantic analysis task, the entire tree T_i can also be represented as a polynomial f_T;

We use the Stanford Natural Language Inference(SNLI) dataset to evaluate the effect of the model.The task can be described as classifying a sentence pair semantically, that is, $(S_1, S_2) \mapsto \{0, 1\}^3$ where (S_1, S_2) is a pair of sentences, and $\{0, 1\}^3$ is a vector that can refer to the semantic "contradiction","neutral","entailment" of the sentence(such as $(0, 1, 0)$ refers neutral). Consider the clause path $t_1 = \{w_i, \cdots\} \in T_1$ and $t_2 = \{w_i, \cdots\} \in T_2$, our method is based on the following basic idea: when there is a direct semantic connection between two sentences(whether it is contradiction or entailment), a sharing semantic part will exist between the clauses about the same pronoun w_i(that is, two sentences with semantic connections are different extensions of a sharing semantic), corresponding to the polynomial algebraic representation of the sentence above,the related clauses of the two sentences with semantic connections can be considered as polynomials f_1 and f_2 to the corresponding clauses, their common factor polynomials f' satisfy $f'|f_1$ and $f'|f_2$, in which f' can be considered to denote some kind of sentence semantic reasoning experience. After the processing of each example $(S_1, S_2) \in (D)$ in the training set by the model, an experience f' can be given, this common factor polynomial f' can be used to achieve semantic reasoning with excellent accuracy and generalization(see details in Chapter **??**).

2 Polynomial Semantic Representation

In this paper, we introduce one kind of algebraic structure,ring (R),to construct the semantic space corresponding to words.The ring is a basic concept of abstract algebra and its definition is as follows:

definition 2.1 ***(Ring)****In the non-empty set R, if two algebraic operations $+$ and $\times$ are defined, and satisfy:*

- *The set R constitutes the Abelian group for the operation $+$.*
- *The operation $\times$ satisfies the conjunction law under the collection R, ie for $\forall a, b, c \in R$,$(a \times b) \times c = a \times (b \times c)$(R forms a semigroup for $\times$).*
- *There is a distribution law for $+$ and $\times$, ie for $\forall a, b, c \in R$, there are:*

$$a \times (b + c) = a \times b + a \times c$$
$$(a + b) \times c = a \times c + b \times c$$

Before performing word representation, we firstly divide all words into pronouns and predicates.Pronouns refer to ordinary nouns and their special forms,like person names, physical names, place names, etc. Chomsky (1965), which is characterized by the only static functions of referencing and naming, but not the abstract ability to describe processes and relationships, and predicates are complements of pronouns, such as verbs, adjectives, adverbs ..., which have descriptions or dynamic function to manipulate the pronouns or other predicates. Consider introducing a ring $(R, +, \times)$, we use $\mathcal{W}^n, \mathcal{W}^p$ to refer to the pronoun space and predicate space,then the word representation method employed in this article is as follows:

$$w_i \in \mathcal{W}^n \mapsto p_{i_1} p_{i_2} ... p_{i_d} w_j^k \in \mathcal{W}^p \mapsto f_j^k(x) = n_k(x - p_j) \tag{2}$$

where the pronoun w_i is mapped into timed primes $p_{i_1} p_{i_2} ... p_{i_d} \in R$, whose length ($i_d$ is an adjustable hyperparameter, which is a set of algebraic elements combined.It differs from vector representations only in vector space using continuous numerical values but abstract algebraic symbols with operational

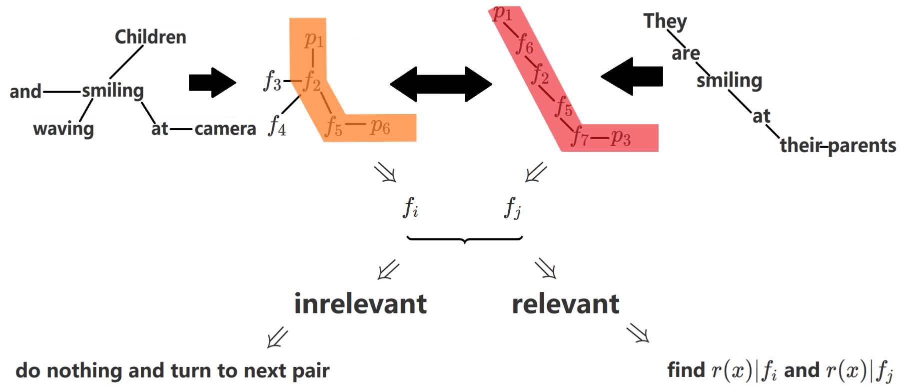

Figure 2: The processing of related parts between two sentences based on using polynomial semantic representation.

capabilities;And the predicate word w_j^k refers to the j-th word of the k-type predicates,which is mapped into a first-order polynomial formula $f_j^k(x) = n_k(x - p_j)$, where the coefficient $n_k \in R$ and the parameter $p_j \in R$ is a prime, and x is an varible is assignable, which can be any pronoun $w_i \in (W)^n$, or another polynomial $f_m^l(x) = n_l(x - p_m)$, so there can be an operable function between words, which is the same as the algebraic feature of words in Montague grammar Morrill (1994).Scince the predicate has the functions of modifying, describing, manipulating or connecting other pronouns or predicates, we can abstract the characteristics of the predicate into a mapping, whose input can be other predicates or pronouns.

3 Semantic Inference Learning Using Polynomial-Algebra Method

3.1 Task Describe

The SNLI corpus(version 1.0) Bowman et al. (2015) is a collection of 570k human-written English sentence pairs manually labeled for balanced classification with the labels entailment, contradiction, and neutral, supporting the task of natural language inference(NLI), also known as recognizing textual entailment(RTE).

Text	Judgments	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction	The man is sleeping.
An older and younger man smiling. at the cats playing on the floor.	neutral	Two men are smiling and laughing
A black race car starts up in front of a crowd of people.	contradiction	A man is driving down a lonely road.
A soccer game with multiple males playing.	entailment	Some men are playing a sport.

The task can be described as semantical inference for sentences pair, that is, $(S_1, S_2) \mapsto \{0,1\}^3$, where (S_1, S_2) is a pair of sentences, $\{0,1\}^3$ is a vector that can refer to the semantic of the sentence:"continuation", "neutral", or "entailment"(such as $(0,1,0)$ for neutral), so it can be regarded as a multi-classification problem. Except for the gold label, the dataset also contains the evaluation results of five human labelers(the subjective judgment of manual labeling),and each sentence is represented in two sentence-parsing methods.

3.2 Learning and Generalization Methods

Now consider a pair of sentences: (S_1, S_2), the corresponding syntax tree is (T_1, T_2), and the set of clauses in the syntax tree is $T_1 = \{t_1^{(1)}...t_c^{(1)}\}$ and $T_2 = \{t_1^{(2)}...t_d^{(2)}\}$, then each clause can also be represented as a set of words as $t_i^{(1)} = \{w_1^{(1)}...w_{l_i}^{(1)}\}$ and $t_j^{(1)} = \{w_1^{(2)}...w_(^{(2)}l_j)\}$, now suppose there is $t_i^{(1)} \cap t_j^{(2)} \neq$, then either the intersection contains at least one pronoun $w_n \in (W)^n$, or it contains two pronouns consisting of similar prime factors, then the two clauses may be related to the same pronoun description event, then these two clauses are defined as "Associative Clauses".If $t_i^{(1)} \cap t_j^{(2)} \neq$ contains pronouns that are irrelevant,then the two clauses can be considered as irrelevant.

The following considers the Association Clauses $(t_i^{(1)}, t_j^{(2)})$, after we use the above polynomial representation method to characterize it, the predicate polynomial can be used to represent the entire clause. Form a polynomial(such as $t_i =$ "man in clothe" and the "man" is the sharing pronoun, the sentence can be characterized as $t_i = \{p_{i_1}p_{i_2}p_{i_3}, n_1(x-p_7), p_{i_4}p_{i_5}p_{i_6}\}$,then which can be calculated as $t_i = n_1(x + p_{i_4}p_{i_5}p_{i_6} - p_7))$ so we can get $(t_i^{(1)}, t_j^{(2)}) \mapsto (f_i^{(1)}, f_j^{(2)})$, and it is easy to know that the polynomial can always be represented as the form $N_ix - P_i$ after calculation,so we end up with a formulation as $(t_i^{(1)}, t_j^{(2)}) \mapsto (N_ix - P_i, N_jx - P_j)$;

When the two sentences are irrelevant, the algorithm does not need to do anything.When the two sentences have an implication relationship, we can assume that a polynomial $f'(x) = N'x - P'$ is a common factor polynomial of $(N_ix - P_i, N_jx - P_j)$(which can be defined as "Associative Polynomials"), then we construct a table $\mathcal{T}^e$ to record such sequence pairs $(f_i^{(1)}, f^(2))_j, f')$(the two sentences in contradiction is $(f^(1))_i, -f^(2))_j, f')$),then the semantic inference experience are obtained and denoted by the factor polynomials. When it is necessary to infer whether a new set of association clauses $(t^(1))_k, t^(2))_l)$ is relevant, the model will do fuzzy search of algebraic operations in the table $(T)^e$ for matched similar contexts and associations, then inference of unknown semantics can be achieved.

3.3 Baseline Models and Expiriment Results

Models	Parameters Num	Train (% acc)	Test (% acc)
LSTM	220k	84.8	77.6
Self-Attention	7.0m	89.0	87.4
Poly-Repr	$\leq$8.0k	100.0	42.0

Baseline Models:we use 100D LSTM encoders Rocktäschel et al. and 2400D Multiple-Dynamic Self-Attention Model Rocktäschel et al. as the baseline models(both neural network models) to evaluate our works,one thing worthing attention is that not only the scoring results(train accuracy and test accuracy)matters but also the parameters number is also important.

Expiriment Results: The results show that the model proposed in this paper is better than the baseline models in terms of parameter capacity and accuracy on the training set, but the performance on the test set is far worse than the neural network models. The main reason may be related to the similarity measurement method of word meaning(the words that should be judged to be similar are judged to be irrelevant in the algorithm in this paper),the next step will focus on improving this area.

4 Conclusion

In this article, we replace the continuous representation method based on vector space with a pure algebraic representation method creatively, and neural network training and learning methods are not employed,finally the proposed model is superior to traditional neural network methods in model size and training difficulty.However, the generalization accuracy of the test set still needs to be improved, which will be the next improvement work.

Appendix

.1 Algorithm for Transforming Sentences into Syntax Trees

Our goal is to transform a string that has been manually labeled in the form like "((A woman) ((is ((doing (a (cartwheel while))) (wearing ((a bikini) (in ((the sand) (next (to (the beach))))))))) .))" into a data structure of a tree in a computer. The algorithm is as follows:

Algorithm 1 The Transform Algorithm **Find**(s,l,p_T):From Language String to Syntactic Dependency Tree

Input: Labeled Nueral Language String s;l =lenght of s;Position of Root of the Tree p_T;
Output: Syntactic Dependency Tree T;

Convert s to an Array s splited by Space ' ';
T.root = $s[p_T]$;
Let N_s = 0 to Count the Number of Brackets.
for $n = p_T + 1$ to l **do**
  IF $s[n] ==' ('$:$N_s = N_s + 1$;
  IF $s[n] ==')'$:$N_s = N_s - 1$;
  // Find the First Sub-Node;
  IF $s[n]! =' ('$ or $')'$:T.root.subnodes.append($s[n]$);
  // Find the Leaving Sub-Nodes;
  Let N'_s = N_s to Help Finding the Next Sub-Node.
  for $m = n$ to l do **do**
    IF $s[n] ==' ('$:$N'_s = N_s + 1$;
    IF $s[n] ==')'$:$N'_s = N_s - 1$;
    IF $N'_s == N_s$:T.root.subnodes.append($s[m]$);
  end for
end for
for $N \in T$.root.subnodes **do**
  T = **Find**(s,l,p_N);
end for
Return T;

Algorithm 2 The Reasoning Algorithm **R**($f_k^{(1)}, f_l^{(2)}$)

Input: Sentence Pairs with Unknown Relationship $(f_k^{(1)}, f_l^{(2)})$**;Sequence Pair Set** $\mathcal{E}$**; Output: Relevant Representation** r**;**

IF $\exists(f_k^{(1)}, f_l^{(2)}) == (f_k^{(1)}, f_l^{(2)})$:**Return** r based on f';
IF $\exists f' ==$ gcd$(f_k^{(1)}, f_l^{(2)})$ and $f' \in \mathcal{E}$: **Return** r based on f';
IF $\exists a|f_k^{(1)}, b|f_l^{(2)}$ and $f ==$ gcd(a, b) and $f \in \mathcal{E}$: **Return** r based on f';
IF $\exists n ==$ gcd$(n_k^{(1)}, n_l^{(2)})$ and $p ==$ gcd$(p_k^{(1)}, p_l^{(2)})$ and $f \in \mathcal{E}$: **Return** r based on $f = nx - p$;
Else :**Return** r = 'neutral';

.2 Semantic Reasoning Algorithm for Polynomial Representation Model

The goal of the algorithm is to infer whether the sentence pair $(f_k^{(1)}, f_l^{(2)})$ with unknown relationship is related based on the existing sequence pair set $\mathcal{E} = \{(f_i^{(1)}, f_j^{(2)}, f')\}$(experience set) abstracted from the training set. The algorithm is as follows:

References

Binder and Jeffrey R. In defense of abstract conceptual representations. *Psychonomic Bulletin and Review*, 23(4):1096–1108.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2015.

Noam Chomsky. *Aspects of the Theory of Syntax*. 1965.

V J Cook, Newson Mark, and Fletcher Janet. *Chomsky's Universal Grammar*. 1988.

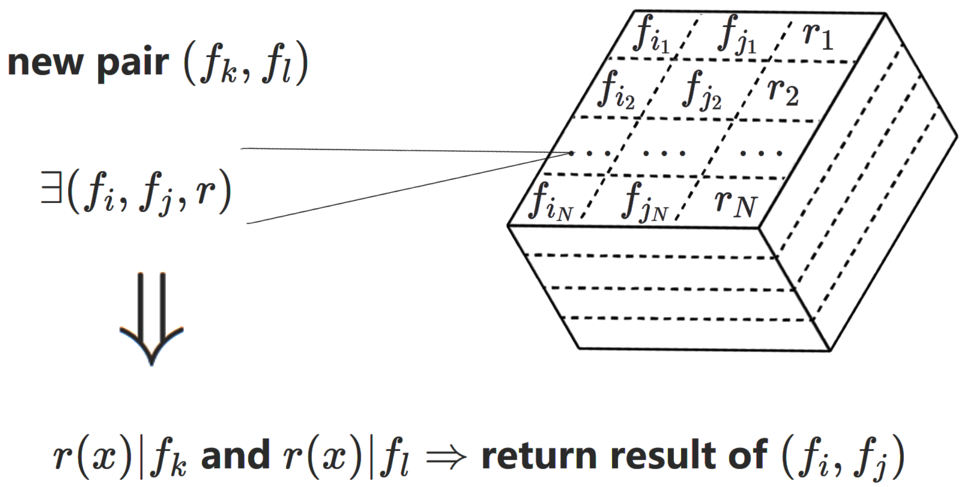

Figure 3: Principle of Sentence Semantic Reasoning Based on Polynomial Semantic Representation.

Aron Culotta and Jeffrey S. Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, 21-26 July, 2004, Barcelona, Spain.*, 2004.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *NAACL-HLT*, 2018.

Seonhoon Kim, Jin-Hyuk Hong, Inho Kang, and Nojun Kwak. Semantic sentence matching with densely-connected recurrent and co-attentive information. In *AAAI*, 2018.

Igor Mordatch. Concept learning with energy-based models. 2018.

Glyn V. Morrill. *Montague Grammar*. Springer Netherlands, 1994.

Boyuan Pan, Yazheng Yang, Zhou Zhao, Yueting Zhuang, Deng Cai, and Xiaofei He. Discourse marker augmented network with reinforcement learning for natural language inference. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 989–999, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1091. URL `https://www.aclweb.org/anthology/P18-1091`.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiský, and Phil Blunsom. Reasoning about entailment with neural attention.

Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Failures of gradient-based deep learning. 2017.

Jean Charles Vialatte and François Leduc-Primeau. A study of deep learning robustness against computation failures. 2018.

Haonan Yu, Xiaochen Lian, Haichao Zhang, and Xu Wei. Guided feature transformation (gft): A neural language grounding module for embodied agents. 2018.

Kun Zhang, Guangyi Lv, Linyuan Wang, Le Wu, Enhong Chen, Fangzhao Wu, and Xing Xie. Drr-net: Dynamic re-read network for sentence semantic matching. In *AAAI*, 2019.