

## Article

# A Deep Fusion Matching Network Semantic Reasoning Model

Wenfeng Zheng <sup>1</sup> , Yu Zhou <sup>1</sup>, Shan Liu <sup>1</sup> , Jiawei Tian <sup>1</sup> , Bo Yang <sup>1</sup> and Lirong Yin <sup>2,\*</sup> 

<sup>1</sup> School of Automation, University of Electronic Science and Technology of China, Chengdu 610054, China; winfirms@uestc.edu.cn (W.Z.); zhouyu\_cq@126.com (Y.Z.); shanliu@uestc.edu.cn (S.L.); jraavis.tian23@gmail.com (J.T.); boyang@uestc.edu.cn (B.Y.)

<sup>2</sup> Department of Geography and Anthropology, Louisiana State University, Baton Rouge, LA 70803, USA

\* Correspondence: yin.lyra@gmail.com

**Abstract:** As the vital technology of natural language understanding, sentence representation reasoning technology mainly focuses on sentence representation methods and reasoning models. Although the performance has been improved, there are still some problems, such as incomplete sentence semantic expression, lack of depth of reasoning model, and lack of interpretability of the reasoning process. Given the reasoning model's lack of reasoning depth and interpretability, a deep fusion matching network is designed in this paper, which mainly includes a coding layer, matching layer, dependency convolution layer, information aggregation layer, and inference prediction layer. Based on a deep matching network, the matching layer is improved. Furthermore, the heuristic matching algorithm replaces the bidirectional long-short memory neural network to simplify the interactive fusion. As a result, it improves the reasoning depth and reduces the complexity of the model; the dependency convolution layer uses the tree-type convolution network to extract the sentence structure information along with the sentence dependency tree structure, which improves the interpretability of the reasoning process. Finally, the performance of the model is verified on several datasets. The results show that the reasoning effect of the model is better than that of the shallow reasoning model, and the accuracy rate on the SNLI test set reaches 89.0%. At the same time, the semantic correlation analysis results show that the dependency convolution layer is beneficial in improving the interpretability of the reasoning process.

**Keywords:** sentence representation; semantic reasoning; attention mechanism; long-short memory network; deep fusion matching network



**Citation:** Zheng, W.; Zhou, Y.; Liu, S.; Tian, J.; Yang, B.; Yin, L. A Deep Fusion Matching Network Semantic Reasoning Model. *Appl. Sci.* **2022**, *12*, 3416. <https://doi.org/10.3390/app12073416>

Academic Editor: Luis Javier Garcia Villalba

Received: 8 February 2022

Accepted: 25 March 2022

Published: 27 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Natural language inference (NLI) is a process in which the abstract representation of natural language text pairs becomes space vectors. The reasoning learns the potential relationship between text pairs. NLI has become one of the most critical benchmark tasks in natural language understanding because of its complex language understanding and in-depth information involved in reasoning. At present, NLI technology mainly consists of three parts: encoding, sentence representation (understanding), and reasoning learning. Among them, sentence reasoning learning is still a long way from the goal of practical application—the construction and optimization of the reasoning model. The research on deep learning for the NLI model is still in its infancy. Although the existing deep learning algorithm models, such as cyclic neural network and convolutional neural network, have achieved initial results in the construction of the reasoning model, they have failed to achieve a breakthrough. Therefore, there is a broad space for research on the construction and optimization of the NLI model. The construction and optimization of sentence semantic representation and reasoning models have become two core problems in NLI. No matter which aspect is improved, the effect of the whole NLI method will be affected. At the same time, it is of great significance to study the influence of the two methods on the NLI method.

In existing studies, to focus on the construction and optimization of a semantic representation and inference model, input is usually simplified as sentence pairs to avoid the interference caused by miscellaneous data. Therefore, NLI technology is also known as sentence representation reasoning technology. Before the appearance of sentence-level representation technology, the method of sentence-level representation was to use CBOW embedded distributed representation technology based on word encoding to represent text as a fixed-length sentence vector. However, with the development of neural networks and deep learning, recently, sentence representation technology has gradually developed from a combination of simple word-embedded models to more complex architecture, such as the convolutional network [1], cyclic neural network [2,3], and its deformation [4]. They have been applied to improve the performance of sentence representation. Inspired by these works, this paper decided to use a tree convolutional network for the extraction of sentence structural information.

Besides sentence representation, semantic reasoning is a process that infers the logical relationship of text pairs by analyzing the internal relationship between text information and text according to a given natural language text pair. NLI mainly adopted the method based on logical formal reasoning [5,6], which transformed sentences expressed in natural language form into logical expressions that computers can understand, and then it realized semantic reasoning using a logic interpreter. Moldovan [7] proposed a logic-based reasoning method COGEX, which is based on logic, to represent the relationship between inferential text pairs, such as syntactic objects, syntactic subjects, and causal relationships. In addition to the logical representation of input text pairs, the method also uses knowledge base content with logical representation. Raina [8] proposed a dependency syntactic logic reasoning method, which parses the syntactic relationship of the text, constructs a syntax tree, and then completes semantic reasoning through the relationship between the parent node and the child node. Logic-based reasoning technology has achieved good results in processing small-scale data. However, with the increasing data volume and the sentence structure complexity, the applicability and accuracy of the model are limited [9]. With the development of deep learning in natural language processing, semantic reasoning technology gradually changes from logic-based reasoning technology to deep learning-based reasoning technology [10]. The core of reasoning technology based on deep learning is to calculate the similarity of two semantic objects and simulate the potential correspondence between different abstract levels and different properties of “semantic objects” [11].

In order to solve the problems of the lack of reasoning depth and interpretability in the reasoning model, this paper designs a deep fusion matching network, which mainly includes a coding layer, matching layer, dependence convolution layer, information aggregation layer, and inference prediction layer. We first improve the matching layer based on the deep matching network and use a heuristic matching algorithm to replace the complex neural network as the interactive fusion mode of matching information. Secondly, the dependency convolutional layer uses a tree convolutional network (TBCNN) to extract the structural information of sentences. Finally, we analyze prediction accuracy, semantic correlation analysis, and ablation analysis of the model’s performance on multiple datasets.

## 2. Materials

### 2.1. SNLI Dataset

The SNLI dataset is a text implication recognition dataset published by Stanford University. SNLI is manually annotated and contains 570 k text pairs. There are three kinds of marks: implication, contradiction, and neutral. In this paper, all data are divided into a training set (549,367 samples), a verification set (9842 samples), and a test set (9824 samples), according to Zhu’s [12] data partition rules, and some SNLI data forms are shown in Table 1.

**Table 1.** Sample data of SNLI dataset.

Premise Sentence	The Label	Hypothetical Sentence
Two women are embracing while holding to-go packages.	Entailment E E E E E	Two woman are holding packages.
A man selling donuts to a customer during a world exhibition event held in the city of Los Angeles.	Contradiction C C C C C	A woman drinks her coffee in a small café.
A man in a blue shirt standing in front of a garage-like structure painted with geometric designs.	Neutral N E N N N	A man is repainting a garage.

## 2.2. Multi-NLI Dataset

The Multi-NLI dataset published by Adina Williams, Nikita Nangia, and Sam Bowman [13] contains 433 k text pairs. Different from the SNLI dataset, it covers more data close to real life, such as novels and telephone voice. The sample data are shown in Table 2. The dataset contains 10 categories of data. Whether the same category appears in the training and test sets simultaneously, it is divided into the matched and unmatched sets.

**Table 2.** Sample data of Multi-NLI dataset.

Type	Premise Sentence	The Label	Hypothetical Sentence
Novel	The Old One always comforted Ca'daan, except today.	neutral	Ca'daan knew the Old One very well.
Message	Your gift is appreciated by each and every student who will benefit from your generosity.	neutral	Hundreds of students will benefit from your generosity.
Cell	yes now you know if everybody like in August when everybody's on vacation or something we can dress a little more casual or	contradiction	August is a black out month for vacations in the company.

In this paper, the text implication task is performed on the unmatched and matching sets. The data are divided: training set (392,702 samples), matching/unmatching verification set (9815/9832 samples). Since the test set data cannot be obtained, this paper uses a verification set instead of a test set.

## 3. Methods

The semantic reasoning model based on matching [14–18] comprises the coding, matching, and prediction layers. The detailed explanation of the sentence representation and semantic reasoning section are presented in the following section. The information extraction method includes the matching model and the syntactic structure extraction model. The semantic reasoning section based on deep fusion matching network includes sentence coding layer, local reasoning, syntactic structure model, global reasoning, and resulting reasoning and prediction.

### 3.1. Reasoning Information Extraction Method

#### 3.1.1. Matching Model Based on AF-DMN

Inspired by the deep neural framework [19], Duan [20] proposed an attention-fused deep matching network, referred to as AF-DMN, based on the matching reasoning model. However, AF-DMN has a more complex matching layer. The matching layer is composed of T identical calculation blocks. Each calculation block contains four sub-modules: (1) cross attention layer; (2) cross attention fusion layer; (3) self-focus layer; (4) self-focus fusion layer.

### 3.1.2. Syntactic Structure Extraction Based on Tree Convolution Network

In order to capture the syntactic structure of a sentence, Mou [21] proposed a tree-based convolutional neural network (TBCNN). It can effectively capture syntactic information compared with a conventional convolutional neural network.

First, sentences are converted into parse trees. Then, the structure information of sentences is extracted along with the tree structure by a sliding window. Finally, the syntactic information is captured through the hidden and output layers. Thus, TBCNN contains syntactic parsing and convolution layers, divided into dependency convolution networks (d-TBCNN) and component convolution networks (c-TBCNN).

### 3.2. Design of Reasoning Model Based on Deep Fusion Matching Network

This paper proposes a semantic fusion deep matching network, referred to as SCF-DMN. The core of the model is as follows: using the improved AF-DMN model to obtain the local inference information between sentences and help to obtain the deep reasoning information; using d-TBCNN to simulate the syntactic structure information of the sentence to improve the interpretability of the reasoning process; finally, the idea of the control gate is used to fuse the local inference information and syntactic structure information of sentences to form the global reasoning information of the reasoning model, thus expanding the reasoning depth and interpretability of the model.

As shown in Figure 1, the whole matching network consists of five parts: coding layer, matching layer, dependency convolution layer, information aggregation layer, and inference prediction layer. The specific functions of each part are as follows:

- (1) Coding layer: it mainly completes the transformation from natural language representation to sentence embedding representation, including sentence preprocessing, vectorization, semantic information coding, and embedded representation generation.
- (2) Matching layer and dependency convolution layer: they mainly complete the extraction of local inference information between sentences and syntactic structure inference information. Moreover, by extracting the interactive information between sentences, implicit logic is introduced into the reasoning process to improve the interpretability of the reasoning process.
- (3) Information aggregation layer: it mainly completes the integration of representation information, interactive reasoning information, and syntactic structure reasoning information. All information is integrated into fixed-length semantic information using cyclic neural networks and pooling in deep learning.
- (4) Reasoning and prediction layer: it mainly completes the output of prediction results of specific reasoning tasks. In general, linear function and multi-layer fully connected network are used to infer the global reasoning information after fusion to predict the implication relationship of a given sentence pair. The detailed structure and function of the sub-networks are given below.

The detailed explanation and design of each part are given below.

#### 3.2.1. Sentence Coding

In order to avoid the result's interference of the sentence representation on the judgment of the reasoning model, the coding layer of the deep fusion matching network is designed in this paper. The bidirectional long-short memory network is used to obtain the sentence embedded representation of the premise sentence pair  $(p, q)$ . If there is no clear indication in the following text, the premise sentence is represented by substitution, and use  $p$  instead of presupposition, use  $q$  instead of hypothetical sentences.

Firstly, this paper preprocesses the  $p$  and  $q$  sentences, including English word segmentation and the removing of stop words, to obtain the word list  $p = (p_1, p_i, \dots, p_m)$  and  $q = (q_1, q_j, \dots, q_n)$ , where  $m$  and  $n$  represent the number of words in the sentence and respectively.

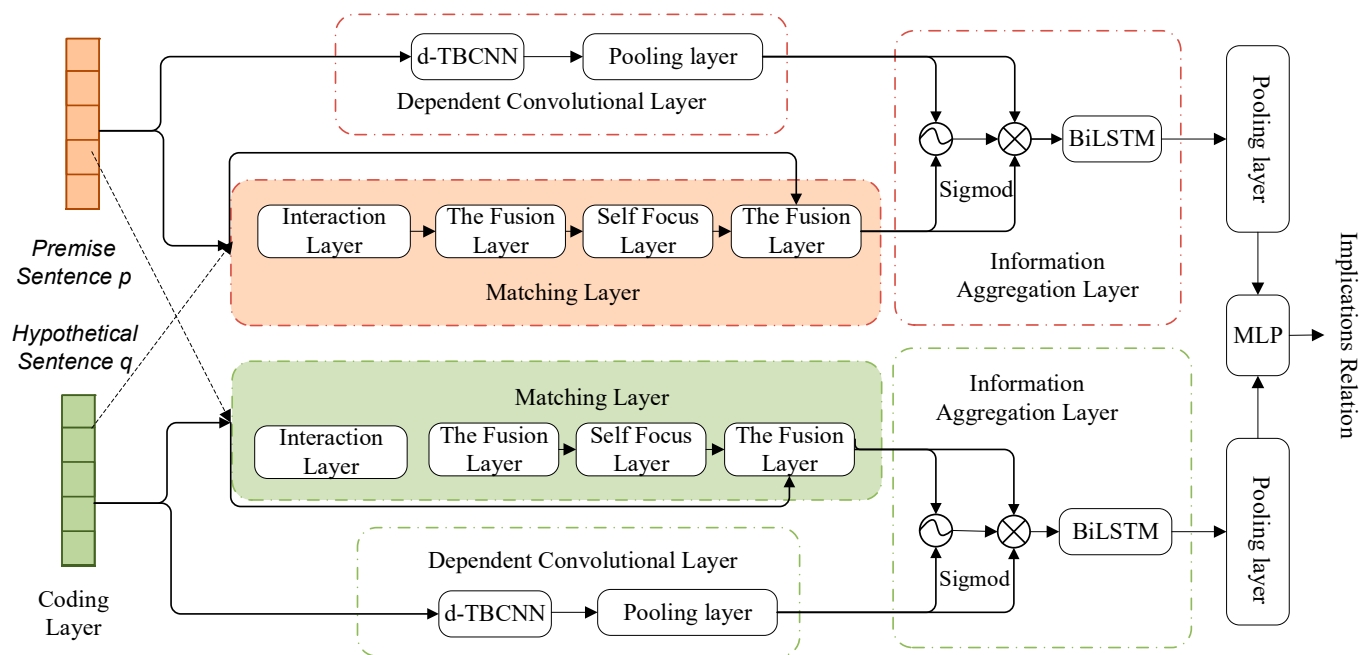


Figure 1. SCF-DMN network structure.

Then, the bidirectional long-short memory network combined with sentence context information is used to encode sentence semantic-information, and the hidden layer state  $h_i$  of the  $i$ th word in the sentence is obtained as shown in Formula (1).

$$h_i = BiLSTM(e_i, h_{i-1}, h_{i+1}) \quad (1)$$

Among them,  $e_i$  represents the  $n_e$  dimension word vector corresponding to the  $i$ -th word, which is generated by word2vec technology;  $h_{i-1}$  and  $h_{i+1}$  represent the hidden layer state corresponding to the previous and the next word of the  $i$ th word, respectively.

Combining the hidden layer state of each word in the sentence, we obtain the sentence embedding representation  $H$ , as shown in Formula (2):

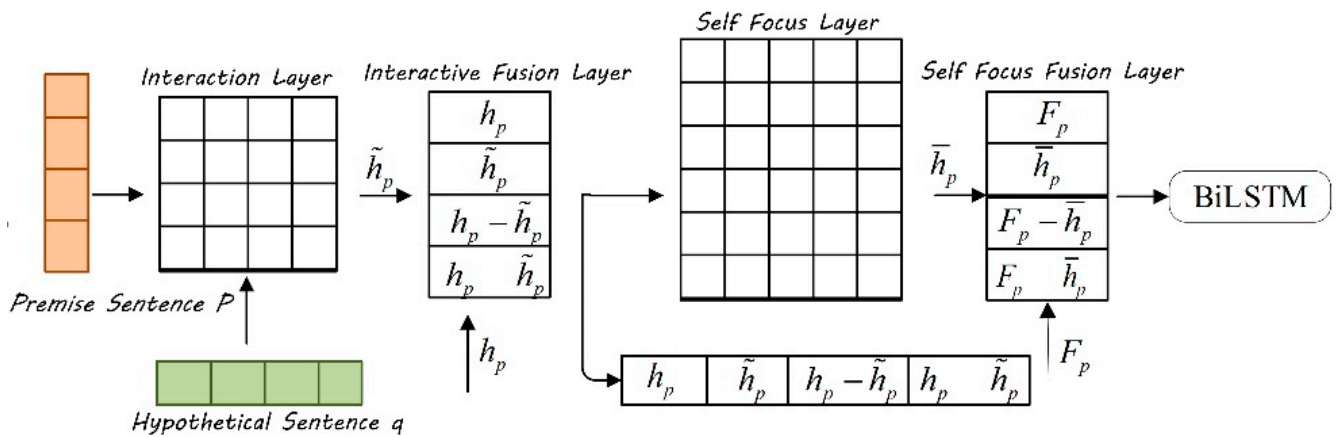
$$H = (h_1, h_i, \dots, h_L) \quad (2)$$

where  $H \in R^{1 \times L}$ ,  $L$  is the length of the sentence. After the final sentence  $p$  and  $q$  pass through the coding layer, the sentences embedded are expressed as  $H_p = (h_{p_1}, h_{p_i}, \dots, h_{p_m})$  and  $H_q = (h_{q_1}, h_{q_i}, \dots, h_{q_n})$ .

### 3.2.2. Local Reasoning Based on Improved AF-DMN

The matching layer of the deep fusion matching network model refers to the chain structure of the AF-DMN model. It then passes through  $T$  identical matching modules to collect the local interactive reasoning information based on the sequence. The reasoning information specifically includes the internal context information of sentences  $p$  and  $q$ , and the interaction information between sentences  $p$  and  $q$ .

As shown in Figure 2, each matching module is divided into four sub-layers: interaction layer, interaction fusion layer, self-focus layer, and self-focus fusion layer. The interaction layer obtains the interactive information between sentences  $p$  and  $q$ . The interaction fusion layer enhances the extraction process of interactive information. The self-focus layer obtains the context information within the sentence to solve the long-term dependence problem. Finally, the self-focus fusion layer enhances the effect of content extraction.



**Figure 2.** The network structure of the matching layer.

Before obtaining the interactive information between the premise statement and the hypothetical statement, it is necessary to obtain the alignment information of the relevant sub-components between the sentences, namely, the interactive attention matrix. Alignment information acquisition methods are divided into hard alignment and soft alignment. Hard alignment [5] requires one-to-one correspondence between words, while soft alignment [22] is closer to semantic information alignment. Words or phrases with consistent semantics have a higher weight on the attention matrix. For example, “near” is aligned with “be close to”. Therefore, the interaction layer of the semantic fusion depth matching network uses the soft alignment proposed by Chen [23] and calculates the inner product between sentences  $p$  and  $q$  to obtain the correlation between sentences.

Firstly, the correlation sub-component weight  $e^t$  between sentences in the  $i$ th matching module is calculated, and  $e_{ij}^t$  represents the correlation between the  $i$ th word in sentence  $p$  and the  $i$ th word in sentence  $q$ . The calculation method is shown in formula (3).

$$e_{ij}^t = f(W^t[p_i, q_j] + b) = h_{p_i}^{t-1} W^t h_{q_j}^{t-1} + \langle U_l^t, h_{p_i}^{t-1} \rangle + \langle U_r^t, h_{q_j}^{t-1} \rangle. \quad (3)$$

where  $W^t \in R^{2h \times 2h}$ ,  $U_l^t \in R^{2h}$ ,  $U_r^t \in R^{2h}$  represents the parameter of the  $t$ th matching module and represents the point multiplication operation.

Then, the weight of relevant sub-components  $e_{ij}^t$  is replaced into Formulas (4) and (5) to calculate the correlation matrix  $a_{p_i}^t$  of the premise sentence  $q$  on the hypothetical sentence  $q$  and the correlation matrix  $a_{q_j}^t$  of the hypothetical sentence  $p$  on the premise sentence.

$$a_{p_i}^t = \sum_{j=1}^n \frac{\exp(e_{ij}^t)}{\sum_{k=1}^n \exp(e_{ik}^t)}, \forall i \in [1, \dots, m] \quad (4)$$

$$a_{q_j}^t = \sum_{i=1}^m \frac{\exp(e_{ij}^t)}{\sum_{k=1}^m \exp(e_{kj}^t)}, \forall j \in [1, \dots, n] \quad (5)$$

where  $m$  and  $n$  denote the participles number of sentences  $p$  and  $q$  respectively, and  $\exp(\cdot)$  denotes the exponential function with natural constant  $e$  as the base.

The interaction information  $\tilde{h}_{p_i}$  between the  $i$ th word in the sentence  $p$  and the sentence,  $q$  is obtained by solving the correlation matrix  $a_{p_i}^t$  and the previous matching module  $H_{p_i}^{t-1}$ , as shown in Formula (6).

$$\tilde{h}_{p_i}^t = H_{p_i}^{t-1} \cdot a_{p_i}^t \quad (6)$$

where  $H_p^{t-1} = (h_{p_1}^{t-1}, h_{p_2}^{t-1}, \dots, h_{p_m}^{t-1})$  is the sentence embedding representation of the  $t$ th matching module connected to the  $t - 1$ th matching module. Similarly, the interactive



information  $\tilde{h}_{q_j}$  between the  $j$ -th word in the sentence  $q$ , and the sentence  $p$  can be obtained, as shown in Formula (7).

$$\tilde{h}_{q_j}^t = H_q^{t-1} \cdot a_{q_j}^t \quad (7)$$

where,

$$H_q^{t-1} = (h_{q_1}^{t-1}, h_{q_j}^{t-1}, \dots, h_{q_n}^{t-1}).$$

In order to further enhance the interaction between sentences  $p$  and  $q$ , SCF-DMN sets a fusion layer after the interaction layer. Because the interactive information between sentences does not depend on the previous state of a single sentence, a bidirectional long-short memory network cannot significantly improve the correlation between sentences. However, it will cause an unnecessary calculation process for the reasoning model. Therefore, the interaction fusion layer of SCF-DMN only uses the heuristic matching method to fuse the interactive information  $\tilde{h}_{p_i}^t$  and  $\tilde{h}_{q_j}^t$  of sentences  $p$  and  $q$ .

The calculation formula of the cross fusion representation  $F_{p_i}^t$  and  $F_{q_j}^t$  of sentence  $p$  and  $q$  in the  $t$ -th matching module are shown in Formulas (8) and (9).

$$F_{p_i}^t = [h_{p_i}^t; \tilde{h}_{p_i}^t; h_{p_i}^t - \tilde{h}_{p_i}^t; h_{p_i}^t \odot \tilde{h}_{p_i}^t] \quad (8)$$

$$F_{q_j}^t = [h_{q_j}^t; \tilde{h}_{q_j}^t; h_{q_j}^t - \tilde{h}_{q_j}^t; h_{q_j}^t \odot \tilde{h}_{q_j}^t] \quad (9)$$

where  $h_{p_i}^t - \tilde{h}_{p_i}^t$  is the similarity difference between the word representation  $h_{p_i}^t$  of the  $i$ -th word of sentence  $p$  in the  $t$ -th matching module and the corresponding interactive information  $\tilde{h}_{p_i}^t$ . In the same way,  $h_{q_j}^t - \tilde{h}_{q_j}^t$  represents the similarity difference of the word representation  $h_{q_j}^t$  of the  $j$ -th word of sentence  $q$  in the  $t$ -th matching module and corresponding interactive information  $\tilde{h}_{q_j}^t$ .  $\odot$  represents point multiplication operation,  $[\dots; \dots; \dots]$  represents splicing operation.

The self-focus mechanism is introduced into the self-focus layer to solve the long-term dependence problem in the reasoning process. Long-term dependence means that the current system's state may have been affected by the system's state a long time ago, especially for long sentences (sentence length is greater than or equal to 17).

For the premise sentence  $p$ , firstly, based on the cross fusion representation  $F_{p_i}^t$  obtained from the interaction fusion layer, the internal correlation degree  $s_{ij}^t$  of each word in the sentence is calculated as follows:

$$s_{ij}^t = \langle F_{p_i}^t, F_{p_j}^t \rangle, \forall i, j \in [1, 2, \dots, m] \quad (10)$$

where  $F_{p_i}^t$  and  $F_{p_j}^t$  represent the cross fusion representation of the  $i$ -th word and the  $j$ -th word of the sentence  $p$  in the first matching module, and  $m$  is the number of participles in the sentence  $p$ , and  $\langle \cdot \rangle$  represents the Euclidean distance solution.

Then, we calculate the self-focus matrix  $S_{p_i}^t$  of the word using the Formula (11).

$$S_{p_i}^t = \sum_{i=1}^m \frac{\exp(s_{ij}^t)}{\sum_{k=1}^m \exp(s_{kj}^t)} \quad (11)$$

Finally, the self-focus vector  $h_{p_i}^t$  of the  $i$ -th word in the sentence,  $p$ , is obtained by multiplying the self-focus matrix with the cross-fusion representation, as shown in Formula (12).

$$h_{p_i}^t = F_{p_i}^t \cdot S_{p_i}^t \quad (12)$$

Similarly, we can obtain the self-focus vector  $h_{q_j}^t$  of the  $j$ -th word in the sentence  $q$ , as shown in Formula (13).

$$h_{q_j}^t = F_{q_j}^t \cdot S_{q_j}^t \quad (13)$$

In this layer, in addition to using the heuristic matching method to model the high-order information between words in a sentence, a bidirectional long-short memory network is used to strengthen the internal information dependence of the  $h_{p_i}^t$  and  $h_{q_j}^t$  of self-focus vectors.

For the premise sentence  $p$ , the heuristic matching method is used to obtain the self-focus fusion information  $\hat{h}_{p_i}^t$  of the word.

$$\hat{h}_{p_i}^t = [F_{p_i}^t; h_{p_i}^t; F_{p_i}^t - h_{p_i}^t; F_{p_i}^t \odot h_{p_i}^t] \quad (14)$$

Then, the fusion information of self-focus  $\hat{h}_{p_i}^t$  is input into bidirectional long-term and short-term memory network after activation function to further enhance the capture of internal information dependency, and the top-level hidden layer state is taken as the enhanced local interactive reasoning information  $\hat{h}_{p_i}^t$ . The whole calculation process is shown in Formulas (15) and (16).

$$h'_{p_i} = \sigma(W_h^t \hat{h}_{p_i}^t + b_h^t) \quad (15)$$

$$h_{p_i}^t = BiLSTM(h'_{p_i}, h_{p_{i-1}}^t, h_{p_{i+1}}^t) \quad (16)$$

where  $\sigma(\cdot)$  represents the activation function,  $W_h^t$  and  $b_h^t$  represent the parameters of the activation function;  $h_{p_{i-1}}^t$  and  $h_{p_{i+1}}^t$  represent the context of the first word in the sentence.

Then, the interactive inference information  $H_p^t$  of the premise sentence  $p$  and the local interactive inference information  $H_q^t$  of the sentence  $q$  obtained by the  $t$ -th matching module in the matching layer are shown in Formula (17) and Formula (18).

$$H_p^t = (h_{p_1}^t, h_{p_i}^t, \dots, h_{p_m}^t) \quad (17)$$

$$H_q^t = (h_{q_1}^t, h_{q_j}^t, \dots, h_{q_n}^t) \quad (18)$$

The matching layer of SCF-DMN adopts a circular chain network, and the whole matching layer consists of  $t$ -th identical matching modules. The above calculation process is repeated in turn. Finally, the output of the  $t$ -th matching module is used as the local interactive inference information  $v_q$  and  $v_p$ .

### 3.2.3. Syntactic Structure Modeling Based on d-TBCNN

The semantic fusion depth matching network designed in this paper uses a dependency tree convolution network (d-TBCNN) to collect sentence syntactic structure reasoning information and improve inference information.

As shown in Figure 3, the dependency convolution network will perform a convolution operation on each subtree according to the result of the dependency analysis tree of the sentence, extract the syntactic structure features of the subtree, and then splice all the syntactic structure features to form the syntactic structure inference information of the sentence. The specific calculation steps are as follows:

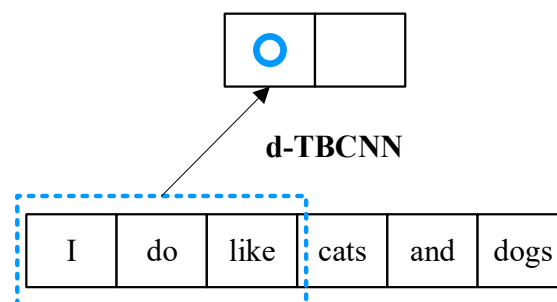


Figure 3. Schematic diagram of dependency convolution network.



- (1). Firstly, the natural language parser [24] proposed by Stanford University is used to transform sentences into a dependency syntax tree. Taking the premise sentence as an example, each node in the syntax tree corresponds to a word in the sentence. The arc between nodes indicates that the node (child node) and node (parent node) have a dependency relationship, and the arc is marked with the syntax relationship between the two nodes [25]. Because there are too many dependencies between words, some are meaningless for inferring sentence structure information. Therefore, referring to the work of Mou [26], the dependency convolution layer only retains 34 grammatical relationships that are frequently used and are more important. Some of the dependencies are shown in Table 3.

**Table 3.** Partial dependency tags and comments.

Sign	Relationship Type
SBV	Subject–verb
VOB	Verb–object
IOB	Indirect–object
FOB	Fronting–object

- (2). (Then, the syntactic structure features corresponding to the subtree are extracted along with the dependency subtree. The feature extractor adopts a double-layer convolution layer [27]. Suppose that the child nodes connected to the parent node  $n_{p_d}$  are  $n_{c_i} (i = 1, 2, \dots, m_c)$ , in which  $m_c$  represents the total number of child nodes. For each subtree  $y_c$ , the extracted local sentence structure features are as follows:

$$y_{p_i}^c = f(W_{p_d}^d \cdot p_d + \sum_{j=1}^n W_{r[c_j]}^d \cdot c_j + b^d) \quad (19)$$

Among them, the structural feature  $y_c \in R^{m_c}$ ,  $p_d$  is the word embedding vector corresponding to the parent node, and the word embedding vector corresponding to the  $j$ -th child node is  $c_j$ . The word vector representations in the dependency tree are obtained by pretraining in the coding layer.  $W_p^d \in R^{m_c \times n_e}$  is the weight corresponding to the node, the weight assigned according to the dependency type between words is  $W_{r[c_i]}^d \in R^{m_c \times n_e}$ .  $b^d \in R^{m_c}$  is the offset vector, in which  $r[c_j]$  represents the dependency relationship between nodes  $p$  and  $c_j$ .

- (3). By pooling the structural features of each subtree in the sentence  $p$ , the syntactic structure features of the sentence  $p$  are shown in Formula (20), and the syntactic structure features  $u_q$  of the sentence  $q$  are shown in Formula (21).

$$u_p = (y_{p_1}^c, y_{p_i}^c, \dots, y_{p_m}^c) \quad (20)$$

$$u_q = (y_{q_1}^c, y_{q_j}^c, \dots, y_{q_n}^c) \quad (21)$$

### 3.2.4. Global Reasoning Information

The purpose of the information aggregation layer is to combine the matching semantics  $v_p$  and  $v_q$  extracted from the matching layer, and the syntactic structure features  $u_p$  and  $u_q$  are extracted from the tree convolution layer to construct the input of the final inference prediction layer.

For the premise sentence  $p$ , firstly, the local interactive inference information  $v_p$  of the sentence is connected with the corresponding syntactic structure features  $u_p$ . The fusion proportion of each part is determined by adding a control gate. For the  $i$ -th word in the

premise sentence  $p$ , the calculation process of the fusion reasoning information is shown in Formula (22).

$$\begin{cases} x'_{p_i} = [v_{p_i}; u_{p_i}] \\ g_{p_i} = \text{sigmoid}(W_p^g x'_{p_i} + b_p^g) \\ x_{p_i} = g_{p_i} \odot x'_{p_i} \end{cases} \quad (22)$$

where  $g_{p_i}$  is the control gate,  $[;]$  represents the splicing operation,  $\odot$  represents the point multiplication operation, and  $W_p^g$  represents the training parameters.

Then, the global inference information  $V_p = (h_{p_1}, h_{p_i}, \dots, h_{p_m})$  of sentence  $p$  is generated by the bidirectional long-short memory network. The calculation formula of the global inference information of each word is shown in Formula (23).

$$\begin{cases} \vec{h}_{p_i} = \text{BiLSTM}(x_{p_i}, \vec{h}_{p_{i-1}}) \\ \overleftarrow{h}_{p_i} = \text{BiLSTM}(x_{p_i}, \overleftarrow{h}_{p_{i+1}}) \\ h_{p_i} = [\vec{h}_{p_i}, \overleftarrow{h}_{p_i}] \end{cases} \quad (23)$$

Similarly, the global inference information  $V_q = (h_{q_1}, h_{q_j}, \dots, h_{q_n})$  of the sentence  $q$  can be obtained.

### 3.2.5. Result Reasoning and Prediction

The length of the global reasoning information generated by the information aggregation layer is consistent with the original length of the sentence, which may lead to the sentence pair being unable to achieve reasoning because of the inconsistent information length. Therefore, in order to unify the sentence dimension without changing the inference information, the inference prediction layer of SCF-DMN designed in this paper uses a pooling operation to convert the inference information  $V_p$  and  $V_q$  and the fixed-length input.

There are two common pooling operations: average pooling and maximum pooling. Maximum pooling only retains the most robust features for the fused semantic information. It discards the weaker features to reduce the impact of noise and improve the robustness of the model. The disadvantage of maximum pooling is that it is easy to lose the feature location information. However, it can be made up by combining with average pooling. Therefore, the SCF-DMN model refers to the work of Chen [23], adopts the method of average pooling and maximum pooling set, and then splices the results to form the final fixed-length sentence pair vector  $V$ . The calculation process is shown in Formulas (24) and (25).

$$\begin{cases} v_{p\_ave} = \sum_{i=1}^m \frac{V_{p_i}}{m}, v_{p\_max}^{i=1} \\ v_{q\_ave} = \sum_{j=1}^n \frac{V_{q_j}}{n}, v_{q\_max}^{j=1} \end{cases} \quad (24)$$

$$V = [v_{p\_ave}; v_{p\_max}; v_{q\_ave}; v_{q\_max}] \quad (25)$$

where  $[;]$  is the splicing operation.

The sentence pair vector  $V$  is input into the multi-layer perceptual classifier to calculate the probability  $P_i$  of each tag in the corresponding task. For all tasks, the objective function of training is to minimize cross-entropy, as shown in Formula (26).

$$\mathcal{L} = - \sum_{i=1}^N [y_i \log P_i + (1 - y_i) \log (1 - P_i)] \quad (26)$$

where  $y_i$  is the relation label;  $N$  represents the total number of sentence pairs.

#### 4. Results

In this paper, an NVIDIA GeForce GTX 1070 video card is used in the experiment. All experimental codes are built based on the Theano framework. Based on the parameters of previous studies on the same datasets, the parameters of the semantic fusion deep matching network are set as follows:

- The maximum length of the sentence is set to 100. The model uses word2vec technology [28] to obtain the word embedding vector, where the dimension of the word embedding vector is 300, and GloVe-840B-300D is used to initialize the pretraining word vector. For words not included in the dictionary, the value of  $[-0.1, 0.1]$  is used for random initialization, and the word vector is kept updated with the training process.
- The dimensions of all LSTM networks in the model are 300, the activation function adopts the Relu function, and the weight parameters in the network are initialized randomly [29].
- The model optimization uses the Adam optimization algorithm [30], the default parameters  $1\alpha$  and  $2\alpha$  are set to 0.9 and 0.99, respectively, and the initial learning rate of the network is set to 0.0002.
- In order to prevent data overfitting, we use the Dropout strategy during training [31]. The input and output layers of each layer of the network are added to the Dropout layer and the dropout is set to 0.8.
- For the SNLI dataset, the training process, the number of training batches, and the number of verification matches are set to 32; for the Multi-NLI dataset, the number of training batches and the number of verification matches in the training process is set to 8.
- For the SNLI dataset, the number of matching modules in the matching layer is set to 3; for the Multi-NLI matching module, the number T is set to 2 [20].
- The models are tested on two datasets to see if they could produce the correct answer or not. The accuracy of their performances is then calculated as the main evaluation index.

##### 4.1. Experimental Results on SNLI Dataset

The reasoning models based on the model of Li and the model of this paper are designed and compared. The experimental results of each model on the SNLI dataset are shown in Table 4.

**Table 4.** Accuracy of each model on SNLI dataset.

Model	Training (%)	Test Set (%)
300D Tree-CNN (Mou et al., 2015)	83.3	82.1
300D NSE (Munkhdalai et al., 2016)	86.2	84.6
100D LSTMs with attention (Rocktäschel et al., 2015)	85.3	83.5
100D Deep Fusion LSTM (Liu et al., 2016)	85.2	84.6
300D Matching-LSTM (Wang et al., 2015)	92.0	86.1
200D Decomposable Attention Models (Parikh et al., 2016)	90.5	86.8
300D Re-read LSTM (Sha et al. 2016)	90.7	87.5
600D ESIM (Chen et al. 2017)	92.6	88.0
AF-DMN (Duan et.al., 2017)	94.5	88.6
Model in this paper	95.8	89.0

The coding-based reasoning model includes (1) the TBCNN model combining sentence structure information into sentence representation [26]; (2) memory-enhancing neural network NES proposed by Munkhdalai [32].

Matching based reasoning models include (1) matching reasoning model based on attention mechanism [33]; (2) matching LSTM model using match LSTM instead of traditional LSTM network based on matching model [34]; (3) re-read LSTM model focusing on attention vector interaction in sentences [35]; (4) deep fusion that pays more attention to text-to-text interaction LSTM model [36]; (5) decomposable attention model using attention

mechanism to decompose the responsible problem into subproblems that can be solved independently [22]; (6) ESIM model including chain LSTM and tree LSTM [23].

#### 4.2. Results on Multi-NLI Datasets

Compared to Multi-NLI datasets, comparison models can be divided into baseline and attention-based reasoning models, as shown in Table 5.

**Table 5.** Accuracy of each model on Multi-NLI dataset.

Model	Matching Set (%)	Unmatched Set (%)
CBOW (Williams et al., 2018)	64.8	64.5
BiLSTM (Williams et al., 2018)	66.9	66.9
ESIM (Chen et al., 2017)	76.8	75.8
AF-DMN (Duan et al., 2018)	76.9	76.3
Model in this paper	77.1	75.3

The performance of each model on the Multi-NLI dataset is shown. The baseline model CBOW and BiLSTM [13] have 64.8% and 66.9% accuracy on the Multi-NLI dataset and 64.5% and 66.9% on the unmatched set, respectively. The accuracy rates of ESIM [37] and AF-DMN model [20] based on attention mechanism are similar on the matching test set, which are 76.8% and 76.9%, respectively, and the accuracy on the unmatched set is 75.8% and 76.3%, respectively. In contrast, the accuracy of the SCF-DMN model designed in this paper is 77.1% on the matched dataset and 75.3% on the non-matching set.

## 5. Discussion

### 5.1. Analysis of Prediction Accuracy

The SNLI results show that the accuracy of the matching-based reasoning model is higher than that of the sentence coding-based reasoning model on the SNLI test set. The highest accuracy rate of the AF-DMN model is 88.6% of that of the AF-DMN model. The reasoning model based on semantic fusion depth matching network (SCF-DMN) designed in this paper achieves 95.8% and 89.0% accuracy in the SNLI training set and test set, respectively, which improves by 1.3% and 0.4% compared with the AF-DMN model. This shows that the SCF-DMN model has deeper reasoning depth and can capture the interactive information between sentences better than the AF-DMN model, indirectly indicating that the SCF-DMN model is added to the reasoning process. The syntactic structure information is effective and can promote the inference result.

The Multi-NLI results show that the performance of the SCF-DMN model is better than AF-DMN on the matching set, but the data on the unmatched set is lower than AF-DMN. This result may be because the data in the unmatched set does not appear in the training data completely. Therefore, some relations not shown in the training data may not be learned. At the same time, the simple bidirectional long-term and short-term memory network (BiLSTM) is only used in the coding layer, which may lead to the deviation of the content expression of complex sentences, resulting in a poor learning effect.

### 5.2. Analysis of Semantic Relevance

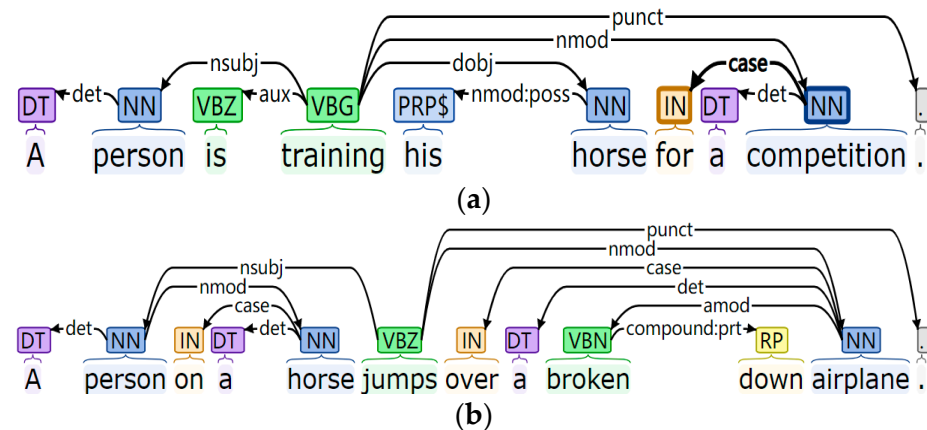
Figure 4 shows the visual results of semantic correlation between the premise sentence “a person is training his horse for a competition.” and the hypothetical sentence “a person on a horse jumper over a broken-down airplane.” the darker the color, the stronger the correlation between them.

The SCF-DMN model focuses on the close relationship between the core word “training” and the core word “jumps” in the hypothetical sentence, and “competition” is closely related to “airport”. At the same time, it is also concerned that the correlation between the subject “person” and “horse” in the premise sentence and the subject “person” and “horse” in the hypothetical sentence is significantly higher than that with other words.



**Figure 4.** Visual results of sentence to semantic correlation analysis.

Figure 5a,b, respectively, show the dependency syntactic relationship between the premise and hypothetical sentences. The result is consistent with the result of sentence pair correlation. Thus, it shows that adding syntactic structure information to the SCF-DMN model can promote the capture of sentence structure information and the interpretation of the reasoning process.



**Figure 5.** Dependency syntax. (a) Presupposition sentence; (b) hypothetical sentence.

### 5.3. Ablation Analysis

In order to verify each module's influence of the SCF-DMN model on the model, an ablation test of the module was carried out on the SNLI dataset. The specific test results and analysis are as follows:

#### 5.3.1. The Influence of Interactive Fusion Mode

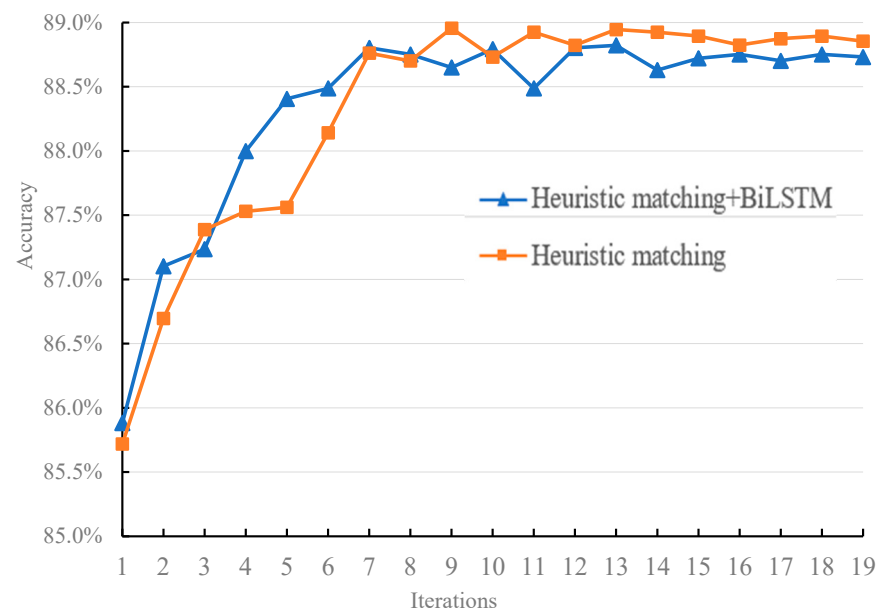
In order to explore the impact of interactive fusion on semantic fusion deep matching networks, this paper compares the impact of two interactive fusion methods on the performance of the SCF-DMN model, namely, heuristic matching mode and heuristic matching + BiLSTM network. The final experimental results are shown in Table 6.

**Table 6.** Comparison of the influence of different interactive fusion methods on semantic fusion depth matching network.

Interactive Integration Mode	Accuracy Rate (%)	Best Iteration	Training Time (h)	Hyper-Parameters
Heuristic matching + BiLSTM	88.7	9	26.87	47,071,203
Heuristic matching	89.0	8	22.87	43,285,803

The experimental results show that the heuristic matching method improves the accuracy rate by 0.3% and reduces the number of super parameters by 8%. Moreover, under the same training parameters, the training time is also reduced by 4 h. However, since there is not much correlation between the interactive information of sentences in the interaction fusion layer, the BiLSTM network does not improve the model's performance. On the contrary, it will introduce redundant information to reduce the reasoning accuracy and increase the super parameters.

Figure 6 shows the learning curve of the SFC-DMN model with the heuristic matching method and SCF-DMN model with BiLSTM network and heuristic matching method on the SNLI dataset. The figure shows that the learning curve of the SCF-DMN model with the heuristic matching method tends to stabilize faster. As a result, the final stable state's accuracy of the former model is higher than that of the latter.

**Figure 6.** The learning curve of SCF-DMN (different fusion layers) on SNLI.

After the above comparison, it can be found that the information of the fusion layer has a certain impact on improving the model's accuracy. This is because the heuristic matching method can emphasize the similarity and differences between sentence pairs. However, at the same time, if the fusion information is further strengthened, it will lead to information redundancy and reduce the model's performance.

### 5.3.2. The Influence of Syntactic Structure Information

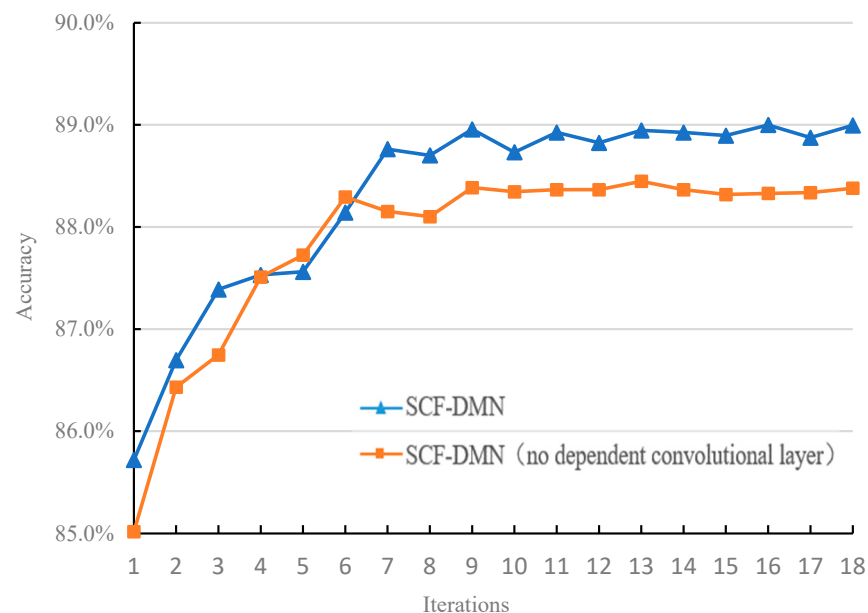
In order to analyze the influence of syntactic structure information on semantic fusion depth matching network, this paper designs and compares the performance of the SCF-DMN model and SCF-DMN model, excluding dependent convolution layer on SNLI dataset. The experimental results are shown in Table 7.

**Table 7.** Comparison of effects of dependent convolution layer on SCF-DMN model performance.

Model Composition	Accuracy Rate (%)	Training Time (H)	Hyper-Parameters
SCF-DMN	89.0	22.87	43,285,803
SCF-DMN (without dependent convolution layer)	88.3	14.15	23,535,603

As shown in Table 7, compared with the SCF-DMN model, the training time of the SCF-DMN model without a dependent convolution layer is reduced by 38.1%. The accuracy rate on the SNLI dataset is reduced by 0.7%, and the accuracy rate is reduced by 0.3% compared with the AF-DMN model.

As shown in Figure 7, the learning curves of each model on the SNLI dataset are shown. It is found that the convergence speed of the SCF-DMN model (without dependency convolution layer) is faster than that of the SCF-DMN model. However, after it is stable, the accuracy rate of the SCF-DMN model on SNLI data is significantly higher than that of the SCF-DMN model (without dependency convolution layer). It is consistent with the results in Table 7, indicating the inference process between sentences by dependency convolution layer. Thus, it has an essential promoting effect.

**Figure 7.** The learning curve on SNLI of SCF-DMN (different fusion layers).

Although this paper explores sentence representation and reasoning methods from the perspective of the reasoning model, which improves the reasoning accuracy to some extent, it is far from achieving the best effect. Given this, future research can be further studied from the following limitations of this study, including improving hardware quality with custom ASIC or FPGA [38], considering the heavy calculation requirements for the proposed model.

## 6. Conclusions

At present, the field of NLI has become another research hotspot after the field of images. Many scholars have carried out research work in this field. This paper first introduces some basic reasoning models with their series of challenges. It then introduces the principles, advantages, and disadvantages of the AF-DMN and tree convolution networks. Finally, this paper proposes a deep fusion matching network to the reasoning model, aiming at the lack of reasoning depth and interpretability. The network consists of the coding, matching, dependency convolutional, information aggregation, and inference prediction



layers. The matching layer is improved based on the deep matching network. The heuristic matching algorithm replaces the complex neural network as the interactive fusion mode of matching information, which improves the reasoning depth and reduces the complexity of the model. The dependency convolutional layer uses a tree convolutional network to extract the structural information of sentences along the dependency tree structure of sentences to make up for the unexplainability of the reasoning process. The experimental results show that the reasoning effect of the network is superior to that of the shallow matching reasoning model, and the accuracy rate on the SNLI test set reaches 89.0%. At the same time, it is found from the visualization results that the explanatory ability of the reasoning process by relying on the convolutional layer is significantly improved.

However, due to cognition and understanding, sentence representational reasoning is still the focus and difficulty in this field. The inference model in this paper is designed only considering the characteristics of the inference domain of sentence representation. Its design and performance are narrowed to this specific idea and purpose. However, with the development of transfer learning, it is found that introducing other natural language processing tasks with similar semantic characteristics to the target domain can help improve the performance of the target domain.

**Author Contributions:** Conceptualization, W.Z. and B.Y.; methodology, S.L.; software, Y.Z.; formal analysis, L.Y. and Y.Z.; data curation, J.T.; writing—original draft preparation, J.T. and L.Y.; writing—review and editing, L.Y. and W.Z.; visualization, J.T.; supervision, B.Y. and S.L.; funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** Supported by the Sichuan Science and Technology Program (2021YFQ0003).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data used in this paper are open-source data which are available at <https://nlp.stanford.edu/projects/snli/> (accessed on 1 February 2022) and <https://cims.nyu.edu/~sbowman/multinli/> (accessed on 1 February 2022).

**Acknowledgments:** The authors express their sincere appreciation and profound gratitude to research assistants Xia Tian, Xubin Ni, Xiaobing Chen, and Yueming Ding for their help and support in collecting and sorting the data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hu, B.; Lu, Z.; Li, H.; Chen, Q. Convolutional neural network architectures for matching natural language sentences. *Adv. Neural Inf. Process. Syst.* **2014**, *2*, 2042–2050.
2. Tan, Z.; Chen, J.; Kang, Q.; Zhou, M.; Abusorrah, A.; Sedraoui, K. Dynamic embedding projection-gated convolutional neural networks for text classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 973–982. [CrossRef] [PubMed]
3. Leng, X.-L.; Miao, X.-A.; Liu, T. Using recurrent neural network structure with Enhanced Multi-Head Self-Attention for sentiment analysis. *Multimed. Tools Appl.* **2021**, *80*, 12581–12600. [CrossRef]
4. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
5. MacCartney, B. *Natural Language Inference*; Stanford University: Stanford, CA, USA, 2009.
6. Liu, Y.; Guan, W.; Lu, D.; Zou, X. A label-oriented loss function for learning sentence representations. *Comput. Speech Lang.* **2021**, *66*, 101165. [CrossRef]
7. Moldovan, D.; Clark, C.; Harabagiu, S.; Maiorano, S. Cogex: A logic prover for question answering. In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Stroudsburg, PA, USA, 27 May–1 June 2003; Volume 1, pp. 87–93.
8. Raina, R.; Ng, A.Y.; Manning, C.D. Robust textual inference via learning and abductive reasoning. In Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05), Pittsburgh, PA, USA, 9–13 July 2005; pp. 1099–1105.
9. Le, N.Q.K.; Yapp, E.K.Y.; Nagasundaram, N.; Yeh, H.-Y. Classifying promoters by interpreting the hidden information of DNA sequences via deep learning and combination of continuous FastText N-grams. *Front. Bioeng. Biotechnol.* **2019**, *305*. [CrossRef]
10. Le, N.Q.K.; Ho, Q.-T.; Nguyen, T.-T.-D.; Ou, Y.-Y. A transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers from sequence information. *Brief. Bioinform.* **2021**, *22*, bbab005. [CrossRef] [PubMed]

11. Ni, X.; Yin, L.; Chen, X.; Liu, S.; Yang, B.; Zheng, W. Semantic representation for visual reasoning. In *MATEC Web of Conferences*; EDP Sciences: Les Ulis, France, 2019; p. 02006.
12. Zhu, Y.; Ko, T.; Snyder, D.; Mak, B.; Povey, D. Self-attentive speaker embeddings for text-independent speaker verification. In *Proceedings of the Interspeech 2018, 19th Annual Conference of the International Speech Communication Association*, Hyderabad, India, 2–6 September 2018; pp. 3573–3577.
13. Williams, A.; Nangia, N.; Bowman, S.R. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv* **2017**, arXiv:1704.05426.
14. Liu, Y.; Wan, Y.; He, L.; Peng, H.; Yu, P.S. KG-BART: Knowledge graph-augmented BART for generative commonsense reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, online. 2–9 February 2021; pp. 6418–6425.
15. Quamer, W.; Jain, P.K.; Rai, A.; Saravanan, V.; Pamula, R.; Kumar, C. SACNN: Self-attentive convolutional neural network model for natural language inference. *Trans. Asian Low-Resour. Lang. Inf. Process.* **2021**, *20*, 1–16. [[CrossRef](#)]
16. Cheng, Z.; Dai, X.; Huang, S.; Chen, J. Variational Explanation Generator: Generating Explanation for Natural Language Inference using Variational Auto-Encoder. *Int. J. Comput. Inf. Eng.* **2021**, *15*, 119–125.
17. Wu, H.; Huang, J. Relative Position Representation over Interaction Space for Natural Language Inference. In *Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME)*, Shenzhen, China, 5–9 July 2021; pp. 1–6.
18. Peng, H.; Li, J.; Wang, S.; Wang, L.; Gong, Q.; Yang, R.; Li, B.; Philip, S.Y.; He, L. Hierarchical taxonomy-aware and attentional graph capsule RCNNs for large-scale multi-label text classification. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 2505–2519. [[CrossRef](#)]
19. Johnson, M.; Schuster, M.; Le, Q.V.; Krikun, M.; Wu, Y.; Chen, Z.; Thorat, N.; Viégas, F.; Wattenberg, M.; Corrado, G. Google’s multilingual neural machine translation system: Enabling zero-shot translation. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 339–351. [[CrossRef](#)]
20. Duan, C.; Cui, L.; Chen, X.; Wei, F.; Zhu, C.; Zhao, T. Attention-Fused Deep Matching Network for Natural Language Inference. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, 13–19 July 2018; pp. 4033–4040.
21. Mou, L.; Peng, H.; Li, G.; Xu, Y.; Zhang, L.; Jin, Z. Discriminative neural sentence modeling by tree-based convolution. *arXiv* **2015**, arXiv:1504.01106.
22. Parikh, A.P.; Täckström, O.; Das, D.; Uszkoreit, J. A decomposable attention model for natural language inference. *arXiv* **2016**, arXiv:1606.01933.
23. Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; Inkpen, D. Enhanced LSTM for natural language inference. *arXiv* **2016**, arXiv:1609.06038.
24. Chen, D.; Manning, C.D. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 25–29 October 2014; pp. 740–750.
25. De Marneffe, M.-C.; MacCartney, B.; Manning, C.D. Generating typed dependency parses from phrase structure parses. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006*, Genoa, Italy, 22–28 May 2006; pp. 449–454.
26. Mou, L.; Men, R.; Li, G.; Xu, Y.; Zhang, L.; Yan, R.; Jin, Z. Natural language inference by tree-based convolution and heuristic matching. *arXiv* **2015**, arXiv:1512.08422.
27. Graves, A.; Jaitly, N.; Mohamed, A.-r. Hybrid speech recognition with deep bidirectional LSTM. In *Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, 8–12 December 2013; pp. 273–278.
28. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, *26*.
29. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Sardinia, Italy, 13–15 May 2010; pp. 249–256.
30. Zhang, Z. Improved adam optimizer for deep neural networks. In *Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, Banff, AB, Canada, 4–6 June 2018; pp. 1–2.
31. Bowman, S.R.; Angeli, G.; Potts, C.; Manning, C.D. A large annotated corpus for learning natural language inference. *arXiv* **2015**, arXiv:1508.05326.
32. Munkhdalai, T.; Yu, H. Neural tree indexers for text understanding. In *Proceedings of the Conference. Association for Computational Linguistics. Meeting*, Vancouver, BC, Canada, 30 July–4 August 2017; p. 11.
33. Graves, A.; Wayne, G.; Reynolds, M.; Harley, T.; Danihelka, I.; Grabska-Barwińska, A.; Colmenarejo, S.G.; Grefenstette, E.; Ramalho, T.; Agapiou, J. Hybrid computing using a neural network with dynamic external memory. *Nature* **2016**, *538*, 471–476. [[CrossRef](#)] [[PubMed](#)]
34. Wang, S.; Jiang, J. Learning natural language inference with LSTM. *arXiv* **2015**, arXiv:1512.08849.
35. Sha, L.; Chang, B.; Sui, Z.; Li, S. Reading and thinking: Re-read lstm unit for textual entailment recognition. In *Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, 11–16 December 2016; The COLING 2016 Organizing Committee: Osaka, Japan, 2016; pp. 2870–2879.
36. Liu, P.; Qiu, X.; Chen, J.; Huang, X.-J. Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, 7–12 August 2016; Volume 1, pp. 1034–1043.

- 
37. Liu, Y.; Zhao, T.; Chai, Y.; Jiang, Y. A Word Elimination Strategy for Learning Document Representation. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *466*, 012091. [[CrossRef](#)]
  38. Bhowmik, P.; Pantho, J.H.; Mbongue, J.M.; Bobda, C. ESCA: Event-based split-CNN architecture with data-level parallelism on ultrascale+ FPGA. In Proceedings of the 2021 IEEE 29th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), Orlando, FL, USA, 9–12 May 2021; pp. 176–180.