Robot for Mathematics College Entrance Examination

Hongguang Fu[1], Jingzhong Zhang[1], Xiuqin Zhong[1,], Mingkai Zha[1], Li Liu[2]*
zhongxiuqin@uestc.edu.cn
[1]University of Electronic Science and Technology of China, Chengdu, China
[2]Southwest Minzu University, Chengdu, China

Abstract: *Automated mathematics has attracted more and more interests to mathematicians and computer scientists. In this article, we propose a new solution, Robot for Mathematics College Entrance Examination (RMCEE), to automatically solve elementary mathematical problems. RMCEE took the college entrance exam of China in 2017 and scored 105 points out of 150. RMCEE takes as input Chinese natural language and outputs human-readable solution processes. It leverages artificial intelligence technologies in multiple domains such as the knowledge graph, natural language understanding, cognitive reasoning, and deep learning. Our project outperforms other similar projects such as Todai Robot of Japan in that RMCEE proposes a natural language understanding model of entity combinations based on knowledge graph, and implemented human-readable solution processes based on a novel cognitive reasoning model.*

1. Introduction

The famous mathematician Descartes said that any problem can be converted into a mathematical problem and any mathematical problem can be converted into solving equations (see [1]). Therefore, if we can solve equations automatically, and eventually solve a big category of mathematical problems, we may expect to attack a multitude of problems in the real world directly or indirectly.

In this direction, many pioneering mathematicians has made profound progress. Tarski first proved the possibility of elementary geometry mechanization using algebraic methods (see [11]). In the 1970's, Wen-Tsun Wu presented Ritt-Wu's Method (see [12]) to prove a lot of theorems systematically (such as Pythagoras'theorem, Simson's theorem, Feuerbach's theorem, and so on). J. Z. Zhang introduced a geometric invariant method (see [13]) with readable proofs. Stimulated by them, many new methods appeared such as the resultant elimination method (see [14]), Grobner basis (see [15]), and so on. These methods are called algebraic elimination methods, they are however hardly comprehensible and not intuitive enough for humans to understand. While logic methods (see [16], [17] and [18]) proposed have always been the dream of mankind, their applications are limited due to the massive search space. Recently, readable simplification of trigonometric expressions has been put forward by Fu (see [19]), and research has been done in different sub areas, such as algebra word problems (see [20]), geometry problems (see [21]).

With these pioneering works, with the big picture of AI leading the current wave of new technology revolution, it is the right time for automated mathematics to make an impact to real world problems. Much effort has been made towards this goal. The Allen Institute for AI at University of Washington proposed an end to end Geometry Problem Solver (GPS) for SAT questions. GPS takes text input in natural language and returns the answer to the question (see [7]). Project Aristo (see [22]) contributes to research on computerized solving of elementary school science and math exam problems, they use a deep learning-based approach to solve the problem of premise selection (see [23]), pre-university entrance exams (see [8]), etc. While these projects have their achievement, their focus is on the utilization of AI technologies. Like IBM Watson project in 2011[2], they are proud of processing natural language, understanding the input geometric graphs, etc. On the other hand, they are relatively weak in solving the mathematical problems themselves. These solvers can only solve standardized tests, and the correctness of the answers is to be improved.

An example utilizing the state-of-the-art in mathematics mechanization is the Todai Robot Project in Japan (see [8]). It aims to develop a mechanical problem-solving system that can pass the University of Tokyo's entrance examination (see [3] and [4]). It began in 2011 and terminated in 2016, utilizing the theory and practice of the computer algebra software Maple (see [5]) and automatic theorem proving (see [6]). Unfortunately, it does not contain a natural language processing model and can take inputs with limited formats.

例 1

设 F_1 是椭圆 C: $\frac{x^2}{a^2}+\frac{y^2}{b^2}$(a>b>0) 的左焦点，M 是 C 上一点，且 MF_1 与 x 轴垂直，若|MF1|= $\frac{3}{2}$ ，离心率为$\frac{1}{2}$.（1）求椭圆 C 的方程；（2）以椭圆 C 的左顶点 A 为△三角形 ABD 的直角顶点，边 AB, AD 与椭圆 C 交于 B, D 两点，求△ABD 面积的最大值.

F_1 is the left focus of ellipse C: $\frac{x^2}{a^2}+\frac{y^2}{b^2}$ (a>b>0), M is on C, and, and MF_1 is perpendicular to X axis. If $MF_1 = \frac{3}{2}$, the eccentricity of ellipse C is $\frac{1}{2}$. (1) Calculate the equation of ellipse C; (2) Point A is the left vertex of ellipse C, point A is the right vertex of △ABD，AB and ellipse C cross at point B, AD and ellipse C cross at point D. Calculate the maximum value of area of △ABD.

Figure 1.1 An example question in a mathematics test

In 2015, China launched a project comparable to Todai, the National High Technology Research and Development Program of China "Key technologies and systems of human-like intelligence based on big data". This program focuses on the annual university entrance examination of four subjects, Chinese (see [9]), history, geography (see [10]), and mathematics. This paper presents the part of the work of mathematics in this project. We aim to develop a mathematics robot for automated solving of elementary mathematics problems just like human being, which can automatically answer both standard and non-standard questions with the input of text in natural language and the output of human readable solutions. Figure 1.1 shows an example question (Example 1) in a mathematics exam. It consists of the topic description and two sub questions. This problem is different from and more complex than problems handled by existing approaches to automated question answering. It handles natural language input and includes knowledge in 2D geometry, analytic geometry, maximum value problems and so on.

In this paper, we firstly present the architecture of RMCEE in section 2, then further describe certain key technologies in detail in section 3. After that, experiments are performed on sample elementary mathematics problems. Finally, conclusions and areas of future work will be discussed in the last section.

2. The architecture

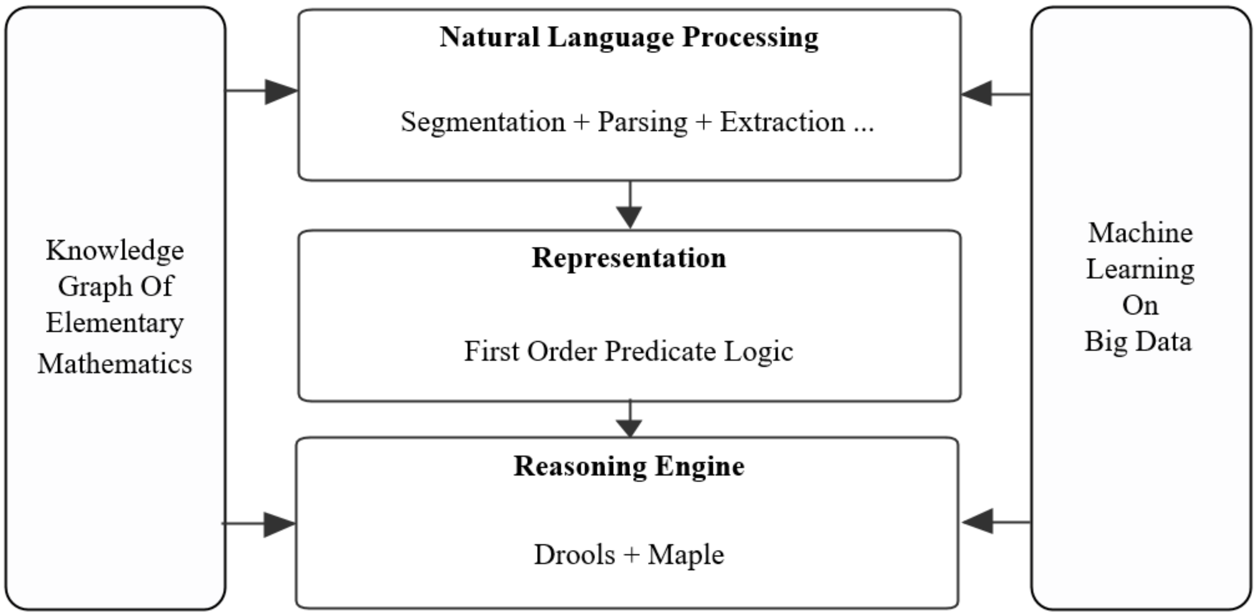


Figure 2.1 The architecture of RMCEE

The architecture of RMCEE is shown in figure 2.1. It includes five parts. The natural language processing module segments, parses, and understands the conditions and conclusions of math problems. The automated reasoning module solves the problems by calculating, reasoning and so on. The natural language processing module and the automated reasoning module are connected by the first order predicate logic forms. A knowledge graph of elementary mathematics (see [24]) and a machine learning framework based on big data are two modules, which effectively support natural language processing and automated problem solving. In the following, we will illustrate some key technologies of RMCEE in detail.

3. Key Technologies

3.1 An elementary mathematics knowledge graph

As we know, construction of an elementary mathematics knowledge graph is primary.

First, we manually construct a core ontology of elementary mathematics based on Chinese textbooks. Next, we extend the domain core ontology based on a synonym list (such as word forest (see [26]) and general ontology (such as WordNet (see [27]). Finally, we use deep learning technology to extract entities and relations from mathematical texts on the internet, in order to realize the dynamic extension and improvement of the elementary mathematics knowledge graph. In this paper, we implemented it by Neo4j (see [25]). In the following, we will describe the process step by step.

Core ontology

We adopt a top-down method in the construction of an elementary mathematics ontology. First, we define two main classes of algebra and geometry. In the algebraic class, we have several subclasses such as sequence, vector, set, function, math expression. In the geometric class we have several subclasses such as angle, curve/line, polygon, polyhedron. This includes elementary mathematics entities and relations, for example, *PointRelation:A, LineCrossCircleRelation{line=CP, circle=⊙O, crossPoints=[C, M], crossPointNum=2}*, etc., as shown in table 1. Furthermore, we improve the attributes of mathematical entities and relations. Some examples include: SubClassOf relation between entities is constructed which can inherit all the attributes from its parent node; PreviousOf relation between entities is constructed which can deal with new concepts based on existing entities and relations; the symmetry of mathematical relations (such as *Intersect*, *Parallel*) is introduced which can process statements such as *Line AB intersects with circle O* and *Circle O intersects with line AB*; synonymy relations are constructed for normalization, and antonymic relations are constructed for propositional calculus.

Table 1 Some elementary mathematics relations and corresponding descriptions

Relations	The description of relations
PointOnLineRelation	点在直线上关系
PointOutOfLineRelation	点在直线外关系
PlaneRelation	平面关系
LineInPlaneRelation	直线在平面内关系
LineParallelPlaneRelation	线面平行关系
LineCrossPlaneRelation	直线与平面相交关系
LinePrepPlaneRelation	直线与平面垂直关系
DualPlaneCrossRelation	面面相交关系
DualPlanePrepRelation	面面垂直关系

Mathematics synonym library

After building the core ontology, because the descriptions of mathematical entities and relations in different mathematical texts will change dramatically, we construct a mathematics synonym library based on the synonym library (see [28]) to manage the situation, so that sentences can be expanded locally and horizontally at the word level, and expressions beyond synonyms can be processed similarly.

Automatic extension of the knowledge graph

Combining relation extraction based on pattern matching and dependency parsing based on HanLP (see [29]) with deep learning based on Bi-LSTM+Attention (see [30]), we propose a new method of automatic extraction of elementary mathematical entities and relations from text. At the same time, we realize automatic verification of relational validity by OpenKG (see [31]).

There are two sources for the corpus, one is data from the internet, and the other is self-built elementary mathematics exercises. Among them, there are 856 encyclopedia pages and 16951 unstructured texts on the internet. There are 3355 self-built elementary mathematics exercises.

Finally, a total of 685 entities and 22638 triples are extracted automatically by this method and corpus. Among them, there are 12264 triples extracted by pattern matching, 4753 triples extracted by dependency parsing, 5535 triples extracted by deep learning, and 86 triples obtained by graph reasoning, which enriches and improves the elementary mathematics knowledge graph. Our elementary mathematics knowledge graph is shown in figure 3.1.

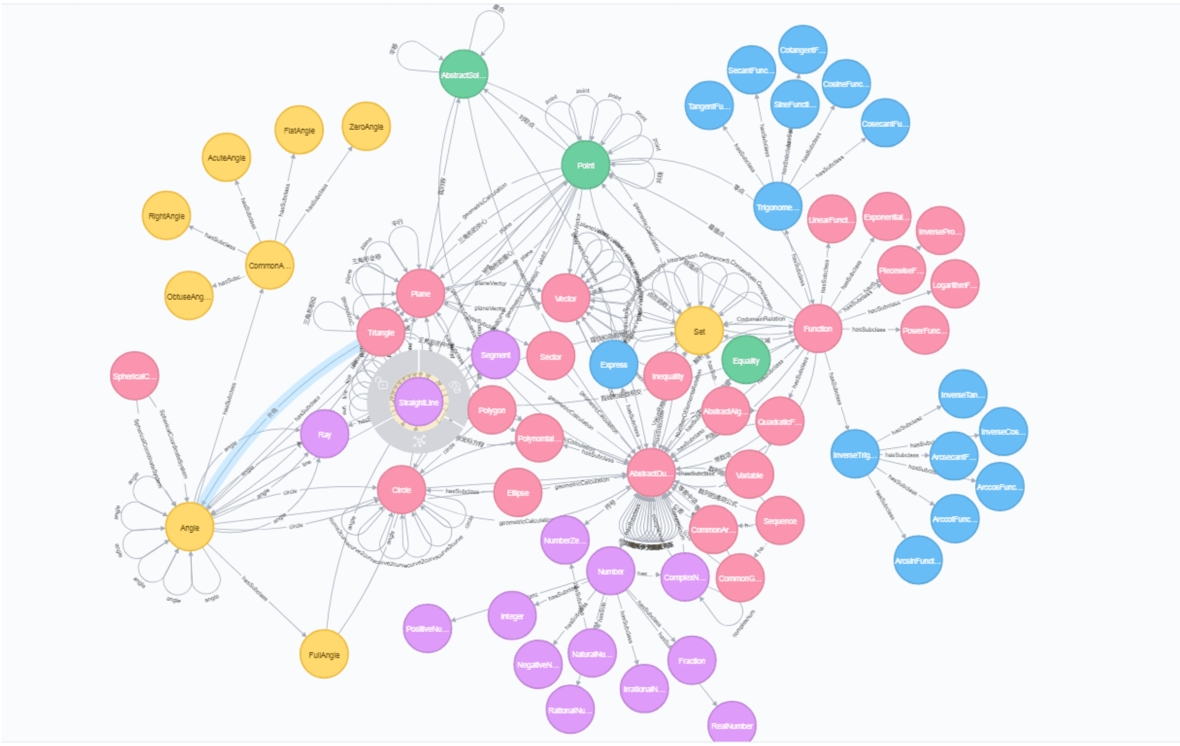


Figure 3.1 An elementary mathematics knowledge graph

Moreover, a rule base with about 1700 mathematics rules (rules including axioms, definitions, theorems, corollaries, etc. of elementary mathematics) is constructed too, as shown in table 2, for instance, rule *"DefinitionOfNeutralityLineOfTriangle"*, rule *"NeutralityLineOfTriangle"*, etc. Among them, rule *"DefinitionOfNeutralityLineOfTriangle"* implemented by Drools (see [32]) is shown in table 3.

Table 2 The table of elementary rules and corresponding descriptions

Rules	The description of rules
DefinitionOfNeutralityLineOfTriangle	中位线的判定定理：连结三角形两边中点的线段叫做三角形的中位线
NeutralityLineOfTriangle	中位线定理：经过三角形一边的中点,平行于第二边的直线必平分第三边
DualPlaneParallelDecisionTheorem	两个平面平行的判定定理：一个平面内的两条相交直线与另一个平面平行，则两个平面平行
DualPlaneParallelDecisionTheorem2	两个平面平行的判定定理：垂直于同一条直线的两个平面平行
LineInOnePlaneParallelToAnotherPlane	相互平行的两个平面里，其中一个平面上的任意直线都平行于第二个平面

Table 3 The rule "DefinitionOfNeutralityLineOfTriangle" implemented by Drools

```
rule "DefinitionOfNeutralityLineOfTriangle"   //rule name
   when   // match conditions
      $triangle : Triangle()
      MiddlePointOfSegmentRelation($triangle.isContainSegment(segment), $segment1 :
segment, $point1 : point)
      MiddlePointOfSegmentRelation(!segment.equals($segment1),
$triangle.isContainSegment(segment), $segment2 : segment, $point2 : point)
      eval(pointUtil.isConnective($point1, $point2))
   then   // execute conclusions
      Segment seg = new Segment($point1, $point2);
      List&lt;Segment&gt; segments = $triangle.getSegments();
      segments.remove($segment1);
      segments.remove($segment2);
      NeutralityLineOfTriangleRelation mid = new NeutralityLineOfTriangleRelation(seg,
$triangle, segments.size() == 1 ? segments.get(0) : null);
      insert(mid);    // insert knowledge base
end
```

Description: The *when* part is the conditions of the rule matching. For example, consider the rule covering a triangle and two *MiddlePointOfSegment* objects. The *segment* is determined by the method inside the triangle (for example: *$triangle.isContainSegment(segment))*, and *eval* is equivalent to the *if* condition, and the *then* part is the rule execution content. In this example, the result of this rule is to generate a *NeutralityLineOfTriangle* object, and then it can insert the knowledge base through the *insert* keyword.

Thus, an elementary mathematics knowledge base is constructed which can provide semantic supports for natural language processing and automatic problem solving.

3.2 A natural language processing module

Our knowledge graph describes the relations between entities in the form of triples. We still construct about 8000 math relation templates manually based on corpus and exercises, as is shown in figure

3.2, so simple elementary mathematical natural language description can be understood by template matching and knowledge graph reasoning.

```
<template id="CIRCLE_MIDDLE_POINT_OF_ARC">
    <description>弧中点</description>
    <pattern>^#圆#%circle%(中)#弧#%arc%(的|)(中点)#是为#?%point%$</pattern>
    <pattern>^#圆#%circle%(中)#弧#%arc%(的|)(中点)(在)#一点#%point%(处)$</pattern>
    <pattern>^#圆#%circle%(中)#一点#?%point%(平分)#弧#%arc%$</pattern>
    <pattern>^#圆#%circle%(中)#一点#?%point%#是为##弧#%arc%(的|)(中点)$</pattern>
    <semantics type="JSON">
        <![CDATA[{"service":"cn.tsinghuabigdata.unclosedShape","code":"geo.JsonMiddlePointOfArcRelation",
        "data":{"arc":"%arc%","point":"%point%","arcType":"@arcType"}}]]>
        <![CDATA[{"service":"cn.tsinghuabigdata.closedShape","code":"geo.JsonCircleRelation",
            "data":{"circle":"%circle%"}
        }]]>

    </semantics>
</template>
```

Figure 3.2 An example of math relation templates

Entity Combinations Based on Knowledge Graph

In fact, elementary mathematics problems often contain a large number of complex sentences and compound sentences. Therefore, in order to solve these problems, we propose a novel math relation understanding model *Entity Combinations Based on Knowledge Graph*, as is shown in figure 3.3, which can handle problems that are not correctly understood by accurate matching.

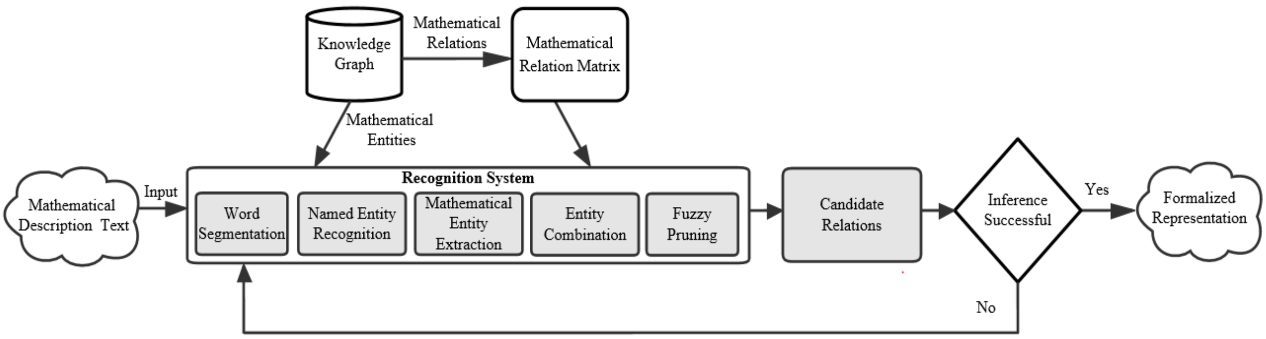


Figure 3.3 Entity combinations based on knowledge graph

First, we add the math lexicon and thesaurus, and then execute the word segmentation and the part of speech tagging with CRF (Conditional Random Field). Next, the named entity recognition is implemented by CRF++ training on manually-Tagged math entities and LaTeX math formulas (such as *line AB*, *triangle ABC*, *plane ABC*, *Expression* $y = a * x + b$ and so on). Finally, in addition to the traditional technologies above, a novel math relation understanding model *Entity Combinations Based On Knowledge Graph* is used to manage co-reference resolution, which is the introduction of variables such as entities and attributes of entities based on elementary mathematics ontology (such as domain, range, attributes and so on), in order to support the computing and reasoning needs of the problem solving module.

If a mathematical problem is known, the entities in the problem are annotated by labeling named entities. A sequence of entities is used to represent all entities and the relations between entity sequences (e1, e2, e3,...) are expressed as a real symmetric matrix A_{ij}, where a_{ij}=Rel(e_i, e_j) represents the candidate relations between entity i and entity j. A concrete example describe the process of matrix A_{ij} construction follows.

Assuming there is a mathematical problem: problem 1*"直线 L 与三角形 ABC 的 BC 边相交于点 P"(The line L intersects the side BC of the triangle ABC at point P)*. The matrix A_{ij} construction steps are as follows:

(1). Extract entities in problem 1: Line L, Triangle ABC, Side BC, and Point P, then obtain an entity sequence (e1, e2, e3, e4).

(2). Construct a matrix A_{ij}= (e1, e2, e3, e4)×(e1, e2, e3, e4) based on entity sequence (e1, e2, e3, e4).

(3). Execute the entities combinations based on the knowledge graph, and then fill the possible relations between each entity and other entities in a_{ij}=Rel(e_i, e_j). For example, fill in the candidate relations between edge BC and line L, because edge BC is a subclass of segment, and segment is a subclass of line, so the relations between edge BC and line L correspond to the relations between line l_1 and line l_2, while the candidate relations between line l_1 and line l_2 are 1) LineCrossRelation, 2) LineParallelRelation, 3) LinePerpRelation, 4) LineFactedRelation, so the corresponding relation sequence should be filled in a_{ij}=Rel(e_i, e_j), where e_i=line L and e_j= Side BC. The entity combinations based on the knowledge graph are shown in table 4.

Table 4 The entity combinations based on knowledge graph

The line L intersects the side BC of the triangle ABC at point P				
	Line L	Triangle ABC	Side BC	Point P
Line L	/			
Triangle ABC	1. LineCrossTriangleRelation 2. LineOutOfTriangleRelation	/		
Side BC	1. LineCrossRelation 2. LineParallelRelation 3. LinePerpRelation 4. LineFacetedRelation	1. FigureHasEdgesRelation	/	
Point P	1. PointOnLineRelation 2. PointOutOfLineRelation	1. PointInTriangleRelation 2. PointOnTriangleRelation 3. PointOutOfTriangleRelation	1. PointOnLineRelation 2. PointOutOfLineRelation 3. MiddlePointOfSegment-Relation	/

(4).Using the method of fuzzy pruning to delete the relations with low reliability. The candidate relations obtained from entity combinations can also be verified by the automatic problem solving model. If the problem is solved successfully, the candidate relation is correct.

Finally, we get the semantic analysis of the sentence: *Line L has a LineCrossTriangleRelation with Triangle ABC, a LineCrossRelation with Side BC, and a PointOnLineRelation with Point P. Triangle*

ABC has a FigureHasEdgeRelation with Side BC, and a PointInTriangleRelation with Point P. Side BC has a PointOnLineRelation with the Side BC.

Thus, the results of all the conditions and conclusions coming from the natural language processing module are produced. Then it will be converted into first order predicate logic forms. Subsequently, it will be regarded as the input to automatic problem solving.

3.3 A hybrid reasoning model

Elementary mathematics includes elementary algebra, plane geometry, solid geometry, analytic geometry, set, function, series, complex numbers. Because of the variety and complexities of elementary mathematics, a hybrid reasoning model (as is shown in figure 3.4) is put forward. It integrates logical reasoning, symbolic computation and deep learning.

Traditional rule-based bidirectional reasoning is achieved based on Drools rule engine. On one hand, backward reasoning based on automatic reduction is applied, which can produce new facts from known conclusions. On the other hand, forward reasoning based on symbolic computation and complex logical reasoning is implemented, which can produce new facts from the known conditions. Various possible symbolic computations such as simplifications and calculations are carried out by Maple (see [5]), which will be called as needed.

Moreover, complex logical reasoning is executed based not only on domain rules in the rule base, but also on all kinds of policies in the policy library and cognitive reasoning models. The latter two will be described in the following section.

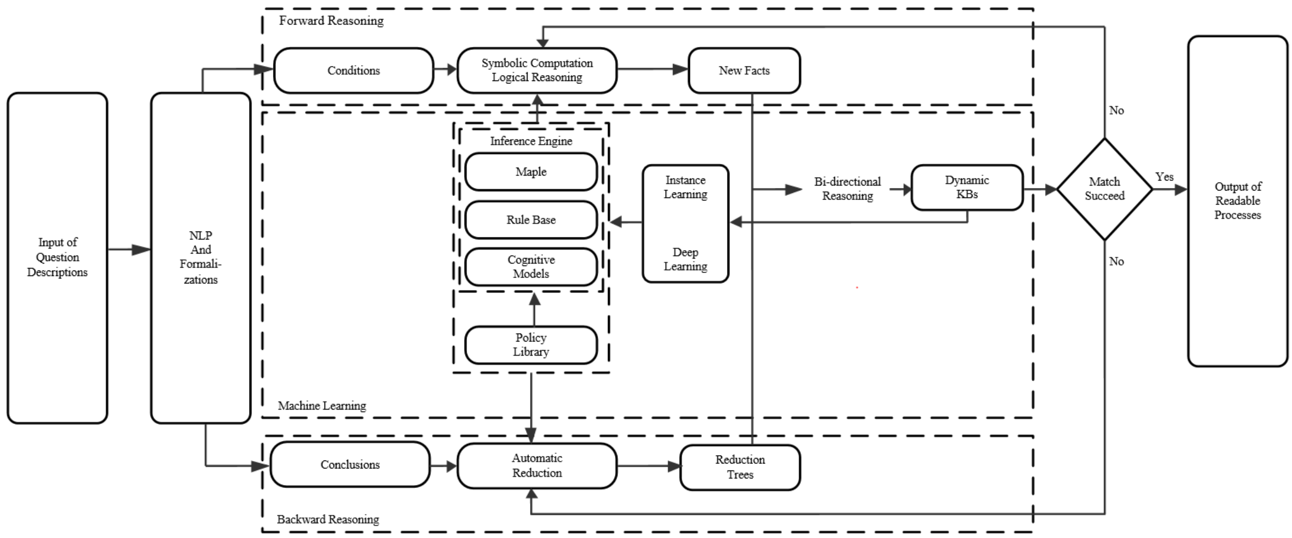


Figure 3.4 the Hybrid Reasoning Model

Policy Library

The policy library is composed of different branching policies, which can be divided into three levels, as is shown in figure 3.5. The first layer is the pre-processing branching policies, which includes auxiliary point and auxiliary line adding policy and variable introduction policy. The second layer is the external branching policies, which contains different methods to solve the same problem, such as mathematical induction method, the unified method, anti-evidence method; the third layer is the internal branching policies, which describes some feature-based skills to reduce the difficulties of problems, such as expression split, quadratic sum, straight slopes.

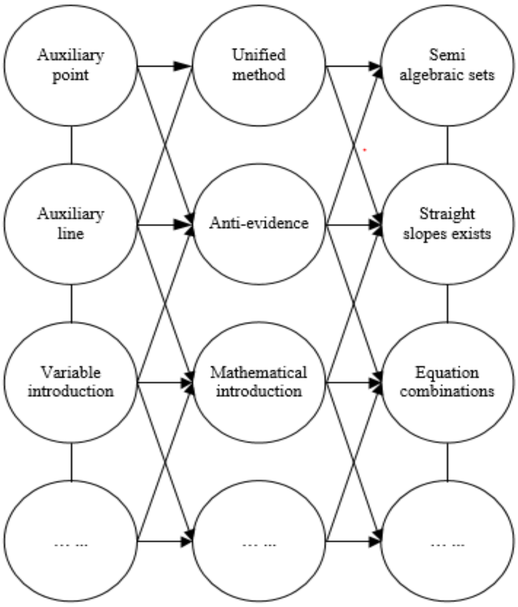


Figure 3.5 The Policy Library

Cognitive Reasoning Model

Through bidirectional reasoning, a big, well-structured data set will be generated when solving thousands of problems. By observing and summarizing the results of numerous cases, human beings obtain the cognition rules of the objective world. In a similar way a rule-based machine learning algorithm is adopted to extract pieces of problem solving sequences, and a DNA sequence assembly algorithm (see [33]) is applied to assemble the problem solving sequences. The generating process of a cognitive model is shown in figure 3.6. Finally a series of cognitive reasoning models (about 300 models) are generated, one of them is shown in figure 3.7.

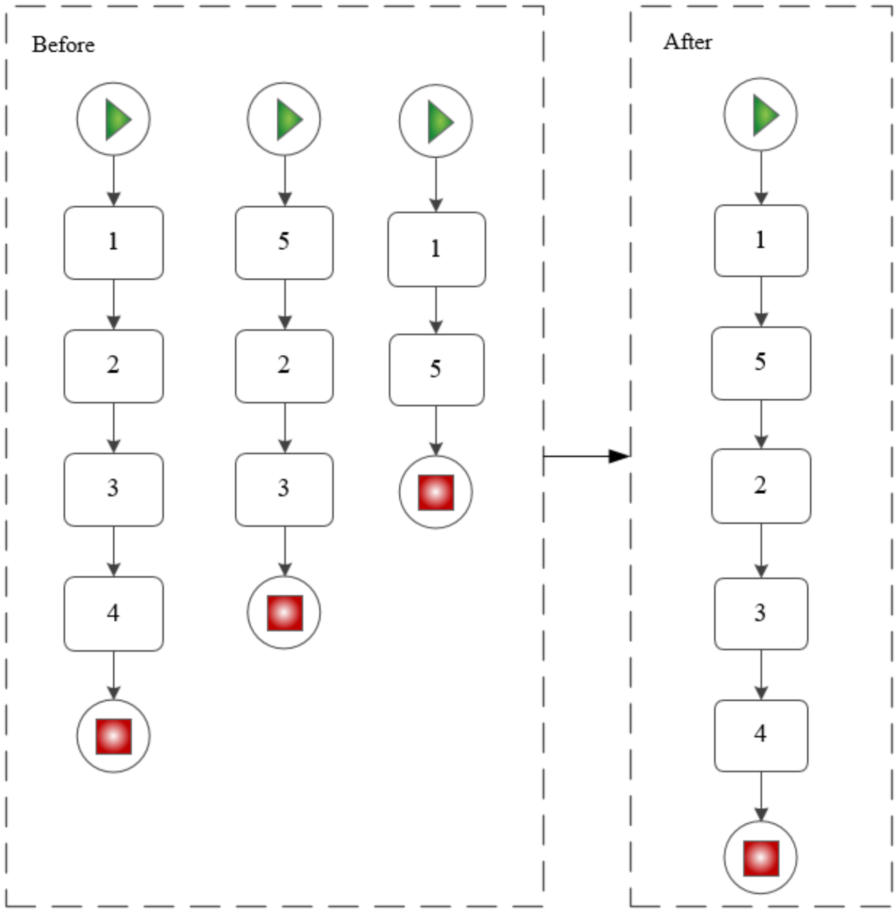


Figure 3.6 The generating process of a cognitive reasoning model

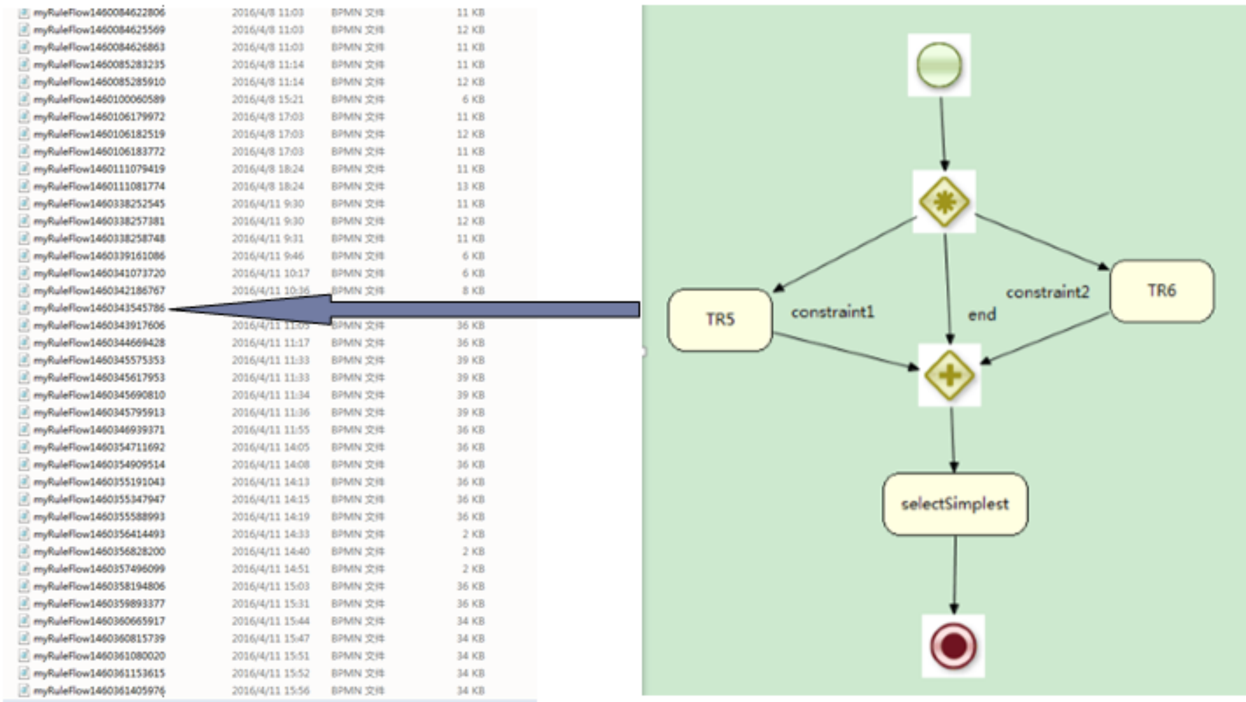


Figure 3.7 A cognitive reasoning model

Thus, by this cognitive reasoning model, all the results will be stored in the dynamic knowledge base. When matching is successful, it will generate readable processes, otherwise, it will return and continue computing and reasoning until it reaches one of the termination conditions.

4. Experiments

For RMCEE, we consider a simulated scenario to take part in the college entrance examination just like high school students. It is disconnected from the outside world (i.e. no network and no additional database). First, it is faced with the input of natural language texts, LaTeX mathematics formula and XML graphics information from the examination paper. Next, it will start to read the problems in order to automatically understand the natural language of the mathematics problems by the entity combination model. Finally, it will solve the math problems automatically by the hybrid reasoning model. The subject areas include elementary algebra, plane geometry, solid geometry, analytic geometry, set, functions, series, complex numbers.

4.1 An Analytical Geometry Example

In example 1 (Fig. 1.1), the solving processes of an analytical geometry problem can be generated to help understand the effective use of different strategies.

(1) Introducing variables: In order to express the logic relations of mathematical problems explicitly, new variables are introduced. For example, in sub-problem (1), step (36) $\therefore$ *by(6): the focal* length *of conic* C *is* $2 * C_3$. The variable "C_3" represents the parameter "c" in ellipse C, which has the relationship between the parameter "a" and the parameter "b" in the analytic

expression "$c^2 = a^2 - b^2$". In sub-problem (2), step (23) $\because S_{\triangle ABD} = v_0$, variable "$v_0$" is also introduced to express the area of $\triangle ABD$.

(2) Classification discussion: In sub-problem (2), step (42) $\therefore$ *by(18,19,20,21,27,28,31,32,33,34,35,37,40,41):* $s_M = -1; a = 2; b = 3^{\frac{1}{2}}; t_M = \frac{3}{2}$ *or* $s_M = -1; a = 2; b = 3^{\frac{1}{2}}; t_M = \frac{3}{2}$. The variable "$s_M$" appears in the multi-solutions, so it needs to be discussed in two classifications.

(3) Expression simplifications and (in)equation combinations: In sub-problem (2) step (26) $\therefore$ *by (9,10,12,13,15,17,22,23,24,25): the maximum value of* $S_{\triangle ABD}$ *is* $\frac{144}{49}$. The problem is solved efficiently by selecting previously valid equations or inequalities from the knowledge base.

The Human-like solving processes of example 1:
Question (1):
(1)∵ line F_1M⊥line X:y=0
(2)∴ by(1): analytic of function F_1M is x=x_F_1M
(3)∴ by(1): analytic of function X is y=0
(4)∴ by(1,2,3): x_F_1M=0
(5)∵ the focus of C is F_1.
(6)∵ analytic of ellipse C is ((x^2)/(a^2))+((y^2)/(b^2))=1
(7)∴ by(5): point F_1
(8)∴ by(5,6,7): point F_1(-(a^2-b^2)^(1/2),0)
(9)∴ by(3,8): point F_1 (-(a^2-b^2)^(1/2), 0) is on line X: y = 0
(10)∴ by(2): point F_1 is on line F_1M: x = x_F_1M
(11)∴ by(9,10): line X:y=0 and line F_1M:x=x_F_1M crossing at point F_1(-(a^2-b^2)^(1/2),0)
(12)∴ by(11): point F_1(-(a^2-b^2)^(1/2), 0) is on line F_1M:x=x_F_1M
(13)∴ by(4,8,12): analytic of function F_1M is x=-(a^2-b^2)^(1/2)
(14)∴ by(13): point M is on line F_1M:x=-(a^2-b^2)^(1/2)
(15)∵ point M is on ellipse C
(16)∴ by(15): point M
(17)∴ by(15,16): point M(s_M, t_M)
(18)∴ by(13,14,17): s_M+(a^2-b^2)^(1/2)=0
(19)∴ by(6): a>b
(20)∴ by(6,15,17): s_M≥-a
(21)∴ by(6,15,17): t_M≥-b
(22)∵ F_1M=(3/2)
(23)∴ by(22): |vector F_1M| is (3/2)
(24)∴ by(23): vector F_1M is (s_M+(a^2-b^2)^(1/2), t_M)
(25)∴ by(24): vector F_1M
(26)∴ by(8,17,25): vector F_1M=(s_M+(a^2-b^2)^(1/2), t_M)
(27)∴ by(23,26): (a^2-b^2+2*s_M*(a^2-b^2)^(1/2)+s_M^2+t_M^2)^(1/2)=3/2
(28)∴ by(6): a^2≠b^2
(29)∴ by(5,6): Point F_1 is on line X:y=0
(30)∴ by(5,29): The focus of C is on X axis.

(31)∴ by(6,30): a^2>b^2
(32)∴ by(6): a>0
(33)∴ by(6,15,17): t_M≤b
(34)∴ by(6,15,17): s_M^2/a^2+t_M^2/b^2-1=0
(35)∴ by(6): b>0
(36)∴ by(6): focal length of conic C is 2*C_3
(37)∴ by(6,36): C_3>0
(38)∴ by(4,14,17): analytic of function F_1M is x=s_M
(39)∴ by(38): point F_1 is on line F_1M:x=s_M
(40)∴ by(8,38,39): -(a^2-b^2)^(1/2)-s_M=0
(41)∴ by(6,15,17): s_M≤a
(42)∴ by(18,19,20,21,27,28,31,32,33,34,35,37,40,41): s_M=-1,a=2,b=3^(1/2),t_M=3/2 or s_M=-1,a=2,b=3^(1/2),t_M=-3/2
Discussions in different conditions:
Condition 1
when [${s}_{M}=(-1)$, $a=2$, $b=\sqrt{3}$, ${t}_{M}=\frac{3}{2}$]:
(1)∵ b=3^(1/2)
(2)∵ a=2
(3)∵analytic of ellipse C is ((x^2)/(a^2))+((y^2)/(b^2))=1
(4)∴ by(2,3): analytic of ellipse C is 1/4*(b^2*x^2+4*y^2)/b^2=1
(5)∴ by(1,4): analytic of ellipse C is 1/4*x^2+1/3*y^2=1
when [${s}_{M}=(-1)$, $a=2$, $b=\sqrt{3}$, ${t}_{M}=(-\frac{3}{2})$]:
Condition 2
The same as Condition 1
To sum up, [the standard equation of ellipse C is x^2/4+y^2/3=1]

Question (2):
(1)∵ the standard equation of ellipse C is x^2/4+y^2/3=1
(2)∵ the left vertex of ellipse C is point A
(3)∴ by (1,2): A(-2,0)
(4)∵ line AB intersects ellipse C at B
(5)∴ by (3,4): the equation of function AB is y=k_AB*(x+2)
(6)∵ line AD intersects ellipse C at D
(7)∴ by (3,6): the equation of function AD is y=k_AD*(x+2)
(8)∵ let B(x_B, y_B)
(9)∴ by (1,4,8): x_B^2/4+y_B^2/3=1
(10)∴ by (3,5,8): k_AB=(y_B-0)/(x_B+2)
(11)∵ let D(x_D, y_D)
(12)∴ by (1,6,11): x_D^2/4+y_D^2/3=1
(13)∴ by (3,7,11): k_AD=(y_D-0)/(x_D+2)
(14)∵ segment AB
(15)∴ by (3,8,14): AB=((x_B+2)^2+(y_B-0)^2)^(1/2)
(16)∵ segment AD
(17)∴ by (3,11,16): AD=((x_D+2)^2+(y_D-0)^2)^(1/2)
(18)∵ Rt△ABD(vertex is point A)
(19)∴ by (18): Rt∠BAD

(20)∴ by (19): AD⊥AB, foot point is A
(21)∴ by (20): segment AB is the height of △ABD
(22)∴ by (21): S_△ABD=((1/2)*AD)*AB
(23)∵ S_△ABD=v_0
(24)∴ by (15,17,22,23): v_0=1/2*((x_D+2)^2+(y_D-0)^2)^(1/2)*((x_B+2)^2+(y_B-0)^2)^(1/2)
(25)∴ by (20): k_AB*k_AD=-1
(26)∴ by (9,10,12,13,15,17,22,23,24,25): the maximum value of S_△ ABD is 144/49

4.2 An Example in the 2017 Math College Entrance Exams

Example 2: The Question Descriptions is shown in Figure 4.1 and Figure 4.2:

如图，在三棱锥P-ABC中，PA⊥AB，PA⊥BC，AB⊥BC，PA=AB=BC=2， D为线段AC的中点，E为线段PC上一点。(I)求证:PA⊥BD;(II)求证:平面BDE⊥平面PAC;(III) 当PA//平面BDE时，求三棱锥E-BCD的体积.

As is shown in figure: In tetrahedron P-ABC, PA⊥AB, PA⊥BC,AB⊥BC,PA=AB=BC=2, D is the midpoint of AC, and E is on segment PC. (1)Prove:PA⊥BD; (2)Prove:plane BDE⊥plane PAC; (3)When PA // plane BDE, calculate the volume of tetrahedron E-BCD.

Figure 4.1 The question descriptions

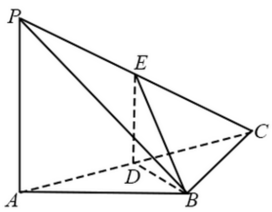


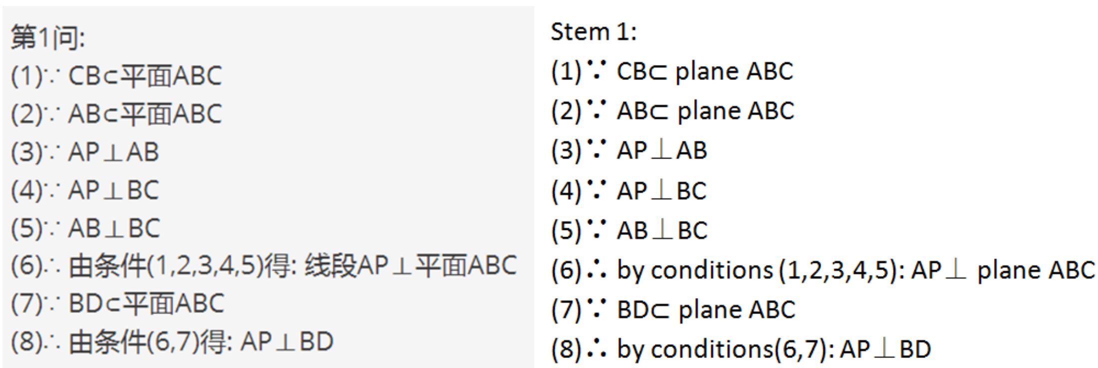
Figure 4.2 The figure of the question

The result of Natural Language Processing is shown in figure 4.3.

```
public stem:
PyramidRelation:Pyramid:3 pyramid P-ABC
LinePerpRelation{line1=PA, line2=AB, crossPoint=A}
LinePerpRelation{line1=PA, line2=BC, crossPoint=}
LinePerpRelation{line1=AB, line2=BC, crossPoint=B}
MultiEqualityRelation [multiExpressCompare=AP=AB=BC=2, originExpressRelationList=[], keyWord=null, result=null]
MiddlePointOfSegmentRelation{middlePoint=D,segment=AC}
PointOnLineRelation{point=E, line=PC, isConstant=false, extension=false}
stem conditions:
stem 1 : []
stem 2 : []
stem 3 : [EqualityRelation{V_EBCD=v_0}, LineParallelPlaneRelation [line=PA, plane=Plane:BDE, crossPoints=NONE]]
stem conclusion:
stem 1 : [ProveConclusionRelation:[LinePerpRelation{line1=PA, line2=BD, crossPoint=}]]
stem 2 : [ProveConclusionRelation:[DualPlanePerpRelation [planeA=Plane:BDE, planeB=Plane:PAC, crossLine=null]]]
stem 3 : [ComputeValueConclusionRelation:[[GeoCalculateExpress{closedShape=Pyramid:3pyramid E-BCD, calculateType=Volume, express=V_EBCD}] REAL]target:VALUE]
```

Figure 4.3 The result of natural language processing

The readable problem solving processes of Automated Solving is shown in figure 4.4.



第1问:	Stem 1:
(1)∵ CB⊂平面ABC	(1)∵ CB⊂ plane ABC
(2)∵ AB⊂平面ABC	(2)∵ AB⊂ plane ABC
(3)∵ AP⊥AB	(3)∵ AP⊥AB
(4)∵ AP⊥BC	(4)∵ AP⊥BC
(5)∵ AB⊥BC	(5)∵ AB⊥BC
(6)∴ 由条件(1,2,3,4,5)得: 线段AP⊥平面ABC	(6)∴ by conditions (1,2,3,4,5): AP⊥ plane ABC
(7)∵ BD⊂平面ABC	(7)∵ BD⊂ plane ABC
(8)∴ 由条件(6,7)得: AP⊥BD	(8)∴ by conditions(6,7): AP⊥BD

第2问:
(1)∵ CA⊂平面ACP
(2)∵ PA⊂平面ACP
(3)∵ CB⊂平面ABC
(4)∵ AB⊂平面ABC
(5)∵ AP⊥AB
(6)∵ AP⊥BC
(7)∵ AB⊥BC
(8)∴ 由条件(3,4,5,6,7)得: 线段AP⊥平面ABC
(9)∵ BD⊂平面ABC
(10)∴ 由条件(8,9)得: AP⊥BD
(11)∵ 等腰△ABC
(12)∵ 点D是线段AC的中点
(13)∵ △ABC
(14)∴ 由条件(12,13)得: 线段BD是△ABC的中线
(15)∴ 由条件(11,14)得: 线段BD是三角形△ABC的高
(16)∴ 由条件(15)得: BD⊥AC
(17)∴ 由条件(1,2,10,16)得: 线段BD⊥平面ACP
(18)∵ BD⊂平面BDE
(19)∴ 由条件(17,18)得: 平面ACP⊥平面BDE

Stem 2:
(1)∵ CA⊂ plane ACP
(2)∵ PA⊂ plane ACP
(3)∵ CB⊂ plane ABC
(4)∵ AB⊂ plane ABC
(5)∵ AP⊥AB
(6)∵ AP⊥BC
(7)∵ AB⊥BC
(8)∴ by conditions (3,4,5,6,7): segment AP⊥ plane ABC
(9)∵ BD⊂ plane ABC
(10)∴ by conditions (8,9): AP⊥BD
(11)∵ isosceles △ABC
(12)∵ point D is the midpoint of AC
(13)∵ △ABC
(14)∴ by conditions (12,13): BD is the median of △ABC
(15)∴ by conditions (11,14): BD is the height of △ABC
(16)∴ by conditions (15): BD⊥AC
(17)∴ by conditions (1,2,10,16): segment BD⊥ plane ACP
(18)∵ BD⊂ plane BDE
(19)∴ by conditions (17,18): plane ACP⊥ plane BDE

第3问:
(1)∵ 棱锥E-BCD
(2)∵ CB⊂平面ABC
(3)∵ AB⊂平面ABC
(4)∵ AP⊥AB
(5)∵ AP⊥BC
(6)∵ AB⊥BC
(7)∴ 由条件(2,3,4,5,6)得: 线段AP⊥平面ABC
(8)∵ 线段AP∥平面BDE
(9)∵ PA⊂平面ACP
(10)∵ △BDE
(11)∴ 由条件(10)得: 平面BDE
(12)∵ △ACP
(13)∴ 由条件(12)得: 平面ACP
(14)∴ 由条件(11,13)得: 平面BDE∩平面ACP于DE
(15)∴ 由条件(8,9,14)得: AP∥DE
(16)∴ 由条件(7,15)得: 线段DE⊥平面ABC
(17)∴ 由条件(1,16)得: 线段DE是EBCD的高
(18)∴ 由条件(1,17)得: V_EBCD=1/3*S_△BCD*DE
(19)∵ △BCD
(20)∵ ∠CBD
(21)∴ 由条件(19,20)得: S_△BCD=(((BC)*(BD))*(sin(∠CBD)))*(1/2)
(22)∵ 等腰直角△BCD
(23)∴ 由条件(22)得: ∠CBD=1/4*π
(24)∵ BC=2
(25)∵ 点D是线段AC的中点
(26)∵ △ABC
(27)∴ 由条件(25,26)得: 线段BD是△ABC的中线
(28)∵ Rt△ABC
(29)∴ 由条件(27,28)得: BD=(1/2)*(AC)
(30)∴ 由条件(28)得: AC=2*2^(1/2)
(31)∴ 由条件(29,30)得: BD=2^(1/2)
(32)∴ 由条件(21,23,24,31)得: S_△BCD=1
(33)∴ 由条件(18,32)得: V_EBCD-1/3*DE=0
(34)∵ △CDE
(35)∴ 由条件(34)得: DE=1
(36)∴ 由条件(33,35)得: V_EBCD=1/3

Stem 3:
(1)∵ pyramid E-BCD
(2)∵ CB⊂ plane ABC
(3)∵ AB⊂ plane ABC
(4)∵ AP⊥AB
(5)∵ AP⊥BC
(6)∵ AB⊥BC
(7)∴ by conditions (2,3,4,5,6): segment AP⊥ plane ABC
(8)∵ segment AP∥ plane BDE
(9)∵ PA⊂ plane ACP
(10)∵ △BDE
(11)∴ by conditions (10): plane BDE
(12)∵ △ACP
(13)∴ by conditions (12): plane ACP
(14)∴ by conditions (11,13): plane BDE∩ plane ACP=DE
(15)∴ by conditions (8,9,14): AP∥DE
(16)∴ by conditions (7,15): segment DE⊥ plane ABC
(17)∴ by conditions (1,16): segment DE is the height of EBCD
(18)∴ by conditions (1,17): V_EBCD=1/3*S_△BCD*DE
(19)∵ △BCD
(20)∵ ∠CBD
(21)∴ by conditions (19,20): S_△BCD=(((BC)*(BD))*(sin(∠CBD)))*(1/2)
(22)∵ isosceles right △BCD
(23)∴ by conditions (22): ∠CBD=1/4*π
(24)∵ BC=2
(25)∵ point D is the midpoint of AC
(26)∵ △ABC
(27)∴ by conditions (25,26): segment BD is the median of △ABC
(28)∵ Rt△ABC
(29)∴ by conditions (27,28): BD=(1/2)*(AC)
(30)∴ by conditions (28): AC=2*2^(1/2)
(31)∴ by conditions (29,30): BD=2^(1/2)
(32)∴ by conditions (21,23,24,31): S_△BCD=1
(33)∴ by conditions (18,32): V_EBCD-1/3*DE=0
(34)∵ △CDE
(35)∴ by conditions (34): DE=1
(36)∴ by conditions (33,35): V_EBCD=1/3

Figure 4.4 The readable problem solving processes of automated solving

4.2 Result of Exams

Training set: About 500 sets of college entrance exams or simulated college entrance exams. A set of exams is mainly composed of standard problems (such as multiple-choice problems, fill-in-the-blanks problems) and non-standard problems (such as Calculation problems, Proof problems).

Test set:

1) For the Liberal arts mathematics college entrance exams in 2017 in the Beijing area, RMCEE earned a score of 105 out of 150, and took 22 minutes to complete.

2) For secret exams simulated college entrance exams from IFLYTEK (see [34]), RMCEE scored 123 out of 150. Furthermore, in 8 sets of Liberal Arts mathematics exams selected as a training set, RMCEE scored an average of 118 out of 150, with a high score of 127 points, and a low score of 106 points. The scores of all the 8 exams are shown in table 5.

Table 5 The scores of 8 set of exams

problems / exams	multiple-choice problems (8*5)	fill-in-the-blanks problems (5*5)	calculation or proof problems (13+13+13+13+14+14)	total points (150)
BJ2013	8*5	4*5	13+0+13+13+7+0	106
BJ2014	7*5	4*5	13+13+8+6+14+14	123
BJ2015	6*5	5*5	13+13+0+14+13+14	122
BJ2016	8*5	5*5	13+13+0+13+6+8	118
BJ2017	8*5	4*5	13+9+13+4+14+8	121
SD2017	9*5	4*5	5+12+7+10+6+6	111
QG22017	12*5	4*5	0+12+0+12+12+0	116
BJ20170722	7*5	5*5	13+13+6+14+13+8	127

5. Conclusion and Discussion

RMCEE contains a series of contributions and innovations, which are discussed here.

First, an elementary mathematics knowledge graph covering nearly all the knowledge points in current textbooks is constructed. It includes about 800 entities and relations, 1700 rules and 300 cognitive models. Therefore, it can provide semantic support for natural language comprehension and automatic problem solving.

Second, a math relation comprehension model is presented. It integrates math relation templates with entity combinations based on a knowledge graph. Consequently, it can effectively solve the natural language understanding of math problems.

Third, a hybrid cognitive reasoning model is presented. It integrates logical reasoning, symbolic computation, and deep learning. Consequently, it can generate readable problem solving processes.

RMCEE is implemented based on the above innovations. It can solve single-choice problems, fill-in-the-blanks problems and report the final results directly. It can also solve calculation problems, proof problems, with human-like readable solving processes generated step by step.

RMCEE attended the college entrance exam in China on June 7th, 2017, which scored 105 points out of 150 in just 22 minutes, while the average score of students was 105.15 points out of 150, but they took 120 minutes.

For further research, it is interesting to consider natural language understanding and problem solving of elementary mathematics problems motivated from real life and commonsense knowledge (see [35], [36] and [37]).

Acknowledgements The authors would like to thank Chengdu ZhunXing YunXue Technology Corporation, Philip Hamish Todd of Saltire Software, Inc, Shengchuan Wu of Franz Corporation, Liang Xu, Xinchao Wu, Siwen Jiang and other students in our lab. The authors also wish to thank the anonymous reviewers for their helpful comments. This work was funded by the National Key R&D Program of China (No. 2018YFB1005100 & No. 2018YFB1005104).

References

[1] https://en.wikipedia.org/wiki/Rene_Descartes

[2] https://www.nytimes.com/2011/02/17/science/17jeopardy-watson.html

[3] Takuya Matsuzaki, Hidenao Iwane,Hirokazu Anai, and Noriko H Arai. The complexity of math problems-linguistic, or computational? In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 73–81, 2013.

[4] Takuya Matsuzaki, Hidenao Iwane,Hirokazu Anai, and Noriko H Arai. The most uncreative examinee: a first step toward wide coverage natural language math problem solving. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[5] www.maplesoft.com

[6] Noriko H Arai, Takuya Matsuzaki, Hidenao Iwane, and Hirokazu Anai. Mathematics by machine. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, pages 1–8. ACM, 2014.

[7] https://news.cs.washington.edu/2015/09/21/ai-breakthrough-by-allen-institute-and-uw-geos-system-matches-student-performance-on-11th-grade-sat-geometry-problems

[8] Akira Fujita, Akihiro Kameda, Ai Kawazoe, and Yusuke Miyao. Overview of todai robot project and evaluation framework of its nlp-based problem solving. *World History*, 36:36, 2014.

[9] Xiaocheng Feng, Ming Liu, Jiahao Liu, Bing Qin, Yibo Sun, and Ting Liu. Topic-to-essay generation with neural networks. In *IJCAI*, pages 4078–4084, 2018.

[10] Gong Cheng, Weixi Zhu, Ziwei Wang, Jianghui Chen, and Yuzhong Qu. Taking up the gaokao challenge: An information retrieval approach. In *IJCAI*, pages 2479–2485, 2016.

[11] Alfred Tarski. *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley, 1951.

[12] Wen-Tsun Wu. On the decision problem and the mechanization of theorem-proving in elementary geometry. *Science in China Ser A*, 29(2):117–138, 1978.

[13] Shang-Ching Chou, Xiao-Shan Gao, and Jingzhong Zhang. *Machine proofs in geometry: Automated production of readable proofs for geometry theorems*, volume 6. World Scientific, 1994.
[14] Dongming Wang. Elimination procedures for mechanical theorem proving in geometry. *Annals of Mathematics and Artificial Intelligence*, 13(1-2):1–24, 1995.
[15] Deepak Kapur. Using grobner bases to reason about geometry problems. *Journal of Symbolic Computation*, 2(4):399–408, 1986.
[16] Shang-Ching Chou, Xiao-Shan Gao, and Jing-Zhong Zhang. A deductive database approach to auto-mated geometry theorem proving and discovering. *Journal of Automated Reasoning*, 25(3):219–246, 2000.
[17] Helder Coelho and Luis Moniz Pereira. Automated reasoning in geometry theorem proving with prolog. *Journal of Automated Reasoning*, 2(4):329–390, 1986.
[18] Noboru Matsuda and Kurt Vanlehn. Gramy: A geometry theorem prover capable of construction. *Journal of Automated Reasoning*, 32(1):3–33, 2004.
[19] Hongguang Fu, Xiuqin Zhong, and Zhenbing Zeng. Automated and readable simplification of trigonometric expressions. *Mathematical and computer modelling*, 44(11-12):1169–1177, 2006.
[20] Nate Kushman, Yoav Artzi, Luke Zettlemoyer, and Regina Barzilay. Learning to automatically solve algebra word problems. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 271–281, 2014.
[21] Minjoon Seo, Hannaneh Hajishirzi, AliFarhadi, Oren Etzioni, and Clint Malcolm. Solving geometry problems: Combining text and diagram interpretation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1476,2015.
[22] Peter Clark. Elementary school science and math tests as a driver for ai: take the aristo challenge! In *Twenty-Seventh IAAI Conference*, 2015.
[23] Mingzhe Wang, Yihe Tang, Jian Wang, and Jia Deng. Premise selection for theorem proving by deep graph embedding. In *Advances in Neural Information Processing Systems*, pages 2786–2796, 2017.
[24] Hongguang Fu, Xiuqin Zhong, and QunanLi. Geometry knowledge base learning from theorem proofs. *Knowledge Engineering and Management*, pages 21–33, 2014.
[25] https://neo4j.com
[26] https://wordforest.com
[27] https://wordnet.princeton.edu/
[28] http://ir.hit.edu.cn
[29] Han He. HanLP: Han Language Processing. https://github.com/hankcs/HanLP. 2014.
[30] Peng Zhou, Wei Shi, Jun Tian, ZhenyuQi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, 2016.
[31] http://www.openkg.cn
[32] https://www.drools.org
[33] Martti T. Tammi. The principles of shotgun sequencing and automated fragment assembly. *Center for Genomics and Bioinformatics*, pages 1–44, 2003.

[34] https://www.iflytek.com/
[35] Mohammad Javad Hosseini, Hannaneh Hajishirzi, Oren Etzioni, and Nate Kushman. Learning to solve arithmetic word problems with verb categorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, page523–533, 2014.
[36] Subhro Roy and Dan Roth. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, page 1743–1752, 2015.
[37] Lei Wang, Dongxiang Zhang, Lianli Gao, Jingkuan Song, Long Guo, and Heng Tao Shen. Mathdqn: Solving arithmetic word problems via deep re-inforcement learning. In *Thirty-Second AAAI Conferenceon Artificial Intelligence*, 2018.