Solving Natural Language Math Problems

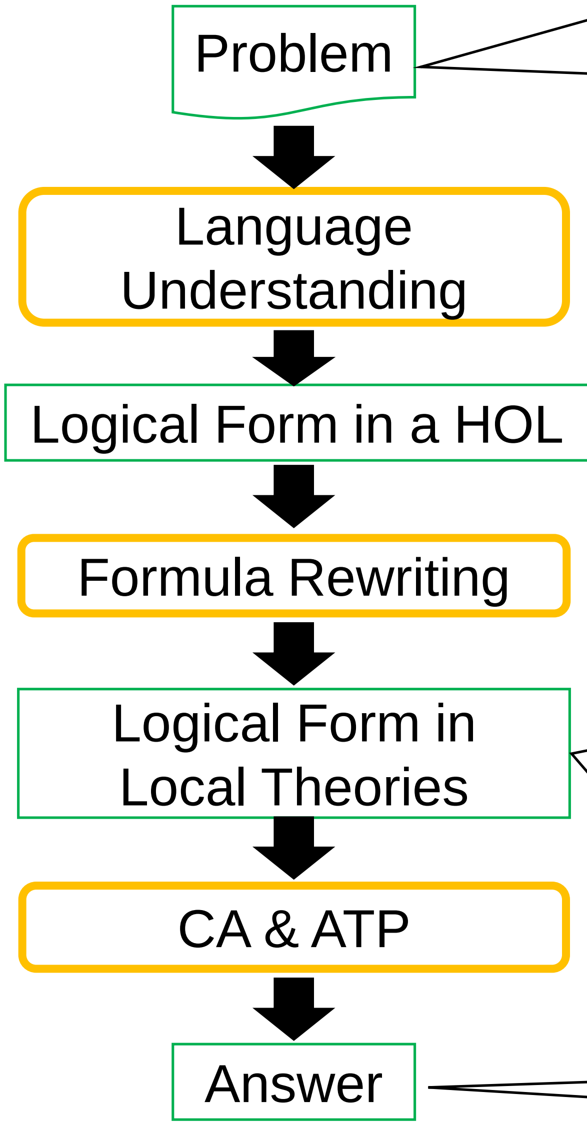
Takuya Matsuzaki
(Nagoya University)

Noriko H. Arai
(National Institute of Informatics)

Solving NL Math – why?

- It is the first and the last goal of symbolic approach to language understanding (LU)
  - Formalization of the domain is the prerequisite for LU
- Problem solving is the only way to compare different LU systems
  - Only the input and output are observable
  - No ground-truth for a mid-layer's output

System Overview



Let l be the trajectory of $(t+2, t+2, t)$ for t ranging over $\mathbb{R}$. $O(0,0,0)$, $A(2,1,0)$, and $B(1,2,0)$ are on a sphere, S, centered at $C(a,b,c)$.

Determine the condition on a, b, c for which S intersects with l.

$$\begin{aligned}
&\exists l \exists u \exists v \exists S \exists R(\\
&\quad l = \mathrm{line}(u,v) \ \wedge \ \forall p(p \in l \leftrightarrow \\
&\quad \exists t(p = (t+2, t+2, t))) \wedge \\
&\quad S = \mathrm{sphere}((a,b,c), R) \wedge (0,0,0) \in S \wedge \\
&\quad (2,1,0) \in S \wedge (1,2,0) \in S \wedge \\
&\quad \exists q(\mathrm{intersect}(S, l, q)))
\end{aligned}$$

$$\begin{aligned}
&\exists u_x \exists u_y \exists u_z(((\neg(u_x = 0)) \vee (\neg(u_y = 0)) \vee (\neg(u_z = 0))) \wedge (\exists v_x \exists v_y \exists v_z((\exists R((\exists t((tu_x + v_x - a)^2 + \\
&(tu_y + v_y - b)^2 + (tu_z + v_z - c)^2 = R^2)) \wedge (0 < R) \wedge (a^2 + b^2 + c^2 = R^2) \wedge ((1-a)^2 + (2-b)^2 + \\
&\wedge((2-a)^2 + (1-b)^2 + c^2 = R^2))) \wedge (\exists v_{lx} \exists v_{ly}((\exists v_{lz}((0 = u_y(v_z - v_{lz}) - u_z(v_y - v_{ly})) \\
&\wedge(0 = u_z(v_x - v_{lx}) - u_x(v_z - v_{lz})))) \wedge (0 = u_x(v_y - v_{ly}) - u_y(v_x - v_{lx})))) \wedge (\forall p_x \forall p_y \forall p_z \\
&(((p_x = p_z + 2) \wedge (p_y = p_z + 2)) \vee (\forall t_2((\neg(p_x = t_2 u_x + v_x)) \vee (\neg(p_y = t_2 u_y + v_y)) \\
&\vee(\neg(p_z = t_2 u_z + v_z)))))) \wedge (\forall p_t(\exists t_3((p_t = t_3 u_z + v_z) \wedge (p_t + 2 = t_3 u_x + v_x) \wedge (p_t + 2 = t_3 u_y + v_y \\
&)) \wedge (\exists u_{lx} \exists u_{ly}((\exists u_{lz}(((\neg(u_{lx} = 0)) \vee (\neg(u_{ly} = 0)) \vee (\neg(u_{lz} = 0))) \\
&\wedge(0 = u_y u_{lz} - u_z u_{ly}) \wedge (0 = u_z u_{lx} - u_x u_{lz}))) \wedge (0 = u_x u_{ly} - u_y u_{lx}))))
\end{aligned}$$

$$6a = 5 \wedge 6b = 5 \wedge (3c \leq 1 \vee 13 \leq 3c)$$

Today’s Topics

- Parsing Math Problem Text with Combinatory Categorial Grammar
- Benchmarking a CAS-based solver with formalized pre-university math problems

Combinatory Categorial Grammar

- Word ⇔ (syntactic category, λ-expression)

Word type	Example
Proper noun	"John" ⇔ (NP, `john`)
Common noun	"cat" ⇔ (N, `λx.cat(x)`)
Intransitive verb	"runs" ⇔ ($S \setminus NP$, `λx.run(x)`)
Transitive verb	"loves" ⇔ ($S \setminus NP/NP$, `λy.λx.love(x,y)`)
Indefinite article	"a" ⇔ ($S/(S \setminus NP)/N$, `λN.λP.∃x(Nx∧Px)`)
Quantifier	"every" ⇔ ($S/(S \setminus NP)/N$, `λN.λP.∀x(Nx→Px)`)

Combinatory rules

Forward application

$$> \frac{X / Y : f \quad Y : y}{X : f\,y}$$

Backward application

$$< \frac{Y : y \quad X \backslash Y : f}{X : f\,y}$$

Forward composition

$$>B \frac{X / Y : f \quad Y / Z : g}{X / Z : \lambda z.f\,(gz)}$$

etc.

John — NP : john

loves — $S \backslash NP / NP$: $\lambda x.\lambda y.\text{love}(y,x)$

a — $S \backslash NP \backslash (S \backslash NP / NP) / N$: $\lambda N.\lambda P.\lambda y.\exists x(Nx \wedge Pxy)$

cat — N : $\lambda x.\text{cat}(x)$

$>$ $S \backslash NP \backslash (S \backslash NP / NP)$: $\lambda P.\lambda y.\exists x(\text{cat}(x) \wedge Pxy)$

$<$ $S \backslash NP$: $\lambda y.\exists x(\text{cat}(x) \wedge \text{love}(y,x))$

$<$ S : $\exists x(\text{cat}(x) \wedge \text{love}(\text{john},x))$

Combinatory rules

Forward application

$$> \frac{X / Y : f \quad Y : y}{X : f\,y}$$

Backward application

$$< \frac{Y : y \quad X \backslash Y : f}{X : f\,y}$$

Forward composition

$$>B \frac{X / Y : f \quad Y / Z : g}{X / Z : \lambda z.f\,(gz)}$$

etc.

a: $S \backslash NP \backslash (S \backslash NP/NP)/N$: $\lambda N.\lambda P.\lambda y.\exists x(Nx \wedge Pxy)$

cat: N : $\lambda x.cat(x)$

$>$ a cat: $S \backslash NP \backslash (S \backslash NP/NP)$: $\lambda P.\lambda y.\exists x(cat(x) \wedge Pxy)$

loves: $S \backslash NP/NP$: $\lambda x.\lambda y.love(y,x)$

$<$ loves a cat: $S \backslash NP$: $\lambda y.\exists x(cat(x) \wedge love(y,x))$

John: NP : john

$<$ John loves a cat: S : $\exists x(cat(x) \wedge love(john,x))$

Syntactic Category = Semantic Type + Syntactic Constraints

Example

“distance” (as in “distance between *P* and *Q*”)

- Syntactic cat.: $NP_{\texttt{Real}}/PP_{between,\texttt{(Point,Point)}}$
- Semantic function: `λp.dist(p)`
- Semantic type: `(Point, Point)` → `Real`

P: NP_{Pnt} : P

and: $NP_{(\alpha,\beta)} \backslash NP_{\alpha} / NP_{\beta}$: `λy.λx.(x,y)`

Q: NP_{Pnt} : Q

$NP_{(Pnt,Pnt)}$: `(P,Q)`

between: $PP_{btwn,(\alpha,\beta)}/NP_{(\alpha,\beta)}$: `id`

$PP_{btwn,(Pnt,Pnt)}$: `(P,Q)`

distance: $NP_{Real}/PP_{btwn,(Pnt,Pnt)}$: `λp.dist(p)`

NP_{Real} : `dist(P,Q)`

Comparison with compilers

- Compilers : source code → machine code
- NL parsing : math problem → logical form

- NL parsing = type check
+ syntax check
+ denotational semantics

- Besides, the grammar is only partially known and ambiguous

Grammar and lexicon: current status

- Size
  - 31 combinatory rules
  - 6,652 different word forms
  - 42,154 triples of <word, category, λ-term>

- What's not in textbook (toy) grammars:
  - Imperatives, pluralities, relation/attribute nouns, context dependent semantics, action verbs, etc.

- Coverage:
  - 70%~80% of university math exam sentences can be parsed (either correctly or wrongly)

Remaining issues

- Lexicon / grammar coverage

- Hypothesis explosion due to local ambiguity
  - "$y = ax^2$": equality or $\lambda x.ax^2$ or $\{ (x,y) \mid y = ax^2 \}$
  - "if A then B and C": $(A \rightarrow B) \& C$ or $A \rightarrow (B \& C)$

- Inter-sentential logical structure analysis. E.g.,
  - Sentence 1: If A then B.
  - Sentence 2: If C then D.

  - $(A \rightarrow B) \& (C \rightarrow D)$
  - $A \rightarrow (B \& (C \rightarrow D))$
  - $(A \rightarrow B) \& (A \rightarrow (B \& C) \rightarrow D)$

Benchmarking CA-based Problem Solver on Formalized Pre-univ. Math Problems

Motivation

- Development of the AR layer of the solver in parallel with the NLU layer
- Evaluation on problems with varying difficulty
- Estimation of the computational cost of the reasoning on NLU output

Benchmark Problems: Sources

- **Ex**: 288 problems from exercise book series
  - 200 problems on geometry
  - 100 problems on integer arithmetic
- **Univ**: 245 problems from the entrance exams of seven national universities
  - Geometry, real arithmetic, pre-calculus etc. expressible in the theory of RCF
- **IMO**: 212 problems from the International Mathematics Olymipiads (1959-2014)
  - All geometry and real arithmetic problems
  - Some of number theory, combinatorics etc.
  - 2/3 of the all past problems till 2014

Encoding process

- Six students (majored in math/CS) and two full-time researchers encoded the problems in a higher-order language
- Literal translation
  - Word-by-word, sentence-by-sentence
  - No inference
  - No paraphrase

Example

Let D be a point inside acute triangle ABC such that

$\angle ADB = \angle ACB + \pi/2$

and

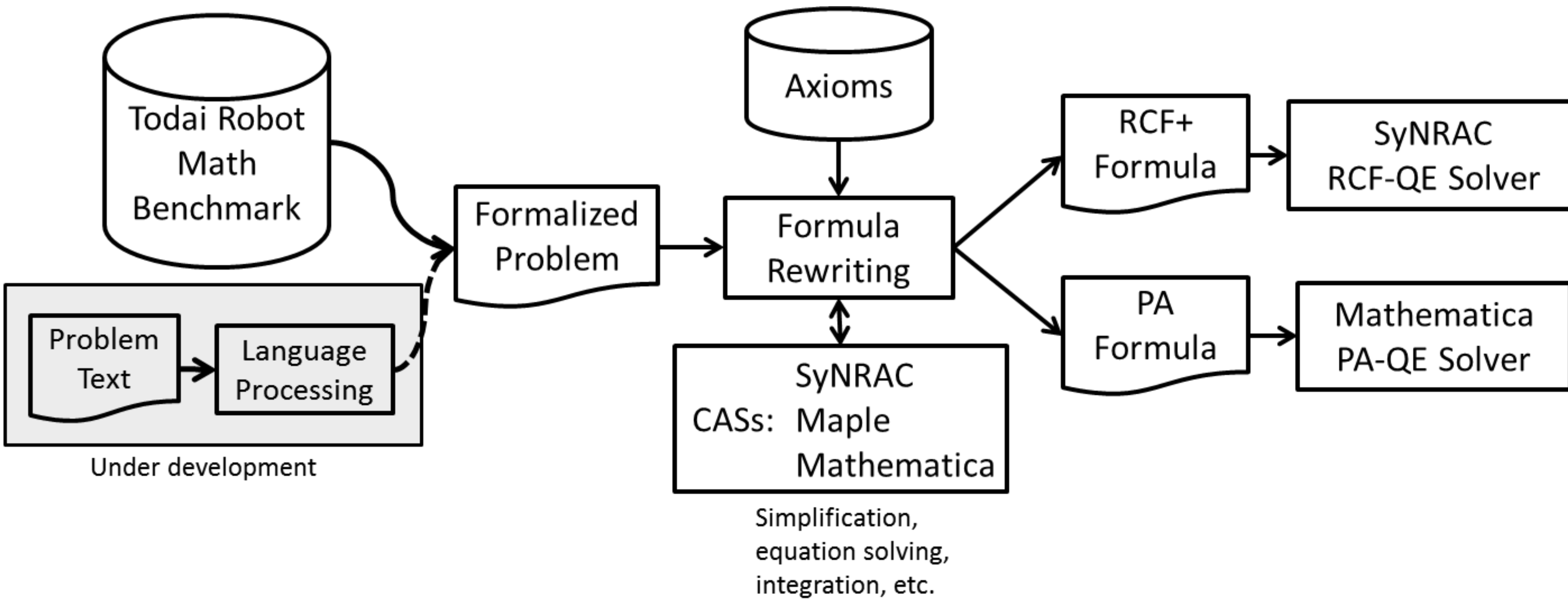
$AC \cdot BD = AD \cdot BC$

Calculate the ratio $(AB \cdot CD)/(AC \cdot BD)$.

(IMO 1993 Problem 2)

```
(Find (x)
  (exists (A B C D)
    (&& (is-acute-triangle A B C)
        (point-inside-of D (triangle A B C))
        (= (rad-of-angle (angle A D B))
           (+ (rad-of-angle (angle A C B)) (/ (Pi) 2)))
        (= (* (distance A C) (distance B D))
           (* (distance A D) (distance B D)))
        (= x (/ (* (distance A B) (distance C D))
                (* (distance A C) (distance B D))))))))
```

CAS-based solver



Syntactic Profile (per problem; medians)

	Pre-univ math benchmark				TPTP-THF
	Ex	Univ	[illegible]	[illegible]	
# Formulas	2	2	[illegible]	1	10
# Atoms	65	95	65	72	88
Avg atoms/Fml	38	54	56	48	6
# Symbols	16	19	[illegible]	[illegible]	9
# Variables	9	[illegible]	[illegible]	[illegible]	19
λ	3	3	1	2	2
∀	0	0	4	0	9
∃	4	6	1	4	2
# Connectives	55	78	58	61	52

Problem scale is at similar level

#of λ abstractions

Different types quantifications

Overall results

		Succeeded		Failed		
		Success %	Time (sec) Min/Med/Avg/Max	Timeout	Wrong	Other
Ex	RCF	63.8% (111/174)	13/18.0/ 37.4/ 343	10.9%	1.7%	23.6%
	PA	57.1% (48/ 84)	12/17.0/ 20.3/ 172	0.0%	0.0%	42.9%
	Other	10.0% (3/ 30)	13/14.0/ 17.7/ 26	0.0%	0.0%	90.0%
	All	56.3% (162/288)	12/17.0/ 32.0/ 343	6.6%	1.0%	36.1%
Univ	All (RCF only)	58.0% (142/245)	12/26.5/ 85.5/1417	15.5%	2.9%	23.7%
IMO	RCF	16.5% (19/115)	14/25.0/ 51.8/ 197	29.6%	0.9%	53.0%
	PA	4.8% (2/ 42)	25/29.5/ 29.5/ 34	16.7%	0.0%	78.6%
	Other	3.6% (2/ 55)	17/24.5/ 24.5/ 32	12.7%	0.0%	83.6%
	All	10.8% (23/212)	14/25.0/ 47.5/ 197	22.6%	0.5%	66.0%

- Difficulty of RCF problmes: Ex < Univ < IMO
- Difficulty of PA problems: Ex << IMO

Results on RCF problems in Ex

# of Stars	Succeeded		Failed		
	Success %	Time (sec) Min/Med/Avg/Max	Timeout	Wrong	Other
1	82.4% (28/34)	13/17.0/20.4/ 65	2.9%	0.0%	14.7%
2	79.4% (27/34)	16/18.0/28.1/230	2.9%	2.9%	14.7%
3	57.6% (19/33)	15/17.0/36.1/341	6.1%	0.0%	36.4%
4	47.4% (18/38)	15/19.0/62.1/343	23.7%	2.6%	26.3%
5	54.3% (19/35)	16/28.0/53.6/279	17.1%	2.9%	25.7%

- # of Stars = difficulty level assessed by the editors of the practice book series

Results on IMO problems by years

Years	Human Efficency	Machine Efficiency	Succeeded	Failed		
				Timeout	Wrong	Other
1959-69	58.23%	21.11%	26.3% (15/57)	22.8%	1.8%	49.1%
1970-79	46.57%	7.00%	13.3% (4/30)	26.7%	0.0%	60.0%
1980-89	44.35%	1.85%	3.1% (1/32)	31.2%	0.0%	65.6%
1990-99	38.27%	3.33%	5.7% (2/35)	11.4%	0.0%	82.9%
2000-13	34.31%	1.19%	1.9% (1/54)	22.2%	0.0%	75.9%

- Human Efficiency: IMO participants' avg. score
- Machine Efficiency: system's score

- IMO problems get harder by year both for human and machines

Summary

- Natural Language Math Solving System combining
  - Grammar-driven semantic analysis
  - Inference by QE

- Benchmark result on the inference part
  - Excercise & entrance exam: ~60%
  - Mathematical Olympiads: 5~15%