Interpreter for Geometric Constructions

10479, Pankaj Prateek

10314, Jeetesh Mangwani

|  |
| --- |
| Objective |
| To design and implement interpreter that   * is language-independent (works for English, Hindi at present) * Receives steps for a geometric construction as input e.g. “Draw a line segment AB of length 4 cm”, “केंद्र B और त्रिज्या 5 सेमी लेकर एक चाप खींचिए जो पहले खींची चाप को C पर काटता हो” etc * Outputs the geometric figure obtained on executing the given sequence of steps |
| Abstract |
| In this project, we take up the problem of designing and implementing a language-independent interpreter for drawing geometric diagrams, focusing mainly on ruler and compass based construction problems. We start with the use cases and motivations on why such a system would be useful and what places will deploying it would be fruitful. We give a glimpse of related research work done on geometry related problems and point out the common trait that they lack the ability to decipher any problem/constraint expressed in a natural language. We, then, move to describing the design of the interpreter and the usage of GIZA++ to provide language-independent interpretation ability. We briefly mention how is the alignment model exploited to realize the powerful translation feature. This is followed by accounts about the work done during this semester, results obtained and plan for the next semester. |
| Introduction |
| It is often the case that students, teachers, architects and artists need to draw complex diagrams manually using simple geometric instruments like ruler, compass, set squares, dividers etc. This demands labor, time as well as expertise. Resorting to sophisticated graphics applications requires knowhow of application-specific details as well as expertise in coordinating hand micro-movements.  In order to save these resources required in drawing geometric diagrams as well as to reduce dependence on complex graphics applications, this project introduces an interpreter for diagram construction steps expressed in a suitable natural language. This not only simplifies the diagram constructions, but also leads to easily understood natural language 'programs'. |
| Related Works |
| We brief about work done related to this field. We observe a common trait in the first three papers that they attempt to solve geometry problems in the literal sense. At the same time, they don't talk about problem statements/constraints expressed in some natural language. The last two papers are closely relate to what has been used in this project.   1. **Geometric Construction Problem Solving in Computer-Aided Learning**   This paper introduces CAD methods to deal with Constraint Satisfaction Problems related to geometry, at the same time claims to have met the requirement met by the educational domain. The methods show that results must be construction programs to take into account particular cases. A knowledge-based system implemented in Prolog has also been presented.   1. **Synthesizing geometry constructions** [citation]   Aiming to solve ruler & compass based geometry construction problems, the tool described in this paper claims to have synthesized construction steps for problems posed from high-school textbooks and examination papers. It focuses on three important insights viz. reduction of symbolic reasoning to concrete reasoning using verification based on random testing, augmenting the instruction set with higher level constructs & simulating backward deduction used by humans by using a goal-based heuristic.   1. **Solving geometry problems using a combination of symbolic and numerical reasoning**   This paper combines deductive, numeric and inductive reasoning to solve geometric problems. Using a mixture of these methods, several problems are solved which, otherwise, wouldn't have been solved using the methods individually. The paper uses the number of nondeterministic choices as a measure of how close a partial program is to the solution, claiming this to be useful in grading and providing hints to the students. The implementation takes an average of few seconds on 18 SAT geometry problems and 11 other ruler & compass-based geometry construction problems.   1. **Learning to map sentences to logical form: Structured classification with probabilistic categorical grammars** [citation]   With the objective of mapping natural language sentences to their counterpart lambda-calculus encoding, this paper introduces an algorithm which learns from the input set of natural language sentences labeled with lambda calculus expressions. The algorithm attempts to infer the underlying distribution over semantic and syntactic analyses conditioned on the input sentences and outputs a log-linear model.   1. **Can modern statistical parsers lead to better natural language understanding for education?**   Focusing on the educational domain, this paper seeks to build robust domain specific natural language understanding modules by suitable preprocessing of the text and use of domain knowledge to deal with parser errors. The paper starts with showing that even off-the-shelf parsers do not perform up to the mark on texts with domain-specific vocabulary and long sentences. The paper also talks about GeoSynth, built using state-of-the-art parsing technology. GeoSynth is a system which can automatically solve geometric constructions word problems. |
| Design |
| The interpreter consists of two main components:   1. **Aligner**   GIZA++v2 (detailed later) is the heart of learning mechanism in this project. We use GIZA to obtain mapping/alignment between a natural language and our carefully designed and structured imperative metalanguage, L0. Since this alignment can be obtained for any natural language, the application is scalable to any number of natural languages.   1. **Metalanguage Interpreter**   The interpreter component exploits the alignment model obtained from the Aligner component. Given a sentence expressed in a natural language, we use the alignment model and characteristics of L0 to get statistically most probable interpretation in L0. A command in L0 is then easily parsed to output the desired figure. |
| GIZA++ |
| GIZA++ is a statical machine translation toolkit that is used to train IBM Models 1-5 and an HMM word alignment model.  For any pair of languages, given sufficiently many sentences as illustrated in the table below, GIZA++ outputs statistical alignment between words of the language pair. In other words, the alignment model assigns probabilities to the event that a particular source-language token is mapped to a target-language token.   |  |  |  | | --- | --- | --- | | **English** | **Hindi** | **Metalanguage** | | Construct a line segment AB of length 4 cm | 4 सेमी लम्बाई का एक रेखाखण्ड AB खींचिए | construct lineSegment AB length 4 cm | | With A as center and radius 3 cm, draw an arc. | केंद्र A और त्रिज्या 3 सेमी लेकर एक चाप खींचिए | construct arc center A radius 3 cm | | With B as center and radius 5 cm, draw an arc cutting the previously drawn arc at C. | केंद्र B और त्रिज्या 5 सेमी लेकर एक चाप खींचिए जो पहले खींची चाप को C पर काटता हो | construct intersectingArc center C radius 5 cm cuts arc ? at C |   An example alignment between English-&-Metalangue and Hindi-&-Metlanguage could be:   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | **English** | **Metalanguage** | **Probability** | | Construct | construct | 0.98 | | Draw | construct | 0.98 | | line segment | lineSegment | 0.95 | | length | length | 0.07 | | arc | arc | 0.99 | | center | center | 0.98 | | radius | radius | 0.97 | | cutting | cut | 0.8 | |  |  |  | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | | **Hindi** | **Metalanguage** | **Probability** | | खींचिए | construct | 0.99 | | रेखाखण्ड | lineSegment | 0.99 | | लम्बाई | length | 0.98 | | चाप | arc | 0.98 | | केंद्र | center | 0.99 | | त्रिज्या | radius | 0.98 | | काटता | cut | 0.8 | |  |  |  | | |
| What has been done |
| 1. English-to-Metalanguage and Hindi-to-Metalanguage sentence pairs have been collected from NCERT textbooks to be used in bootstrap corpus. 2. The approach has been shown to work for simple construction steps. Output diagram for the steps above is shown below.   A description... |
| Results |
| 1. Number of sentences in the corpus: 360 each in English-to-Metalanguage and Hindi-to-Metalanguage corpus 2. A sample prototype implementation that demonstrates the performance of the system for simple construction steps of drawing line segments and arcs. 3. Number of unique tokens in each of the three languages:  |  |  | | --- | --- | | **Language** | **Number of unique tokens** | | English | 181+ | | Hindi | 169+ | | Metalanguage | 110+ | |
| What needs to be done |
| 1. Enrich the corpus to cover larger domains of expressions e.g. having larger number of primitive steps, providing for higher-level primitives, specifying relative distances, dealing with anaphoras and references 2. Expand the functionality of the system to work for all kinds of construction steps claimed so far 3. Deploy the system on a webserver to demonstrate its functionality and work towards making it more robust 4. In order to capture colloquial (non-academic) ways of expressing construction steps, we need to develop a user interface that augments the corpus with input sentences |
| References |
| 1. Schreck, Pascal, Pascal Mathis, and Julien Narboux. "Geometric Construction Problem Solving in Computer-Aided Learning." Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on. Vol. 1. IEEE, 2012 2. Gulwani, Sumit, Vijay Anand Korthikanti, and Ashish Tiwari. "Synthesizing geometry constructions." ACM SIGPLAN Notices. Vol. 46. No. 6. ACM, 2011 3. Ahmed, Umair Z., et al. "Can modern statistical parsers lead to better natural language understanding for education?." Computational Linguistics and Intelligent Text Processing. Springer Berlin Heidelberg, 2012. 415-427 4. Itzhaky, Shachar, et al. Solving geometry problems using a combination of symbolic and numerical reasoning. Technical report, Tel Aviv University, 2012 5. Cheema, Salman, Sumit Gulwani, and Joseph LaViola. "Quickdraw: improving drawing experience for geometric diagrams." Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems. ACM, 2012 6. Zettlemoyer, Luke S., and Michael Collins. "Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars." arXiv preprint arXiv:1207.1420 (2012) |