BIG DATA MATH:
NATURAL LANGUAGE PROCESSING AND AUTOMATIC THEOREM PROVING
SPECIAL TOPICS COURSE PROPOSAL

MAREK RYCHLIK, PROFESSOR OF MATHEMATICS

MOTIVATION

Imagine a computer program which utilizes the tremendous processing power of computers to answer any question posed. As this power quickly approached that of the human brain, the challenges are numerous. As the IBM computer named "Watson" gained popularity by answering trivia type questions, its perceived intelligence is based primarily on memorization and search. What if you asked Watson: "Prove or disprove the Riemann Hypothesis"? At best you would see the knowledge on the subject being regurgitated.

A new generation of intelligent systems will be endowed with new processing capabilities, deeply rooted in mathematics. We would like the computer to "understand" the question posed and independently "reason" about the problem posed. We may ask the machine to recognize "interesting" mathematical theorems, and formulate "original" research hypotheses, and go about researching them autonomously. In this course we will review a number of approaches to the problem, involving mathematical logic, computational algebra and their relationship to natural language processing, and various automated theorem proving systems. The view is towards a computer program that can be queried about mathematics and autonomously write "original" mathematical papers. Questions like "Computer, prove the Pythagorean Theorem" and other questions of point plane geometry will be easy exercises. It should be noted that problems posed in this course share many characteristics of "big data" challenges at the center of progress in science and technology.

COURSE CONTENT

Automated deduction systems, such as Free systems, tableaux, resolution refutation proofs, first order logic with equality (para modulation); λ-calculus of Alonzo Church and Turing machines. DCG grammars. Formal mathematical theories such as the Zermelo-Fraenkel set theory and Peano arithmetic; geometric theorem proving via Gröbner bases. Making computers think like mathematicians do.

Textbooks and other course materials. The main text for the course will be a set of lecture notes developed by the instructor. Another text is the book by Melvin Fitting entitled "First-Order Logic and Automated Theorem Proving" (ISBN 0-387-94593-8). Many supplementary materials will be provided by the instructor in electronic form.

Assignments and Exams. This course will be project oriented. The grade in the course will be based on a number of assignments and small-to-medium programming projects.

Prerequisites. A course in mathematical logic or advanced discrete mathematics. Experience writing formal mathematical proofs. Real software systems will be used, including PROLOG, Lisp and free computer algebra systems. Basic programming skills are required, although the knowledge of Lisp or PROLOG is not.

Contact information. Please contact the instructor by e-mail at **rychlik@email.arizona.edu**.