CSC490: Cognitive Computing Capstone

Cognitive computing is the a combination of machine learning, artificial intelligence and natural language processing, used to perform tasks on unstructured domains. Cognitive computing systems use training and learning algorithms to sense, predict and infer conclusions that cannot be handled by current search techniques. The focus of this capstone course is the use of IBM's Watson technology to create a marketable product that solves a significant problem using cognitive computing.

Instructor Information

Name	**Office**	**Phone**	**Email**
Steve Engels	BA4266	(416) 946-5454	sengels@cs.toronto.edu (put "490" in subject)
Mario Grech	BA5224		mario.grech@utoronto.ca
Helen Kontozoloulos	BA5224		helen@cs.toronto.edu

Course Materials

- **Web Page:** http://www.cs.utoronto.ca/~csc490h
- **Instructor Contact:** csc490@cs.utoronto.ca

Course Schedule

Lecture	**Topics**	**Milestone**	**Weight**
Sept 11	Intro to CSC490	Watson Hackathon (*Saturday*)	5%
Sept 18	Intro to Watson	Corpus Training	5%
Sept 25		Product Pitch	10%
Oct 2	NLP Basics		
Oct 9		Project Design	15%
Oct 16	UI/UX (Basics)		
Oct 23		Minimum Viable Product	15%
Oct 30	UI/UX (Advanced)		
Nov 6		Software Beta (feature complete)	15%
Nov 13	Topic TBA		
Nov 20		Final Product	20%
Nov 27		Demo Showcase	5%

Participation component: 10%

Project Milestones

Note: *For each presentation, you are expected to critique other groups when you are not presenting.*

Watson Hackathon

- Learn the different user roles and APIs for Watson.
- Create corpus from documents, assign training questions, get responses.

Corpus Training

- Build Watson corpus from training documents.

Product Pitch (in-class presentation)

- Propose project idea (focus on technical feasibility).

Project Design (in-class presentation)

- Present proof of concept that illustrates the core technical needs of project.
- **Submit:** Design document (use cases, software structure diagrams, product backlog, scheduling diagrams, etc)

Minimum Viable Product (in-class presentation)

- Implement most use cases
- Present product demo
- Basic product prototype
- Initial product testing (for Beta stage)

Software Beta (in-class presentation)

- Present feature complete product
- User experience goals
- Present initial testing report, perform secondary testing (for Final Product)

Final Product (in-class presentation)

- **Submit:** final software
- **Submit:** final testing report
- Demo to industry panel

Demo Showcase

- Demo at event for general public

Administrative Details

Plagiarism is very bad. Please don't do it. It just makes things unpleasant for everybody involved. In case you need clarification on the university's policies on plagiarism, please consult the *Code of Behaviour on Academic Matters* from this website: www.artsci.utoronto.ca/osai/students