**CS590 Week2**

* Collect HW1

**What is Publishable Research?**

**Research Techniques and Steps**

I will use my dissertation topic as an example below.(E.G.s)

**1. Select a topic/issue that nobody else has addressed adequately**

- nobody has worked on it

- others have worked on it but left some issues unresolved

(e.g. the algorithm is too slow for large data)

- others have worked on it but for another application area

(e.g. the algorithm works for medical applications but not for others)

This requires literature review that we talked about. Unless you have reviewed other people’s work thoroughly, you cannot confidently claim that your work is unique.

E.G. Machine translation from Japanese to English

* Few people have worked on it
* Other works have not handled ungrammatical / telegraph style sentences.
* Other works did not use contextual information.
* Other works have not dealt with scientific papers and manuals.
* Other works were mostly translation aids and not fully automatic translators.

**2. Define your problem and goals well (in detail)**

* which of the above unresolved issue(s) are you going to address?
* for what application areas is this useful?
* what are the expected results? **(\*\*\*)**

e.g. speed improvement?

e.g. additional cases it will handle?

e.g. it matches human cognition?

e.g. it improves human learning or behaviour?

EG. Automatic Machine Translation of Instruction Books from Japanese to English

* Handle ungrammatical sentences
* Handle any instruction books including recipes simply

by editing the data file and dictionary

* Be able to translate at least 30 different instruction books of different types
* Translate in real time
* Interact with the user only when unknown words are found.

**3. Formulate your theory (proposed solution)**

* study the characteristics of the application area
* study what others have done in this application area
* study any related findings in other areas of CS, Linguistics or math, etc.

**===> your theory**

E.G.

Studied how humans read and understand Japanese

* + Use forward expectations from nouns to predict the verb
  + Use the verb to understand the roles of the nouns.
  + Use knowledge of the subject to predict the next action.

*This is similar to top down + bottom up reasoning in AI*

*and Scripts in AI.*

*e.g. Egg o bowl ni 🡺 predict Place*

*e.g. Egg bowl place 🡺 infer Egg is the object*

*e.g. Place then Mix 🡺 predict Heat => Serve*

**4. Proving your theory (implement, test and evaluate)**

**Design** a system embodying your theory, implement it, and test it.

* test it on a variety of data to see if it works as expected as per **\*\*\*** above, or
* test it on human subjects to see if it works as expected as per **\*\*\*** above (this requires working with human subjects.)
  + - There are professionals to design questionnaire and interviews, and they do analyses.

**Analyse** the test results:

* you did not have solved all of the problems you intended to solve?
* you noticed additional issues?

E.G. Implemented my theory as a translation system in Lisp.

* Tested it on a variety of instruction books
* Produced correct translations

Unresolved issues:

* difficult and time consuming to build the dictionary by hand
* interaction with a human user on unknown words is unfriendly.

**Other Examples of tests you can do:**

\*\* If the goal was a faster algorithm

* do O/Theta analyses of your algorithm
* graph speed change for different sizes of data
* did the algorithm work correctly in all cases?

\*\* If the goal was to match human cognition

* the results of your system matches

how human subjects did in all cases?

(i.e. made the same errors, went through the same steps)

\*\* If the goal was a “better” interface

* test your interface on human subjects

\*\* If the goal was a better learning system

* test your system on students but evaluating effectiveness is very difficult.

**In class exercise on proving your theory**

* **Each person will pick one of the following and come up with the answers. (30 min)**
* **Each person will present his or her answers.**
* **Others must give feedback on the technique.**
* **Requirement: actively and seriously participate.**

**How about a new software engineering methodology?**

-- your methodology will be better in what ways?

-- how would you prove your theory?

**How about a new operating system?**

-- your new OS will be better in what ways?

-- how would you prove your theory?

**How about a new database query language?**

-- your new language will be better in what ways?

-- how would you prove your theory?

**How about a new learning software?**

-- your new software will be better in what ways?

-- how would you prove your theory?