KBAI Project 3 (learning)

Version 11/16/2018

Date	Changes (highlighted in yellow in the specification)
11/07/2018	Modified example questions to remove ‘daily’.
11/07/2018	Student agent should return string as opposed to int.
11/07/2018	Corrected some spelling.
11/10/2018	Released version 11/10/2018.
11/11/2018	Removed reference to ‘class’ in example statement.
11/12/2018	Added to statement requirements. Highlighted green words in example statements.
11/13/2018	Added sections: Question Answering Rules, Domain Limitations.
11/16/2018	Reorganized sections to move vocabulary next to statement/question rules. Removed ‘specification’ and ‘specifications’ from vocabulary.

	Project 1 Understanding	Project 2 Classification	Project 3 Learning
KBAI Algorithms	• Thematic Roles • Frames	• Semantic Networks • Generate and Test • Production Systems (rules based systems) (decision trees) • Means-Ends Analysis • Problem Reduction • Case Based Reasoning	• Incremental Concept Learning • Version Spaces • Analogical Reasoning • Logic • Planning • Scripts • Learning by Correcting Mistakes • Learning by Recording Cases

Learning

"*Much of human learning can be viewed as a gradual process of concept formation. In this view, the agent observes a succession of objects or events from which he induces a hierarchy of concepts that summarize and organize his experience*." [2]

Incremental Concept Learning

"*Incremental concept learning is intimately connected with human cognition. We can adopt two views of learning. In one view of learning, the intelligent agent is given with a large number of examples. The agent's task, then, is to detect patterns of regularity in those examples and learn those patterns of regularity. In the alternative view, the agent is given one example at a time. And the agent has to gradually, incrementally learn concepts out of those examples.*" [1]

Version Spaces

"*Version spaces is a technique for learning concepts incrementally. This means that a technique is going to learn from a small set of examples, that are going to arrive one example at a time.*" [1]

Analogical Reasoning

"*Analogical Reasoning involves understanding new problems, in terms of family of problems. It also involves addressing new problems, but transferring knowledge of relationships from known problems across domains.*" [1]

Logic

"*Logic is a formal language that allows us to make assertions about the world in a very precise way.*" [1]

Planning

"*Planning uses states, operators and goals in formal logic.*" [1]

Scripts

"*A script is a knowledge representation for capturing causally coherent set of events. Casually means that one event sets of another.*" [1]

Learning by Correcting Mistakes

"*Learning by correcting mistakes is a fundamental process of human learning. In fact, it may closely resemble the way you and I learn and practice. In our lives, we rarely are passive learners. Most of the time we're active participants in the learning process.*" [1]

Learning by Recording Cases

"*Learning by storing cases in memory has a very strong connection to cognition. Cognitive agents like you and I are situated in a world. Our interactions with the world have certain patterns of regularity. The world offers us the same problems again and again. If we think about it, the kinds of problems that you and I deal within a routine everyday basis are the same problems that occurred yesterday and the day before.*" [1]

Project Goals

Project 1 covered inference. Project 2 covered classification. Project 3 builds on project 2 with the addition of learning. The goal for this project is to understand learning by developing an agent that learns. "*Roughly speaking, learning is the process of converting experience into expertise or knowledge. The input to a learning algorithm is training data, representing experience, and the output is some expertise, which usually takes the form of another computer program that can perform some task.*" [9]

In project 2, the syllabus knowledge was coded into the agent directly. For project 3, the agent must learn the syllabus. Your agent is given a knowledge base in the form of a vocabulary to bootstrap the learning process. The vocabulary is coded into the agent. The agent must use the vocabulary to learn the syllabus knowledge at run time. Bootstrapping the learning process. After learning the syllabus. The agent is asked questions about the syllabus, similar to project 2.

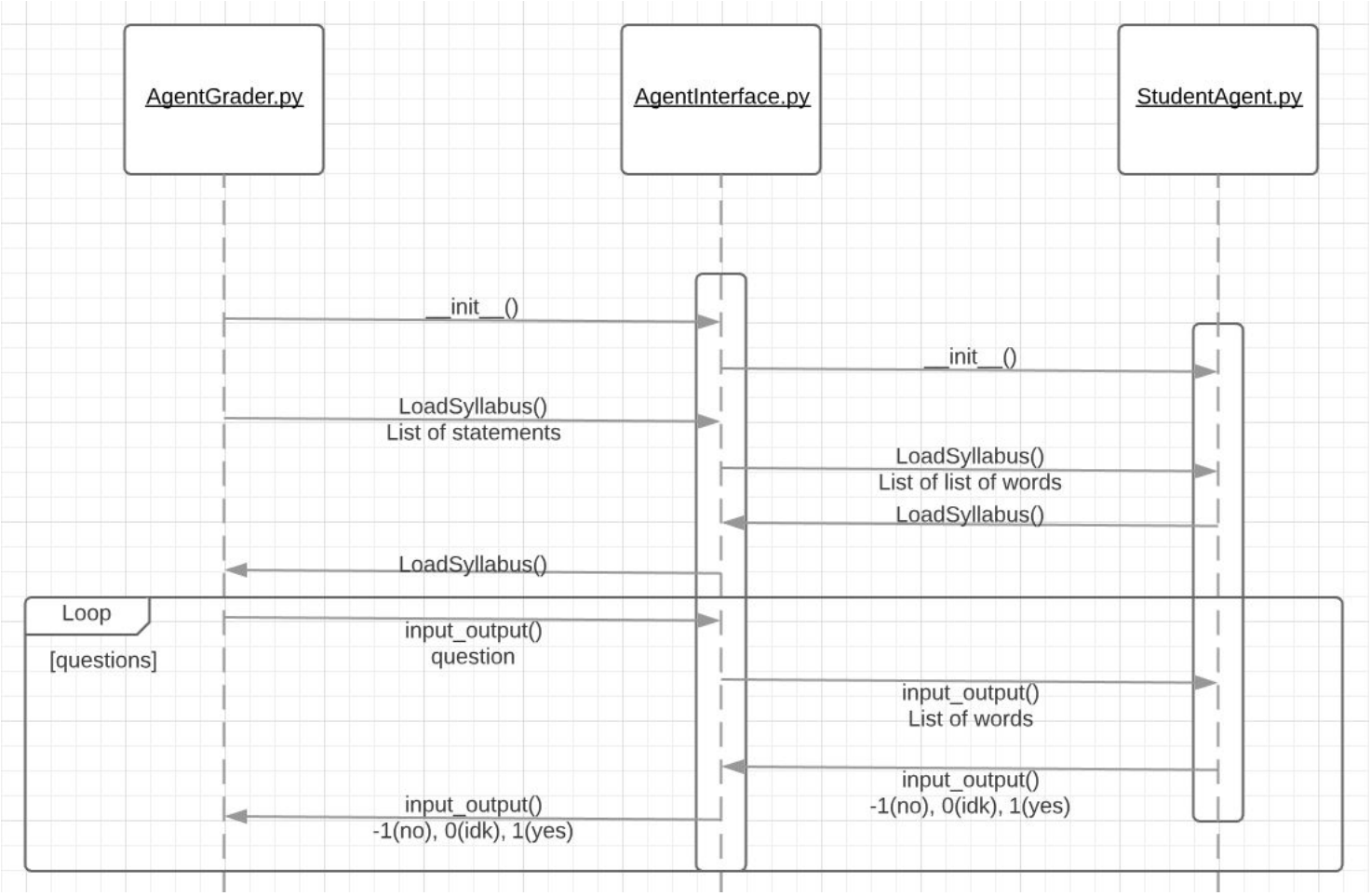
The resulting project 3 agent will be capable of learning a syllabus automatically and answering questions about the class defined by the syllabus. A simple virtual TA. Although simple, the KBAI techniques used in projects 2 and 3 are the same KBAI techniques used in Jill Watson.

Project Details

For project 2 you were given a syllabus in advance. You "trained" your agent by coding the syllabus details into the agent. For project 3, your agent must learn a syllabus at runtime. The only prior knowledge is the vocabulary.

Your agent will be given a list of statements using the vocabulary to describe a syllabus (world). Your agent must then answer yes/no/I Don't Know(IDK) questions about the resulting syllabus (world). The only available prior knowledge of the syllabus (world) is the vocabulary.

Sequence Diagram



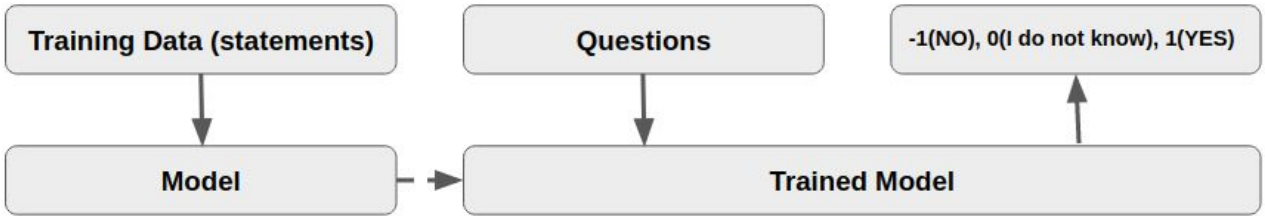
Sequence

1. AgentGrader instantiates AgentInterface which instantiates StudentAgent
2. AgentGrader calls LoadSyllabus() with a list of **statements**
3. The StudentsAgent should build its model based on the **statements** before returning from LoadSyllabus()
4. The AgentGrader will call input_output() with **questions** one at a time (just like P1 and P2).
5. For each **question**, the StudentAgent should return string: “no", "idk" , "yes

Syllabus Definition

After your agent is instantiated, it will be passed a list of statements using the vocabulary. The statements are training data. Your agent must build a model of the syllabus based on the training data. After training, your agent will be asked questions about the syllabus. Your agent must use its model to answer the questions.

Your agent does not know anything about the syllabus in advance. The examples (statements and questions) below are strictly examples. The test syllabus will be unique within the limitations of the defined below.



Domain Vocabulary

project	the	AI	piazza	yaroslav	monday
projects	will	during	due	mandatory	tuesday
assignment	be	long	slack	python	wednesday
assignments	in	and	ashok	canvas	thursday
midterm	is	pdf	goel	learning	friday
final	of	doing	start	everyday	saturday
course	has	need	litvak	finish	sunday
announcements	to	into	posted	complete	procedure
instructor	are	turn	class	check	reading-list
report	a	xx%	week	many	contribute
reports	this	java	weeks	submit	class-grade
~~specification~~	as	c++	videos	there	morning
~~specifications~~	by	close	hours	reading	midnight
exams	for	list	video	human	classroom
strategies	on	book	late	occurs	available
strategy	all	does	end	example	submitted
policy	occur	docx	files	regularly	distributed
peer-feedback	must	text	after	should	attendance
office-days	no	worth	write	method	knowledge-based
content	every	file	learn	planning	self-reflection
communication	do	open	have	cognition	peer-to-peer
submissions	can	turned	teach	released	decision-making
TA	get	credit	0-100	preferred	collaboration
component	I	zip	grade		
code	if	gz	work		
policies	my	bz2	begin		

To avoid confusion, this table is the single source vocabulary reference

green background	Syllabus objects. These objects represent concepts for a generic classroom. These concepts are derived from reviewing various class syllabi. Some concepts are discrete, but others may overlap. **To learn more about how some of these concepts apply to a syllabus, please review the syllabus for this class as an example.**
white background	These words are either assigned to concepts, or used to make assignments to concepts.

Example Syllabus

Statements (loaded at initialization)	Questions (asked after initialization)	Ans	Explanation
the course is 10 weeks	is this a 16 week class	no	course = 16week
project 1 will be available start of week 3	does project 1 begin on week 3	yes	project 1 = week 3
Project 1 will be due end of week 6	is project 1 due on week 6	yes	project 1 = week 6
projects must be turned into canvas	do I submit projects to canvas	yes	project = canvas
assignment 1 will be released in week 6	can I get assignment 1 in week 6	yes	Assignment 1 = week 6
assignment 1 is due week 7	do I have 1 week to complete assignment 1	yes	Assignment 1 = week 7 - week 6
the midterm occurs on week 8	is there a midterm	yes	Midterm = week 8
the final occurs on week 9	Is the final during week 9	yes	Final = week 9
the final must be submitted to piazza	do I need to check piazza everyday	**idk**	We have no data to infer this
projects are submitted as zip files	do I turn in my project as a pdf	no	project = zip
assignments are submitted as pdf files	do I turn in my assignments as zip files	no	assignment = pdf
	are there 2 projects	no	project 2 = NULL
	is the midterm after assignment 1	yes	Assignment 1 = week, Midterm = week 8
	will I have 3 weeks to finish project 1	yes	project 1 = available week 3
	can I have 4 weeks to finish project 1	no	project 1 = 3 weeks (has release and due)
	can I start project 1 in week 2	no	project 1 = available week 3
	do I need to check Piazza regularly	**idk**	We have no data to infer this
	will I learn human cognition	**idk**	We have no data to infer this
	is project 1 worth 15%	**idk**	We have no data to infer this
	is assignment 1 worth 20%	**idk**	We have no data to infer this
	does this class teach knowledge-based AI	**idk**	We have no data to infer this
	are projects distributed on Canvas	**idk**	See question and answer rules
	does this class have projects and assignments	yes	project 1, Assignment 1
	is there a final in this class	yes	Final = week 9
	are there many projects	no	project 2 = NULL
	~~is there a procedure for project 1~~	~~**idk**~~	This question is too complex for this project
	can I submit projects to Piazza	no	project = canvas
	should I code my projects in Python	**idk**	We have no data to infer this
	does project 1 contribute 15% to my grade	**idk**	We have no data to infer this
	do I need to write a report for project 1	**idk**	We have no data to infer this
	do I need to check Piazza regularly	**idk**	We have no data to infer this
	If I turn in project 1 late do I get credit	**idk**	We have no data to infer this
	Is Piazza the preferred method of collaboration	**idk**	We have no data to infer this

Test Statement/Question Constraints

Statement Limitations

Statements are limited to the domain vocabulary.
Statements are limited to 11 words including digits.
Statement must contain one and only one green word.
Statements are both simple and semantically correct.

Question Limitations

Questions are limited to the domain vocabulary
Questions are limited to 11 words including digits
Question must be answerable with yes, no, or I do not Know (IDK)
Questions are semantically correct.

Domain Limitations

- No object will be assigned 100% of the grade.
- No question will require inferring a grade %.
- There will be no conflicting information.
- If project X exists, so does projects (X-1), (X-2), (X-3), …..
- Many means more than 1.
- No commas in statements or questions.
- Questions will be similar to project 2.

Question Answering Rules

The goal is to provide an answer to the user that is as accurate as possible.

- To do this, the agent must be able to answer 'I Do not Know".
- IDK tells the user to look for the answer using another resource.
- We need to use open-world assumption, but with some rules to explicitly define when to use **idk** versus **no**.
- With the goal of helping the user as much as possible, **we want to answer no, only when we are sure the answer is no.**

yes	If the object/concept is assigned a parameter the questions explicitly asks about. If the question asks about data that can be inferred from a combination of parameters.
no	If the object/concept is assigned a parameter the questions explicitly asks about.
idk	If the data required for the answer is not explicitly defined or cannot be inferred from explicit definitions. If the agent has conflicting information. **Are projects distributed on Canvas?** Although projects are turned into Canvas, it is risky to infer they are distributed by Canvas. It would be best for the user to get this information elsewhere.

Student Examples

Statement: project 1 is due week 3
Statement: project 2 is due week 6
Statement: project 3 is due week 9
Statement: projects must be turned into canvas
Question: Must I turn project 4 into canvas? <<< is this idk or no
conflicting information - There is no project 4, all projects are turned into Canvas
Answer: idk
Statement: The class is on Monday and Wednesday.
Statement: Attendance is mandatory on Monday and Wednesday.
Question: Does the class have mandatory attendance? - YES
Question: Is attendance mandatory everyday in class? - YES

What to Do

1. Create a syllabus defining the attributes of the objects (green in vocabulary chart).
2. Encode the syllabus into statements using the vocabulary (this is your training set).
3. Share your statements on Piazza.
4. Create a list of yes/idk/no questions about your syllabus.
5. Share your questions on Piazza.
6. Help your fellow classmates, and ask questions about their syllabus on Piazza.
7. Test your agent with your syllabus statements, and those of your classmates.

Getting the code

```
git clone https://github.gatech.edu/Dilab/CS7637AOProjects.git
```

Look in project 3 directory

Executing the code (you must use Python 3)

```
python AgentGrader.py -v
```

Code (in project 3 directory)

File	Change?	Description
syllabus.json	YES	Contains an example syllabus defined by statements along with example questions. Overwrite with your syllabus.
StudentAgent.py	YES	Add your code here
Syllabus.json	YES	Example statements and questions
AgentInterface.py	NO	Autograder to agent interface
AgentGrader.py	NO	The autograder will test your agent and output a result

You can ADD more files for your project. The autograder simply overwrites the files in red above.

Grading

The autograder will pass a syllabus to your agent as a list of statements. The autograder will then ask your agent a set of yes, idk, and no questions. Your agent gets one point for each correct answer.

Reflection Rubric (50% of grade) (scientists ask why)

Item #	P	Address in your report
1	10	**Metacognition:** Given a syllabus described by a "list of statements": ● As you read the "list of statements", explain your cognitive process. ● What happens with your mental model after your read each statement? ● What prior knowledge, did you (as a human) require to create your mental model? ● How did the number of statements affect your resulting mental model?
2	10	**Map human thought processes to KBAI techniques:** Based on your experience answering question 1. ● Create a diagram that describes the cognitive process **you** used to creating **your mental model** from question 1. ● How do the KBAI techniques taught in class compare to the cognitive processes you followed in question 1? ● Are there techniques you use for which there is no corresponding KBAI technique? ● What are the tasks or processes at which human brains perform better than computational systems?
3	10	**Program a learning agent based on your cognitive process:** Based on your diagram from question 2. ● Create a diagram of **your agent**. ● Compare this diagram to the diagram of **your mental model** (created in question 2). ● Explain the comparison with an emphasis on why. Why are there differences? ● Please mention any tradeoffs you faced and explain the rationale behind your design decisions.
4	10	**Reflection**: ● How are these differences reflected in the approaches we use to solve problems? ● Assuming an infinitely scalable vocabulary, what, if any, are the limitations of your chosen process? ● Will it be close to human-level intelligence, why or why not?
5	10	**Speculation:** ● What makes a question answering system “robust” or “brittle”? ● Where does your agent fall on that spectrum? ● There is an agent (written using KBAI techniques) running on Piazza for this class. ● Who do you think is the agent? Why?

Please refer to the class syllabus for guidance on report writing and word limits.

Libraries

Only libraries listed below are allowed. Your code can only import modules you have created or modules listed in the table below.

Legal Libraries (ALL OTHER LIBRARIES ARE NOT PERMITTED)

Python 3 standard library (https://docs.python.org/3/library/)

The autograding machine will not have any additional libraries installed. Please verify your imports do not require any libraries not listed above. **If you use any libraries not listed above, your code will not run on the autograding machine and you will get a 0.**

Some Python IDE's will automatically import debug libraries. Please test your final code outside of the IDE directly from the command line in a clean Python environment.

References

1. KBAI Ebook
2. Gennari, John H., Pat Langley, and Doug Fisher. "Models of incremental concept formation." Artificial intelligence 40.1-3 (1989): 11-61.
3. Aha, David W. "Case-based learning algorithms." Proceedings of the 1991 DARPA Case-Based Reasoning Workshop. Vol. 1. 1991.
4. Structural Case-Based Reasoning and Ontology-Based Knowledge Management: A Perfect Match?
5. Explanation-Driven Case-Based Reasoning
6. An Ontology-based Similarity Measurement for Problem-based Case Reasoning
7. An Ontology-Supported Case-Based Reasoning Technique for FAQ Proxy Service
8. Amjad Abou Assali, Dominique Lenne, Bruno Debray. Case retrieval in ontology-based CBR systems. MERTSCHING, B. ; HUND, M. ; AZIZ, Z. 32. Annual Conference on artificial intelligence (KI 2009), Sep 2009, Paderborn, Germany. Springer, 5803, pp.564-571, 2009, Lecture Notes in Artificial Intelligence. <10.1007/978-3-642-04617-9_71>.
9. Understanding Machine Learning: From Theory to Algorithms
10. Building Machines That Learn and Think Like People
11.