Deceptive and Counter-Deceptive Machines
Papers from the AAAI 2015 Fall Symposium

Toward An Intelligent Agent for Fraud Detection — The CFE Agent

Joe Johnson
Department of Computer Science, Rensselaer Polytechnic Institute (RPI)
Troy, NY 12180 USA

Abstract

One of the primary realms into which artificial intelligence research has ventured is that of psychometric tests. It has been debated since Alan Turing proposed the Turing Test whether performance on tests should serve as the metric by which we should determine whether a machine is intelligent. This is an idea that may either solidify or challenge, depending on the reader's predisposition, one's sense of what artificial intelligence really is. As will be discussed in this paper, there is a history of efforts to create agents that perform well on tests in the spirit of an interpretation of artificial intelligence called "Psychometric AI". However, the focus of this paper is to describe a machine agent, hereafter called the CFE Agent, developed in this tradition. The CFE Exam is a gateway to certification in the Association of Certified Fraud Examiners (ACFE), a widely recognized professional credential within the fraud examiner profession. The CFE Agent attempts to emulate the successful performance of a human test taker, using what would appear to be simplistic natural language processing approaches to answer test questions. But it is also hoped that the the reader will be convinced that the same core technologies can be successfully applied within the larger domain of fraud detection. Further work will also be briefly discussed, in which we attempt to take these techniques to the next level, a deeper level, by which we can get a better sense of the knowledge the agent is using, and how that knowledge is being applied to formulate answers.

1 Introduction

Psychometric AI, or PAI, is grounded on the notion that by making machines that can pass tests, the machine can, in some sense, be called "intelligent"(Bringsjord 2004). Although it's readily conceded that the ability to pass a particular test (or particular set of tests) is an inadequate basis for defining the full range of what it means to be intelligent, it does offer the very attractive element of concreteness(Bringsjord 2004) — a very appealing feature given the struggle one commonly has when attempting to delineate what it means to make machines that can "think". After all, tests offer a number of concrete aspects — a well-defined start, end, set of goals implied by the questions themselves, and ultimately, a score.

Evans implicitly began a tradition of test-taking agents by creating ANALOGY in 1968, an agent that could answer geometric analogy questions(Evans 1968). PERI, an agent created by RPI's RAIR lab in early 2000s, was a robotic agent that could successfully answer Raven's Progressive Matrices questions(Raven 1962) as well geometric shape questions similar to those found on the WAIS (Wechsler Adult Intelligence Scale) exam (Bringsjord 2004). Attempts to create such agents even explored the creative realm, such as the one developed to generate stories in a test-based context(Bringsjord 2004; 2000; Torrance 1990).

The agent described in this paper is designed to take a professional exam, the CFE Exam, that tests knowledge in the fraud detection domain. This paper will provide some context for the CFE Exam by giving a brief description of its sponsoring organization, the Association of Certified Fraud Examiners (ACFE), and the general features of the test. Then, it will discuss the approach for building the Agent its performance, as well as directions for future research.

2 The ACFE and the CFE Exam

The ACFE, (www.acfe.com), describes itself on its website as "the world's largest anti-fraud organization and premier provider of anti-fraud training and education". Generally speaking, in order to become a readily employable expert in the field of fraud detection, certification by this organization is required. The ACFE has well-defined requirements for becoming certified, based on a point system that considers a combination of professional experience and academic credentials. However, the CFE exam is the credentialing centerpiece for the ACFE, and the details of this exam are described briefly below.

The CFE Exam is a computer-based exam. The mechanics for preparing for and taking the exam begins with downloading a software package from the prep course page of the ACFE website. This package includes the exam software, the Fraud Examiners Manual (on which the test is based), and a self-study application consisting of a battery of sample test questions, a complete practice exam, and tools for monitoring progress.

The CFE exam consists of 4 sections, listed below:

- Financial Transactions and Fraud Schemes
- Law

- Investigation
- Fraud Prevention and Deterrence.

Each section consists of 125 multiple choice and true-false questions. The candidate is limited to 75 seconds to complete each question and a maximum total allocated time of 2.6 hours to complete each section. Each CFE Exam section is taken separately. The timing for each section is subject to the candidates discretion. However, all four sections of the exam must be completed and submitted to the ACFE for grading within a 30 day period.

The goal was to develop a computer system that could pass the CFE exam subject to the constraints outlined above. This project involved procuring the study package and exam resources, customizing their format for automated processing, and developing and testing the CFE Agent, an extensible system that utilizes various algorithms along with these resources to answer test questions on a given exam.

3 Resource Procurement and Customization

The first phase of this project after downloading the study package was to convert its contents to a form that could be processed by an automated system. The content in the package include a Fraud Examiners Manual in pdf format, practice questions embedded within study package software, and a practice exam, also consisting of questions embedded within custom software. It was necessary to convert each of these components into text-formatted documents for the CFE Agent.

The first of these components, the Fraud Examiners Manual, is a 2,200 page pdf document that serves as the principal resource from which all of the exam questions are drawn. Conversion of this document involved an extensive amount of work for two reasons: First, the pdf document itself was password protected, rendering most pdf conversion programs useless. And second, the few software packages that were capable of bypassing the password protection produced intermittent errors in the generated text, inserting spaces randomly between the letters of words. Given the large volume of content, a program employing various parsing techniques was written to sequentially walk through the text, detect errors, make corrections automatically based on certain rules, or prompt for correction if an error was detected but manner of correction was uncertain. Although a significant amount of effort was expended to develop this technology, in the end much of the document was edited manually.

The other main component of the prep package is the database of 1,500 questions embedded in the study package software. The extraction of these questions from the software was an involved, multi-step process. First, a screenshot file, as shown in Figure 1, was created for each question. Second, optical character recognition (OCR) software was used to convert the graphic screen shot files into text files, as shown in Figure 2. Unfortunately, the important data needed for each question was intermingled among nonsensical overhead text in each of these text files. So, a custom program was created to parse these text files, remove the unnecessary data, and decompose each question into its component parts

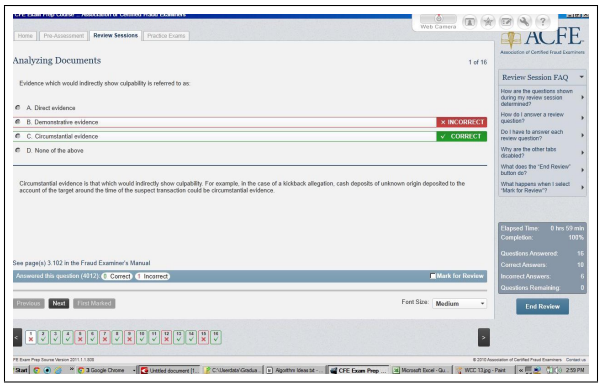


Figure 1: Embedded Practice Question

```
* CFE Exam Prep Course :: Association of Certified Fraud Examiners
,©
Web Camera
B00
Pre-Assessment Review Sessions Practice Exams
Analyzing Documents
Evidence which would indirectly show culpability is referred to as: C A. Direct
evidence
1 of 16
C B. Demonstrative evidence
C C. Circumstantial evidence
x INCORRECT
C D. None of the above
Circumstantial evidence is that which would indirectly show culpability. For example,
in the case of a kickback allegation, cash deposits of unknown origin deposited to the
account of the target around the time of the suspect transaction could be
circumstantial evidence.
See page(s) 3.102 in the Fraud Examiner's Manual
Answered this question (4012):QQBB5) (MQZE3D
■I Mark for Review
Previous I Next
Font Size:
Medium
1
X
2 II 3 II 4 □ M □
```

Figure 2: Converted, Unscrubbed Practice Question

```
Section=Analyzing Documents

Stem=Evidence which would indirectly show culpability is referred to as:

Options=Direct evidence | Demonstrative evidence | Circumstantial evidencex | None of the
above

CorrectResponse=2

Explanation=Circumstantial evidence is that which would indirectly show culpability. For
example, in the case of a kickback allegation, cash deposits of unknown origin deposited
to the account of the target around the time of the suspect transaction could be
circumstantial evidence.See page(s) 3.102 in the Fraud Examiner's Manual
```

Figure 3: Scrubbed Practice Question

— topic, question stem, options, correct answer, and explanation — as shown in Figure 3. Finally, these files were manually reviewed and edited to ensure proper format for the Agent. The product of this process was a collection of 1,500 text files, one for each practice exam question, and each containing name-value pairs for the relevant question fields named above.

4 CFE Agent Design

The CFE Agent is written in Java. It takes as input a collection of question files and produces as output a sequence of answers which are, in turn, compared against the corresponding correct answers in order to calculate a score. There are four principal elements to the design of the CFE Agent — the CFE manual, the question profile, the algorithm set,

and the selection algorithm.

CFE manual

The Fraud Examiners Manual (known by the CFE Agent as the CFE Manual) is the text corpus on which all of the questions of the CFE Exam are based. It is also the corpus, therefore, on which the CFE Agent executes algorithms to answer exam questions. Fortunately, each question includes a section heading that loosely maps to an individual section within the manual. However, these sections are rough-grained; i.e., relatively large (often 20–100 pages). So, although these headings provide some advantage to the agent, there is still work to be done to shrink the text that is analyzed for each question. Efforts to narrow the scope of the text using information retrieval techniques are currently under way, and are discussed the Future Work section. However, even these rough-grained sections provide significant success in accurately answering questions.

Question Profile

After having reviewed the entire database of questions in the study package, it was determined that there are certain characteristics certain questions share in common. This was thought to be a natural way of partitioning the questions according to a crude ontology each class of which could serve as a means for optimizing an algorithm for the questions of that class. For example, one of the most obvious of these characteristics is whether the question is a true-false question or a multiple choice question. And if the question is true-false, does it include any of the following terms: "always", "never", "must", or "only"? On the other hand, if a question is multiple choice, is its last option, "All of the above"? Some other criteria discovered that are worth noting are whether the question includes options with more than four words in it, whether the question contains the word "except", and whether the question's last option is "None of the above".

Algorithm Set

Various algorithms were developed an incorporated into the model. Furthermore, the system is highly extensible so that new ones yet to be developed can easily be incorporated. The system currently employs nine custom algorithms that utilize natural language processing, information retrieval, and general test-taking techniques. Descriptions of a few example algorithms are given below:

Max Joint Probability Algorithm This algorithm considers only the possible answers, treating each as a sequence of words whose joint probability is to be calculated. It calculates this joint probability for each option by finding the probability of each word among the words of the manual section from which the question was drawn (indicated by topic) and by applying the simplifying assumption of independence between words. The joint probability, then, is simply calculated as the product of the probabilities of the individual words of the option, normalized for the number of words (so that short answers are not over-weighted). The algorithm selects the option with maximum joint probability.

Max Frequency Algorithm The Max Frequency Algorithm simply counts up the total number of times each answer option is encountered in the manual section from which the question was drawn and selects the option whose tally is the largest. Unlike the Max Joine Probability Algorithm, the frequencies are based on occurrences of entire phrases, not on the counts of individual words within them.

False Select This algorithm applies only to true-false questions, simply selecting false, always. Remarkably, this algorithm proved extremely effective for such questions that have "must", "always", "only", or "never" in the stem, achieving an astonishing 78.7% accuracy rate on this class of questions. The remarkable success of such a simple algorithm is an indication of the effectiveness of gaming strategies on certain questions, but also an indication of weaknesses in the way certain questions on this exam are designed.

The Selection Algorithm

The selection algorithm is the procedure for selecting the proper algorithm to apply to a given question. It is based on the experience data collected from executing every algorithm on every question in the study package. For a given question, the CFE Agent chooses the algorithm with the highest accuracy rate for questions with the same profile.

5 CFE Agent Demo

This section outlines a walkthrough of the execution of the CFE Agent for a very short exam, consisting of only four questions. Despite its short length, it should provide a good idea of the basic mechanics of the agent.

Since the CFE Agent is written in java, starting the agent requires invoking the java virtual machine on the main class of the program, appropriately called, CFEAgent. Figure 4, below, shows the startup of the agent. The name of a configuration file is passed in as a command line parameter to the program. This file provides a number of settings for the execution environment, including the mode (interactive vs. batch), the name of the exam file, and log detail level. In this demo, the agent is executing with full detail so that it is easier to understand what is happening under the covers for each question.

The detail logging shows that one of the first things the CFE Agent does when it starts up is construct an internal representation of the Fraud Examiners Manual. This internal representation is in the form of a graph, each of whose nodes represents a section of the document. More specifically, the graph is a tree whose root node represents the entire manual, which in turn possesses links to child nodes each representing one of the four main sections of the manual (Financial Transactions and Fraud Schemes, Law, Investigation, and Fraud Prevention and Deterrence) each of which in turn possesses child nodes representing sub-sections to those sections, and so on. Each node stores information about the section it represents in addition to the text itself, including a hash table for word counts, sub-subheadings, etc. This information facilitates the text analytic computations the agent performs as it answers questions on the exam. It should

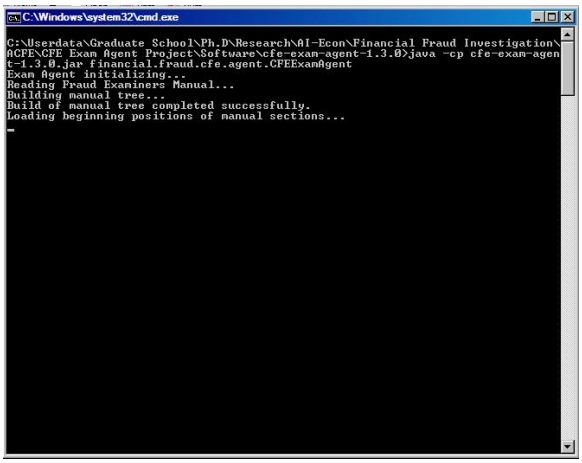


Figure 4: CFE Agent Startup

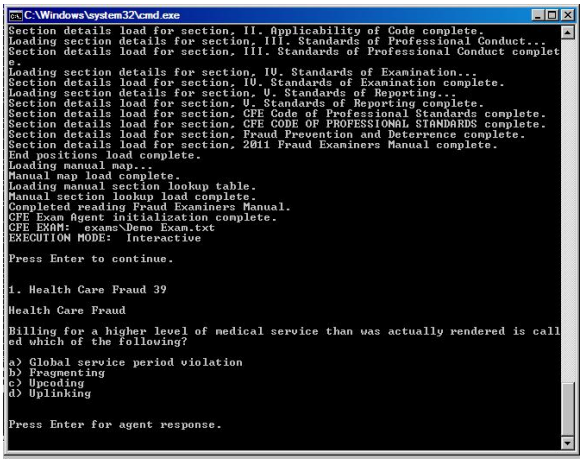


Figure 6: Question 1

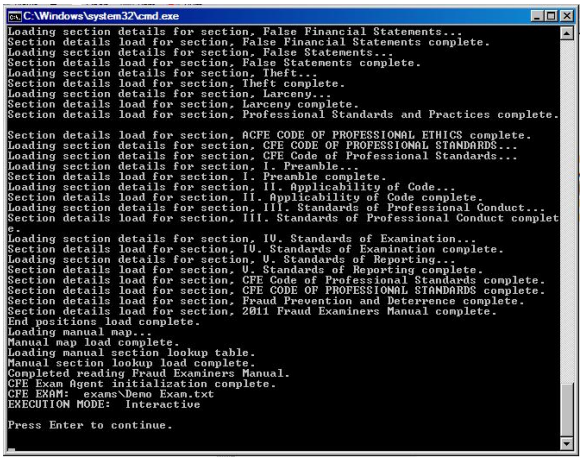


Figure 5: CFE Agent Startup Completed

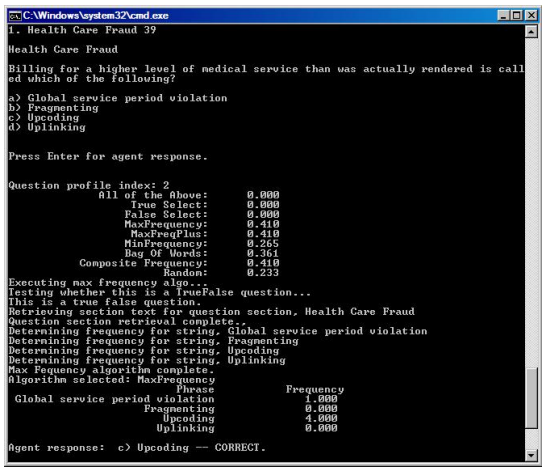


Figure 7: Question 1 Completed

be noted that much of the basis of this tree data structure is based on the items listed in the manuals table of contents, and for even finer-grained detail nodes, on text features within the corpus itself. (That is, lines of text consisting of all capital letters terminating in carriage return generally denote a sub-section of text). Finally, in addition to the tree data structure, the internal representation of the manual includes additional data structures mapping the text of the manual for optimizing access to particular manual subsections.

In Figure 5, the screen shows the completion of the startup sequence of the CFE Agent. First, the CFE Agent gives a line-by-line report showing the nodes loaded into the tree data structure. Then, it shows some additional configuration information showing the exam file, execution mode, and runtime status. At this point, the CFE Agent is ready to begin the exam.

In Figure 6, the first question of the exam is shown along with four possible answers. After the user presses return, the CFE Agent selects the optimal algorithm for that particular question and executes it.

Figure 7 shows log detail for the CFE Agent as it attempts to choose the correct answer for the first question. First, as described in the discussion of the question profile, the agent makes an assignment to a question profile category based on some key features of the question, (whether it's a true-false question, a question at least one of whose options contains more than 4 words, etc.). Once it assesses the question profile, it chooses an algorithm which based on prior experience exhibits maximum accuracy on questions having the same profile. For this question, the profile assignment is category 2, meaning the question has been categorized as a "definition" question essentially one which attempts to test for knowledge of domain-specific vocabulary. The experience data of the Agent indicates the best performing algorithm is the Max Frequency algorithm, which when applied for this question results in the selection of choice c, the correct answer.

Figure 8 shows the CFE Agent answer the second question. This question appears to be similar to the first in that

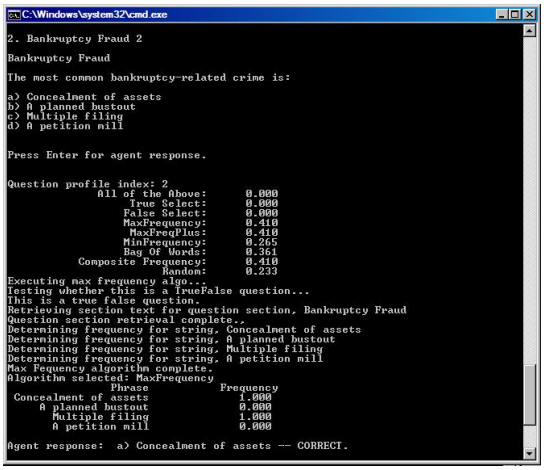


Figure 8: Question 2

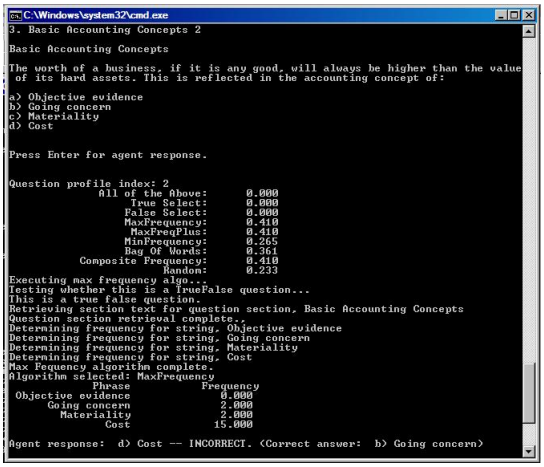


Figure 9: Question 3

it also earns a category assignment of 2, meaning it is another "definition" question. By the same experience-based reasoning as in the first question, the agent applies the Max Frequency algorithm and selects answer a, the correct answer.

Figure 9 shows an attempt at the third question. However, this time the agent is not successful at choosing the right answer. Again, it assigns this question to the same category as the two prior questions and uses the same Max Frequency algorithm, but it is led astray by the fact that the fourth option, "cost" is a common word whose frequency in the section corpus is over-weighted relative to the other options for this question. This is an example where the current level of sophistication of the agent is insufficient to correctly answer questions one of the options may consists of a word or phrase that might be over-represented in the corpus. Further work will need to be done to develop the natural language processing algorithm to compensate for this over-weighting, while preserving its relative success on other questions in the same category.

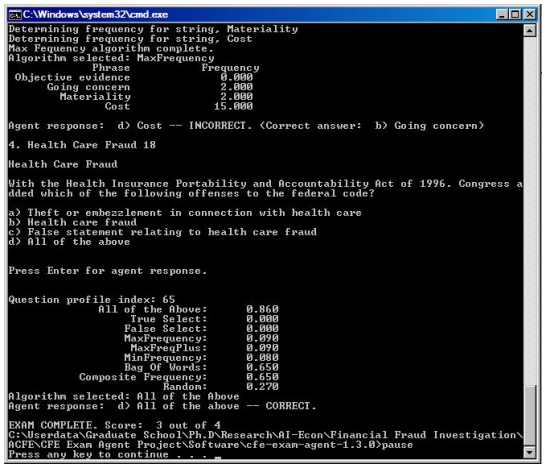
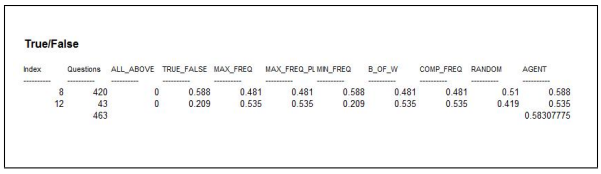


Figure 10: Question 4 and Termination

Figure 10 shows the final question, which is one of a different type — an "All of the Above" question. Here, the CFE Agent chooses an algorithm appropriately called "All of the Above" because it simply selects "All of the Above" whenever its an option. (As shown in the data table, this algorithm has a remarkable 86% success rate, lending one to question certain aspects of the CFE Exams design, as mentioned earlier.) The agent applies this algorithm and gets the answer correct. Finally, the agent terminates.

6 CFE Agent Results

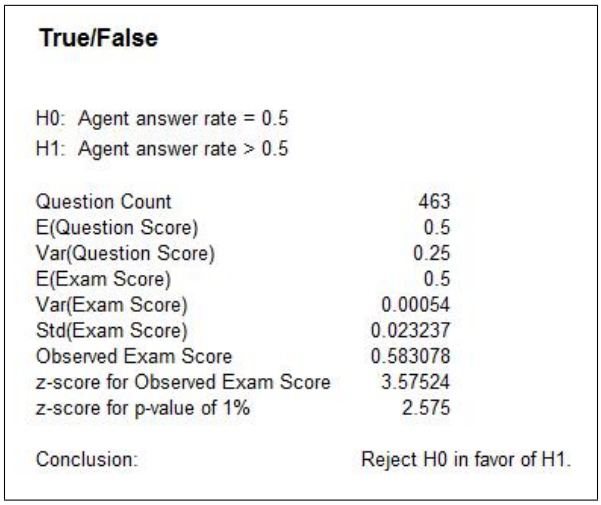
The goal of this project is for the CFE Agent to pass the test. In particular, achieve 75% accuracy on each of the four sections. After having the agent take an exam consisting of all 1,500 questions in the battery included with the study package, the results were gathered, analyzed, and tabulated. At the current time, the agent does not pass the exam. However, the agent demonstrates promising statitstically significant performance. Consider the question of whether the agent actually performed any better than random guessing. For true-false questions, random guessing would be expected to yield an accuracy rate on the of approximately 50%. (Given there are a relatively large number (463) of true-false questions, it is reasonable to assume the experimental accuracy rate should be very close to this theoretical value.) Likewise, for multiple choice questions with four options, random guessing should yield an accuracy rate of approximately 25%. (Again, for the same reason as for true-false questions, the experimental accuracy should be close to the theoretical expected value.) However, the empirical data shown in Figure 12 and Figure 14 indicates that for true-false questions, the observed accuracy rate is over 58%, giving a z-score of over 3.575 (the z-score for a p-value of 1% is 2.575), and that for multiple choice questions, the accuracy rate is just over 49%, giving a z-score of 16.67. For both classes of questions, then, the null hypothesis of performance consistent with random guessing is rejected by a large margin at the 99% significance level. This is considered an important success on which to build further research



True/False

Index	Questions	ALL_ABOVE	TRUE_FALSE	MAX_FREQ	MAX_FREQ_PL	MIN_FREQ	B_OF_W	COMP_FREQ	RANDOM	AGENT
8	420	0	0.588	0.481	0.481	0.588	0.481	0.481	0.51	0.588
12	43	0	0.209	0.535	0.535	0.209	0.535	0.535	0.419	0.535
	463									0.58307775

Figure 11: Results for True-False Questions

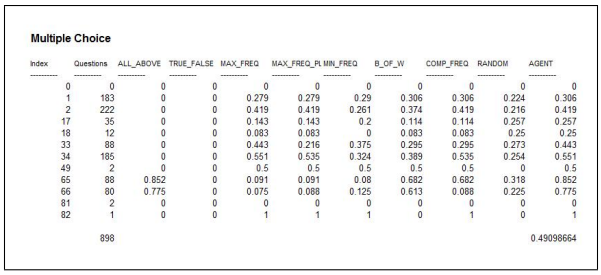


True/False

H0: Agent answer rate = 0.5
H1: Agent answer rate > 0.5

Question Count	463
E(Question Score)	0.5
Var(Question Score)	0.25
E(Exam Score)	0.5
Var(Exam Score)	0.00054
Std(Exam Score)	0.023237
Observed Exam Score	0.583078
z-score for Observed Exam Score	3.57524
z-score for p-value of 1%	2.575
Conclusion:	Reject H0 in favor of H1.

Figure 12: Hypothesis Test for True-False Questions



Multiple Choice

Index	Questions	ALL_ABOVE	TRUE_FALSE	MAX_FREQ	MAX_FREQ_PL	MIN_FREQ	B_OF_W	COMP_FREQ	RANDOM	AGENT
0	0	0	0	0	0	0	0	0	0	0
1	183	0	0	0.279	0.279	0.29	0.306	0.306	0.224	0.306
2	222	0	0	0.419	0.419	0.261	0.374	0.419	0.216	0.419
17	35	0	0	0.143	0.143	0.2	0.114	0.114	0.257	0.257
18	12	0	0	0.083	0.083	0	0.083	0.083	0.25	0.25
33	88	0	0	0.443	0.216	0.375	0.295	0.295	0.273	0.443
34	185	0	0	0.551	0.535	0.324	0.389	0.535	0.254	0.551
49	2	0	0	0.5	0.5	0.5	0.5	0.5	0	0.5
65	88	0.852	0	0.091	0.091	0.08	0.682	0.682	0.318	0.852
66	80	0.775	0	0.075	0.088	0.125	0.613	0.088	0.225	0.775
81	2	0	0	0	0	0	0	0	0	0
82	1	0	0	1	1	1	0	1	0	1
	898									0.49098664

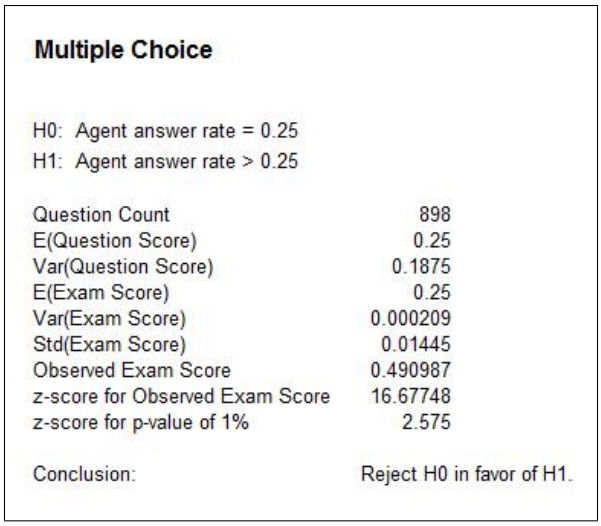
Figure 13: Results for Multiple Choice Questions

in the future.

7 Future Work

The goal in this project was to see the level of performance that could be achieved with only very simple text mining approaches. The idea was to set a baseline against which a more advanced agent, using more sophisticated techniques, could be compared to determine the relative value of such techniques. In this initial version of the CFE Agent, the algorithms are based on a very simple information retrieval algorithm — a simple mapping between a topic label provided with each question and a relatively broad section in the Fraud Examiners manual. This algorithm does not make full use of the information provided by the manual tree data structure whose nodes break the manual down to a finer level of detail.

A main focus of future research is to develop this algorithm to leverage compositional semantics i.e., lexical aspects of the text sections, subsections, and sub-subsections, and so on, from which higher levels of performance could be achieved. In fact, this research is already under way. Attempts to apply the information retrieval algorithms, jaccard coefficient and tf-idf (term frequency weighting, inverted document frequency weighting), have yielded promising results for reliably zeroing in on narrow, correct target subsections in the manual.



Multiple Choice

H0: Agent answer rate = 0.25
H1: Agent answer rate > 0.25

Question Count	898
E(Question Score)	0.25
Var(Question Score)	0.1875
E(Exam Score)	0.25
Var(Exam Score)	0.000209
Std(Exam Score)	0.01445
Observed Exam Score	0.490987
z-score for Observed Exam Score	16.67748
z-score for p-value of 1%	2.575
Conclusion:	Reject H0 in favor of H1.

Figure 14: Hypothesis Test for Multiple Choice Questions

Machine learning may also be utilized to further fine tune the information retrieval algorithm. It is true that in some cases the application of the information retrieval algorithms listed above may not yield the desired subsections of the manual. In these cases, machine learning can be of great help, and will thus be applied in this project as needed.

Also, certain aspects of computational semantics may be used in instances where text is semi-structured, such as in itemized lists found through automated means in the manual, and in practice battery questions that could be used to build semantic representations of assertions in the domain.

References

Bringsjord, S. & Ferrucci, D. 2000. *Artificial Intelligence and Literary Creativity: Inside the Mind of Brutus, a Storytelling Machine*. Mahwah, New Jersey: Lawrence Erlbaum.

Bringsjord, S. & Schimanski, S. 2004. What is Artificial Intelligence? Psychometric AI as an Answer. *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 03)* 887–893.

Evans, G. 1968. A Program for the Solution of a Class of Geometric-Analogy Intelligence-Test Questions. *Semantic Information Processing* 271–353.

Raven, J. C. 1962. *Advanced Progressive Matrices Set II*. London, United Kingdom: H. K. Lewis.

Torrance, E. P. 1990. *The Torrance Tests of Creative Thinking*. Bensenville, Illinois: Scholastic Testing Service.