BioWorldParser: A Suite of Parsers for Leveraging Educational Data Mining Techniques

Tenzin Doleck[1], Ram B. Basnet[2], Eric Poitras[3], Susanne Lajoie[1]
[1]McGill University, Montreal, Canada
{tenzin.doleck, susanne.lajoie}@mcgill.ca
[2]Colorado Mesa University, Grand Junction, Colorado
rbasnet@coloradomesa.edu
[3]University of Utah, Salt Lake City, Utah
assistlaboratory@gmail.com

Abstract— **There has been a dramatic expansion in both the amount of available large-scale educational databases and educational mining techniques. Educational data mining has been a fertile subject of research in recent times; further, the use of educational data mining has become popular among both researchers and practitioners. Log files generated by computer-based learning environments like Intelligent Tutoring Systems contain a wealth of information about learner behaviors that characterize academic success. There is growing interest in mining these data sources for knowledge-based discovery to reveal relevant, meaningful, and useful educational information to illuminate our understanding of learners' behaviors and outcomes. All too often however, extracting the pertinent information from the data to leverage the data mining techniques can be a major roadblock; for example, the asynchronous nature of the data logged in computer-based learning environments and data mining tools pose several challenges for mining data. We sought to mitigate this by developing a parser for the BioWorld System. In this paper, we explore the viability of a hand-coded parser by presenting BioWorldParser (a suite of scripts), which was developed to parse and retrieve data from raw log files generated by the BioWorld system, to help leverage educational data mining techniques in the context of an Intelligent Tutoring System for the medical domain.**

Index Terms— data mining, parsers, machine learning, medical education, computer-based learning environments, clinical reasoning, intelligent tutoring systems.

I. INTRODUCTION

We live in a data rich world. There has been a proliferation in the use of educational data mining to facilitate and augment education research [1, 2]; concomitantly there has been an increase in the availability of a myriad of data mining tools and techniques to analyze users' learning behaviors in technology rich environments. However, the ability to take advantage of the affordances of data mining techniques is complicated by the fact that before one can leverage the power of data mining solutions, extracting the pertinent information from the data (input file) presents challenges. Although there are many popular parser generators available -Antlr [3], Bison [4], and Lemon [5] to name a few- to extract data, we developed our own parser to have greater flexibility and control that hand-coded parsers afford coupled with the fact that it helps mitigate constraints imposed by parser generators. Furthermore, hand-coded parsers are especially useful in providing a better error-handling mechanism and in ameliorating the debugging process. In the BioWorld learning system, server log files are used for logging learner actions. The log files constitute a wealth of information about how learners reason as they solve virtual patient cases. In order to mine the data, it is crucial to extract and process the data. In this spirit, we propose and develop BioWorldParser (a suite of scripts) for the BioWorld system that would afford us the flexibility to parse the system logs and generate desired input data for conducting data mining experiments.

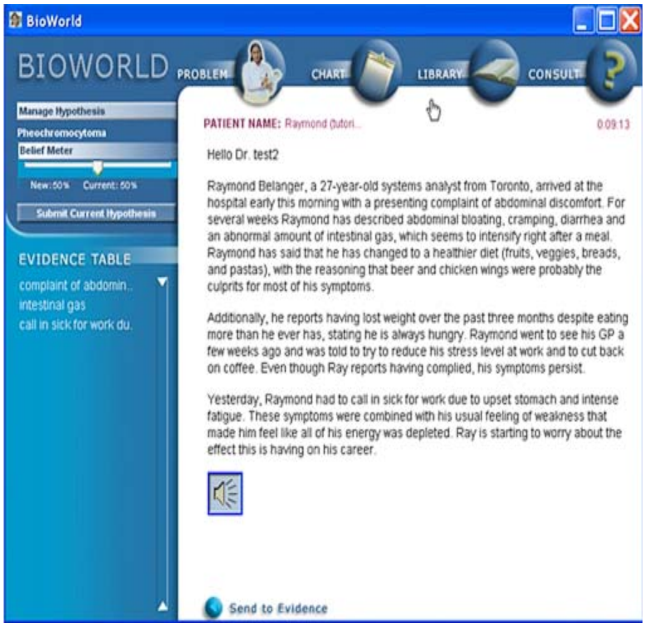
Research on mining data has shown to provide important and useful information for a large number of practical applications. We have recently explored the use of data mining techniques [6, 7, 8] for a number of tasks, such as diagnosis correctness, examining help-seeking behaviors, and augmenting the novice-expert overlay model in the BioWorld system. As we expand our reach into exploring new and powerful data mining techniques, the BioWorldParser will be especially helpful in facilitating our future endeavors. We implement BioWorldParser as a Python package, and demonstrate its utility via a use case in generating sequence data.

II. THE CONTEXT: BIOWORLD, AN INTELLIGENT TUTORING SYSTEM

BioWorld (figure 1) is a Medical Intelligent Tutoring System that is designed to support novice physicians in practicing diagnostic reasoning skills while receiving feedback [9, 10]. The system was created using a cognitive apprenticeship framework [11] where learners practice realistic clinical reasoning tasks and are scaffolded in the context of their learning with expert models. In BioWorld, novice physicians learn clinical reasoning by diagnosing virtual patient cases by identifying relevant symptoms, generating likely diagnosis, ordering lab-tests, and reasoning about the nature of the underlying disease [10]. The learning system has tools embedded that support the cognitive and metacognitive activities that mediate performance in diagnosing virtual patient cases. The user-system interactions are captured by the logging system; examples of such interactions include adding evidence items, ordering lab tests, etc. The learning session ends with the user submitting the final diagnosis, justifying and

978-1-4799-6876-3/14/$31.00 ©2014 IEEE

prioritizing their selection (by sorting the evidence), and finally receiving individualized feedback on their submitted solution.



complaint of abdomin..
intestinal gas
call in sick for work du.

PATIENT NAME: Raymond (tutori... 0:09:13

Hello Dr. test2

Raymond Belanger, a 27-year-old systems analyst from Toronto, arrived at the hospital early this morning with a presenting complaint of abdominal discomfort. For several weeks Raymond has described abdominal bloating, cramping, diarrhea and an abnormal amount of intestinal gas, which seems to intensify right after a meal. Raymond has said that he has changed to a healthier diet (fruits, veggies, breads, and pastas), with the reasoning that beer and chicken wings were probably the culprits for most of his symptoms.

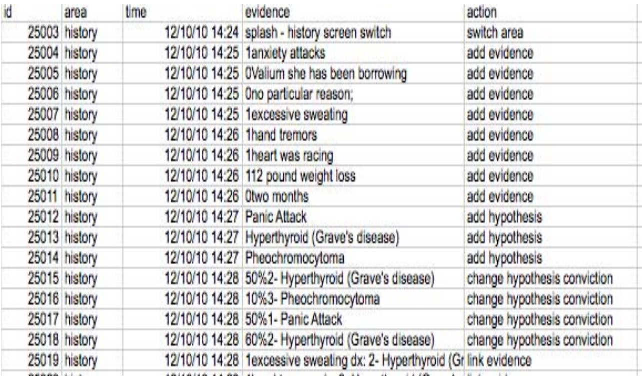
Additionally, he reports having lost weight over the past three months despite eating more than he ever has, stating he is always hungry. Raymond went to see his GP a few weeks ago and was told to try to reduce his stress level at work and to cut back on coffee. Even though Ray reports having complied, his symptoms persist.

Yesterday, Raymond had to call in sick for work due to upset stomach and intense fatigue. These symptoms were combined with his usual feeling of weakness that made him feel like all of his energy was depleted. Ray is starting to worry about the effect this is having on his career.

Fig. 1. BioWorld Interface

III. LOG-FILE

The BioWorld system tracks and logs user actions in log files. For each action a learner performs, one line representing such action is logged. Three types of performance metrics are in the log files (figure 2), namely, diagnostic efficacy (e.g., accuracy, count of matches with experts, and percentage of matches with experts), efficiency (e.g., number of tests ordered and time to solve the case), and affect (e.g., confidence). Information saved in the log-file includes the attempt identifier (participant and case ID), a timestamp, the BioWorld space (e.g., chart), the specific action taken (e.g., add test), and details in relation to the action (e.g., Thyroid Stimulating Hormone (TSH) Result: 0.2 mU/L). In order to mine information from the raw log files, the data has to be preprocessed; parsers are especially useful in aiding the preprocessing activity by providing a means to draw out required data in the desirable format.



id	area	time	evidence	action
25003	history	12/10/10 14:24	splash - history screen switch	switch area
25004	history	12/10/10 14:25	1anxiety attacks	add evidence
25005	history	12/10/10 14:25	0Valium she has been borrowing	add evidence
25006	history	12/10/10 14:25	0no particular reason;	add evidence
25007	history	12/10/10 14:25	1excessive sweating	add evidence
25008	history	12/10/10 14:26	1hand tremors	add evidence
25009	history	12/10/10 14:26	1heart was racing	add evidence
25010	history	12/10/10 14:26	112 pound weight loss	add evidence
25011	history	12/10/10 14:26	0two months	add evidence
25012	history	12/10/10 14:27	Panic Attack	add hypothesis
25013	history	12/10/10 14:27	Hyperthyroid (Grave's disease)	add hypothesis
25014	history	12/10/10 14:27	Pheochromocytoma	add hypothesis
25015	history	12/10/10 14:28	50%2- Hyperthyroid (Grave's disease)	change hypothesis conviction
25016	history	12/10/10 14:28	10%3- Pheochromocytoma	change hypothesis conviction
25017	history	12/10/10 14:28	50%1- Panic Attack	change hypothesis conviction
25018	history	12/10/10 14:28	60%2- Hyperthyroid (Grave's disease)	change hypothesis conviction
25019	history	12/10/10 14:28	1excessive sweating dx: 2- Hyperthyroid (Gr	link evidence

Fig. 2. Snapshot of log-file generated by BioWorld

IV. PARSER

The role of a parser in a general knowledge-discovery process is highlighted in figure 3. To facilitate the preprocessing steps in data mining, we have developed a parser, BioWorldParser (a suite of scripts) to tailor data extraction and processing for use in data mining. BioWorldParser, coded in Python, comprises a suite of scripts for accomplishing various data extraction tasks. The python scripts afford a couple of advantages, namely, platform independency (can be run on any platform) and simplicity (run as a simple command line tool). In this section we present the algorithm and a snapshot of the code (figure 4) for the parser (User Actions sequence Generation).

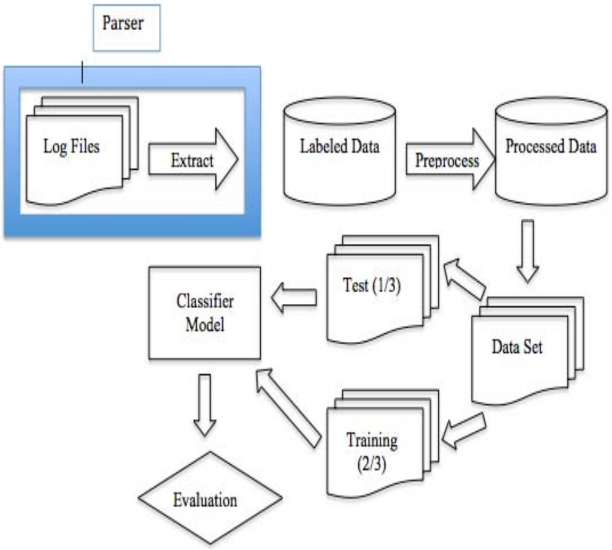


Fig. 3. The role of a parser in a general knowledge-discovery process

GenerateSeqData.py *User Actions Sequence Generation*

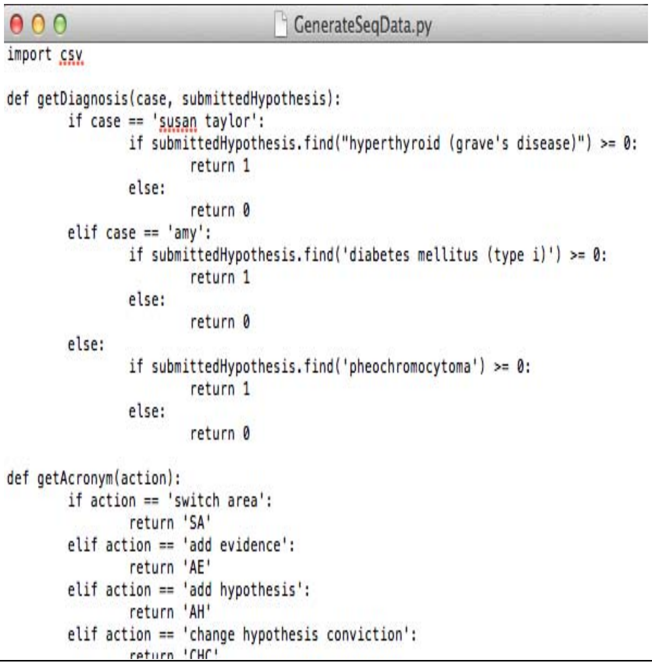
```
Input: BioWorldLog.xls
Output: HMMAmyPhase1.txt
        HMMCynthiaPhase1.txt
        HMMSusanTaylorPhase1.txt

1: Convert BWlog.xls into BWlog.csv file for easier parsing.
2: Open the converted BWlog.csv file using Python's standard csv library
3: Open all the required files in output mode such as HMMAmy.txt, HMMSusanTaylor.txt, and HMMCynthia.txt to write the results of parser.
4. For each row of data, until the end of file:
i. The first line contains the title for each column, so ignore it.
ii. Convert each column data into lower case for easy comparison.
iii. If the evidence column equals 'splash - history screen switch', reset the variables to collect fresh data. 'splash -
```

history screen switch' value is logged every time a user enters into case diagnosis mode.

iv. Else if the evidence column equals 'summarize - expert screen switch', write data to corresponding file based on the case the user currently is working on.

v. If the action column equals 'submit hypothesis', compare the user submitted hypothesis with the correct hypothesis and record CORRECT or INCORRECT for the result.

vi. Else if the action equals 'abort submit hypothesis', continue collecting more actions for this case.

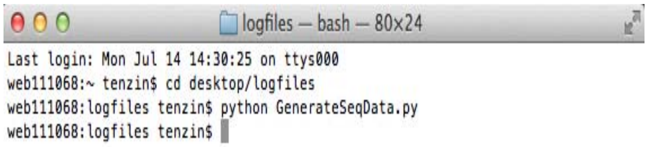


```
import csv

def getDiagnosis(case, submittedHypothesis):
        if case == 'susan taylor':
                if submittedHypothesis.find("hyperthyroid (grave's disease)") >= 0:
                        return 1
                else:
                        return 0
        elif case == 'amy':
                if submittedHypothesis.find('diabetes mellitus (type i)') >= 0:
                        return 1
                else:
                        return 0
        else:
                if submittedHypothesis.find('pheochromocytoma') >= 0:
                        return 1
                else:
                        return 0

def getAcronym(action):
        if action == 'switch area':
                return 'SA'
        elif action == 'add evidence':
                return 'AE'
        elif action == 'add hypothesis':
                return 'AH'
        elif action == 'change hypothesis conviction':
                return 'CHC'
```

Fig. 4. Parser Algorithm & Code (snapshot)

V. Example Use Case

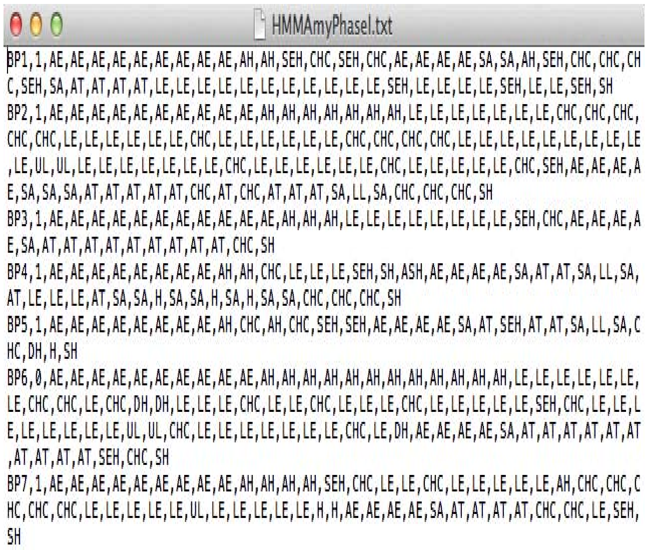
In this section, we demonstrate the functionality of the parser with an example. Our goal was to develop scripts that would best help us exploit the multitudes of data mining tools and techniques. The script can be run in the Terminal (as shown in Figure 5). The script takes the log file (generated by the BioWorld system) as input and generates the sequence data in text format.



```
Last login: Mon Jul 14 14:30:25 on ttys000
web111068:~ tenzin$ cd desktop/logfiles
web111068:logfiles tenzin$ python GenerateSeqData.py
web111068:logfiles tenzin$ 
```

Fig. 5. Running the Script

After running the script (GenerateSeqData.py), the code generates output files with sequence data (as shown in figure 6). We can now use various data mining tools and techniques on this newly generated sequence data. For example, various data mining techniques such as sequence mining or Hidden Markov Model (HMM) analysis can now be applied on this sequence data.



```
BP1,1,AE,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,SEH,CHC,SEH,CHC,AE,AE,AE,AE,SA,SA,AH,SEH,CHC,CHC,CH
C,SEH,SA,AT,AT,AT,AT,LE,LE,LE,LE,LE,LE,LE,LE,LE,LE,LE,SEH,LE,LE,LE,LE,SEH,LE,LE,SEH,SH
BP2,1,AE,AE,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,AH,AH,AH,AH,AH,LE,LE,LE,LE,LE,LE,LE,CHC,CHC,CHC,
CHC,CHC,LE,LE,LE,LE,LE,LE,CHC,LE,LE,LE,LE,LE,LE,CHC,CHC,CHC,CHC,LE,LE,LE,LE,LE,LE,LE,LE,LE
,LE,UL,UL,LE,LE,LE,LE,LE,LE,LE,CHC,LE,LE,LE,LE,LE,LE,CHC,LE,LE,LE,LE,LE,CHC,SEH,AE,AE,AE,A
E,SA,SA,SA,AT,AT,AT,AT,AT,CHC,AT,CHC,AT,AT,AT,SA,LL,SA,CHC,CHC,CHC,SH
BP3,1,AE,AE,AE,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,AH,LE,LE,LE,LE,LE,LE,LE,LE,SEH,CHC,AE,AE,AE,A
E,SA,AT,AT,AT,AT,AT,AT,AT,AT,AT,CHC,SH
BP4,1,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,CHC,LE,LE,LE,SEH,SH,ASH,AE,AE,AE,AE,SA,AT,AT,SA,LL,SA,
AT,LE,LE,LE,AT,SA,SA,H,SA,SA,H,SA,H,SA,SA,CHC,CHC,CHC,SH
BP5,1,AE,AE,AE,AE,AE,AE,AE,AE,AH,CHC,AH,CHC,SEH,SEH,AE,AE,AE,AE,SA,AT,SEH,AT,AT,SA,LL,SA,C
HC,DH,H,SH
BP6,0,AE,AE,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,AH,AH,AH,AH,AH,AH,AH,AH,AH,AH,LE,LE,LE,LE,LE,LE,
LE,CHC,CHC,LE,CHC,DH,DH,LE,LE,LE,CHC,LE,LE,CHC,LE,LE,LE,CHC,LE,LE,LE,LE,LE,SEH,CHC,LE,LE,L
E,LE,LE,LE,LE,LE,UL,UL,CHC,LE,LE,LE,LE,LE,LE,LE,CHC,LE,DH,AE,AE,AE,AE,SA,AT,AT,AT,AT,AT,AT
,AT,AT,AT,AT,SEH,CHC,SH
BP7,1,AE,AE,AE,AE,AE,AE,AE,AE,AE,AH,AH,AH,AH,SEH,CHC,LE,LE,CHC,LE,LE,LE,LE,LE,AH,CHC,CHC,C
HC,CHC,CHC,LE,LE,LE,LE,LE,UL,LE,LE,LE,LE,LE,H,H,AE,AE,AE,AE,SA,AT,AT,AT,AT,CHC,CHC,LE,SEH,
SH
```

Fig. 6. Text file generated by the Parser

VI. Conclusion

Recently, we have witnessed the rise of educational data mining as a field, and as it does, researchers must have the appropriate tools to take advantage of the affordances of data mining tools and techniques. In this paper, we investigate the viability of a hand-coded parser by presenting and illustrating the use of BioWorldParser to leverage the continued growth in data mining capabilities. Instead of resorting to using parser generators, we have proposed and developed the BioWorldParser to afford greater ease of use and flexibility. We have implemented the parser, as a platform independent command line tool in Python. The hand-coded BioWorldParser is both an interesting exercise in development as well as a highly useful application. We are eager to explore a number of other avenues for future direction by expanding and further developing this suite of tools for a range of tasks.

References

[1] R.S.J.D. Baker and K. Yacef, “The state of educational data mining in 2009: A review and future visions,” Journal of Educational Data Mining, vol. 1, no. 1, pp. 3-17, 2009.

[2] C. R. Romero and S. Ventura, “Educational Data Mining: A Review of the State of the Art,” IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, vol.40, no. 6, pp. 601–618, 2010.

[3] Antlr. (n.d.). Antlr. Retrieved June 14, 2014, from http://www.antlr.org/

[4] Bison. (n.d.). Bison. Retrieved June 14, 2014, from http://www.gnu.org/software/bison/

[5] Lemon. (n.d.). Lemon. Retrieved June 14, 2014, from http://www.hwaci.com/sw/lemon/

[6] E. Poitras, T. Doleck, and S. Lajoie, “Mining Case Summaries in BioWorld,” In Proceedings of International Conference on Computer Science & Education (ICCSE 2014), Vancouver, Canada, Aug, 2014.

[7] E. Poitras, A. Jarell, T. Doleck, and S. Lajoie, “Supporting Diagnostic Reasoning by Modeling Help-Seeking,” In Proceedings of International Conference on Computer Science & Education (ICCSE 2014), Vancouver, Canada, Aug, 2014.

[8] S. P. Lajoie, E. G. Poitras, T. Doleck, and A. Jarell, “Modeling Metacognitive Activities in Medical Problem-Solving with BioWorld,” In Peña-Ayala (Ed.), Metacognition: Fundamentals, Applications, and Trends. Springer Series: Intelligent Systems Reference Library, 2015.

[9] S. P. Lajoie, L. Naismith, E. Poitras, Y.-J. Hong, I. Cruz-Panesso, J. Ranellucci, S. Mamane, and J. Wiseman, “Technology-rich tools to support self-regulated learning and performance in medicine,” in International Handbook of Metacognition and Learning Technologies, vol. 28, R. Azevedo and V. Aleven, Eds. New York: Springer, 2013, pp. 229-242.

[10] S. P. Lajoie, “Developing professional expertise with a cognitive apprenticeship model: Examples from Avionics and Medicine,” in Development of professional expertise: Toward measurement of expert performance and design of optimal learning environments, K. A. Ericsson, Ed. Cambridge UK: Cambridge University Press, 2009, pp. 61-83.

[11] A. Collins, “Cognitive apprenticeship,” in Cambridge Handbook of the Learning Sciences, R. K. Sawyer, Ed. Cambridge UK: Cambridge University Press, 2006, pp. 47-60.