2005

Hands-On NLP for an Interdisciplinary Audience

Elizabeth D. Liddy
Syracuse University, liddy@syr.edu

Nancy McCracken
Syracuse University, njmccrac@syr.edu

Hands-On NLP for an Interdisciplinary Audience

Elizabeth D. Liddy and Nancy J. McCracken
Center for Natural Language Processing
School of Information Studies
Syracuse University
liddy@syr.edu, njm@ecs.syr.edu

Abstract

The need for a single NLP offering for a diverse mix of graduate students (including computer scientists, information scientists, and linguists) has motivated us to develop a course that provides students with a breadth of understanding of the scope of real world applications, as well as depth of knowledge of the computational techniques on which to build in later experiences. We describe the three hands-on tasks for the course that have proven successful, namely: 1) in-class group simulations of computational processes; 2) team posters and public presentations on state-of-the-art commercial NLP applications, and; 3) team projects implementing various levels of human language processing using open-source software on large textual collections. Methods of evaluation and indicators of success are also described.

1 Introduction

This paper presents both an overview and some of the details regarding audience, assignments, technology, and projects in an interdisciplinary course on Natural Language Processing that has evolved over time and been successful along multiple dimensions – both from the students' and the faculty's perspective in terms of accomplishments and enjoyment. This success has required us to meet the challenges of enabling students from a range of disciplines and diverse experience to each gain a real understanding of what is entailed in Natural Language Processing.

2 A Course Within Multiple Curricula

The course is entitled Natural Language Processing and is taught at the 600 graduate course level in a School of Information Studies in a mid to large-size private university. While NLP is not core to any of the three graduate degree programs in the Information School, it is considered an important area within the Information School for both professional careers and advanced research, as well as in the Computer Science and Linguistic Programs on campus. The course has been taught every 1½ to 2 years for the last 18 years. While some aspects of the course have changed dramatically, particularly in regards to the nature of the student team projects, the basic structure – the six levels of language processing – has remained essentially the same, with updates to topics within these levels reflecting recent research findings and new applications.

3 Audience

At the moment, this is the only course offering on NLP within the university, but a second-level, seminar course, entitled Content Analysis Research Using Natural Language Processing, geared towards PhD students doing social science research on large textual data sets, will be offered for the first time in Fall 2005. Given that the current NLP course is the only one taught, it cannot, by necessity, have the depth that could be achieved in curricula where there are multiple courses. In a more extensive curriculum, courses provide a greater depth than is possible in our single course. Our goal is to provide students with a solid, broad basis on which to build in later experiences, and to en-

able real understanding of a complex topic for which students realize there is a much greater depth of understanding that could be reached.

The disciplinary mix of students in the course is usually an even mix of information science and computer science students, with slightly fewer linguistics majors. Recently the Linguistics Department has established a concentration in Information Representation and Retrieval, for which the NLP course is a required course. Also, the course is cross-listed as an elective for computer science graduate students. All of the above facts contribute to the widely diverse mix of students in the NLP course, and has required us to develop a curriculum that enables all students to be successful in achieving solid competency in NLP.

4 Topics Covered

The topics in the course include typical ones covered in most NLP courses and are organized around the levels of language processing and the specific computational techniques within each of these. Discussions of more general theoretic notions such as statistical vs. symbolic NLP, representation theories, and language modeling are interspersed. A single example of topics that are taught within the levels of language processing include:

>Morphology - Finite state automata
>Lexicology - Part-of-speech tagging
>Syntax - Parsing with context free grammars
>Semantics - Word sense disambiguation
>Discourse - Sublanguage analysis
>Pragmatics - Gricean Maxims

Each of the topics has assigned readings, from the course's textbook, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* by Daniel Jurafsky & James H. Martin, as well as from recent and seminal papers.

5 Methods

What really enables the students to fully grasp the content of the course are the three important hands-on features of the course, namely:

1. Small, in-class group simulations of computational processes.

2. Team posters and public presentations reporting on the state-of-the-art in commercial NLP applications such as summarization, text mining, machine translation, question answering, speech recognition, and natural language generation.
3. Team projects implementing various levels of human language processing using open-source software on large collections.

Each of these features of the course is described in some detail in the following sections.

The course is designed around group projects, while the membership of the teams changes for each assignment. This is key to enabling a diverse group to learn to work with students from different disciplines and to value divergent experience. It has also proven extremely successful in forming a class that thinks of itself as a community and in encouraging sharing of best practices so that everyone advances their learning significantly further than if working alone or with the same team throughout the course. The way that teams are formed for the three types of projects varies, and will be described in each of the following three sections.

Furthermore, constant, frequent presentations to the class of the group work, no matter how brief, enable students to own their newly-gained understandings. In fact, this course no longer requires any written papers, but instead focuses on application of what is learned, first at the specific level of language processing, then to new data for new purposes, and then, to understanding real-world NLP systems performing various applications – with the group constantly reporting their findings back to the class.

5.1 In-class Group Simulations of Computational Processes

During the first third of the course, lectures on each level of language processing are followed by a 30 to 45 minute exercise that enables the students who work in small groups to simulate the process they have just learned about, i.e. morphological analysis, part-of-speech tagging, or parsing some sample sentences with a small grammar. These groups are formed by the professor in an ad hoc manner by counting off by 4 in a different pattern each week to ensure that students work with stu-

dents on the other side of the room, given that friends or students from the same school tend to sit together. After the exercise, each group has 5 minutes to report back to the class on how they approached the task, with visuals.

We've found that the formation of these small groups is pedagogically sound and enables learning in three ways. First, the groups break down social barriers and as the course advances the students find it much easier to work together and are more comfortable in sharing their work. Secondly, the students begin to understand and value what the students from different disciplines bring to bear on NLP problems. That is, the computer scientists recognize the value of the deeper understanding of language of the linguistic students, and the linguistic students learn how the computer science students approach the task computationally. Thirdly, while there were concerns on our part that these simulations might be too easy, the students have affirmed in mid-term course evaluations (which are not required, but do provide invaluable insight into a class's engagement with and assimilation of the material) that these simulations really help them to understand conceptually what the task is and how it might be accomplished before they have to automate the processes.

5.2   Real World Applications of NLP

This year, two semester-long team projects were assigned – the usual team-based computer implementation of NLP for a particular computational task – and an investigation into how NLP is utilized in various state-of-the-art commercial NLP applications. The motivation for adding this second semester-long team project was that a number of the students in the course, particularly the masters students in Information Management, are most likely to encounter NLP in their work world when they need to advise on particular language-based applications. It has become clear, however, that as a result of this assignment, all of the students are quite pleased with their own improved ability to understand what a language-based technology is actually doing. Even if a student is more research-focused, they are intrigued by what might be done to improve or add to a particular technology.

Students are given two weeks to familiarize themselves outside of class with the suggested applications sufficiently to select a topic of real interest to them. This year's choices included Spell Correction, Machine Translation, Search Engines, Text Mining, Summarization, Question Answering, Speech Recognition, Cross-Language Information Retrieval, Natural Language Generation, and Dialogue Agents.

Students then sign up, on a first-come basis, for their preferred application. The teams are kept small (up to four) to ensure that each student contributes. At times a single student is sufficiently interested in a topic that a team of one is formed. Students arrange their own division of labor. There are three 10 to 20 minute report-backs by each team over the course of the semester, the first two to the class and the final one during an open invitation, school-wide Poster & Reception event. There are guidelines for each of the three presentations, as well as a stated expectation that the teams actively critique and comment on the presentations, both in terms of the information presented as well as presentational factors. Five minutes are allowed for class comments and students are graded on how actively they participate and provide feedback.

The 1st presentation is a non-technical overview of what the particular NLP application does and includes examples of publicly available systems / products the class might know. The 2nd presentation covers technical details of the application, concentrating on the computational linguistic aspects, particularly how such an application typically works, and the levels of NL processing that are involved (e.g., lexical, syntactic, etc). The 3rd presentation involves a poster which incorporates the best of their first two presentations and suggestions from the class, plus a laptop demo if possible.

As stated above, the 3rd presentation is done in an open school-wide Poster and Reception event which is attended by faculty and students, mainly PhD students. The Poster Receptions have proven very successful along multiple dimensions – first, the students take great pride in the work they are presenting; second, posters are better than one-time, in-class presentations as the multiple opportunities to explain their work and get feedback improve the students' ability to create the best presentation of their work; third, the wider exposure of the field and its applications builds an audience for future semesters and instills in the student body a sense of the reach and importance of NLP.

5.3   Hands-On NL Processing of Text

The second of the semester-long team projects is the computer implementation of NLP. The goal of the project is for students to gain hands-on experience in utilizing NLP software in the context of accomplishing analysis of a large, real-world data set. The project comprises two tasks, each of which is reported back to the class by each team. These presentations were not initially in the syllabus, but interestingly, the students requested that each team present after each task so that they could all learn from the experiences of the other teams.

The corpus chosen was the publicly available Enron email data set, which consists of about 250,000 unique emails from 150 people. With duplication, the data has approximately 500,000 files and takes up 2.75 gigabytes. The data set was prepared for public release by William Cohen at CMU and, available at http://www-2.cs.cmu.edu/~enron/. This data set is useful not only as real text of the email genre, but it can be easily divided into smaller subsets suitable for student projects. (And, of course, there is also the human interest factor in that the data set is available due to its use in the Enron court proceedings!)

The goal of the project is to use increasing levels of NLP to characterize a selected subset of Enron email texts. The project is designed to be carried out in two parts, involving two assigned levels of NLP. The first level, part-of-speech tagging, is accomplished as Task 1 and the second, phrase-bracketing or chunk-parsing, is assigned as Task 2. However, the overall characterization of the text is left open-ended, and the student teams chose various dimensions for their analyses. Projects included analyzing the topics of the emails of different people, social network analyses based on people and topics mentioned in the email text, and analyses based on author and recipient header information about each email.

Teams are established for these projects by the professor based on the capabilities and interests of the individual students as reported in short self-surveys. This resulted in teams on which there is a mix of computer science, linguistics and information science expertise. The teams accomplished the tasks of choosing a data analysis method, processing data subsets, designing NL processing to accomplish the analysis, programming the NL processing, conducting the data analysis, and preparing the in-class reports.

5.3.1 Tools Used in the Project

For preliminary processing of the Enron email files, programs and data made available by Professor Andrés Corrada-Emmanuel at the University of Massachusetts at Amherst, and available at http://ciir.cs.umass.edu/~corrada/ were used. The emails were assigned MD5-digest numbers in order to identify them uniquely, and the data consisted of mappings from the digest numbers to files, as well as to authors and recipients of the email. The programs contained filters that could be used to remove extraneous text such as headers and forwarded text. The teams adapted parts of these programs to convert the email files to files with text suitable for NL processing.

For the NL processing, the Natural Language Toolkit (NL Toolkit or NLTK), developed at the University of Pennsylvania by Loper and Bird (2002), and available for download from Source-Forge at http://nltk.sourceforge.net/ was used. The NL Toolkit is a set of libraries written in the Python programming language that provides core data types for processing natural language text, support for statistical processing, and a number of standard processing algorithms used in NLP, including tokenization, part of speech (POS) tagging, chunk parsing, and syntactic parsing. The toolkit provides demonstration packages, tutorials, example corpora and documentation to support its use in educational classes. Experience using the Toolkit shows that in order to use the NL Toolkit, one member of each team should have at least some programming background in order to write Python programs that use the NL Toolkit libraries. The use of Python as the programming language was successful in that the level needed to use the NL Toolkit was manageable by the students with only a little programming background and in that the computer science students were able to adapt to the Python programming style and could easily utilize the classes and libraries.

At the beginning of the term project, the students were offered a lab session and lab materials to get them started. Since no one knew the Python programming language at the outset, there was an initial learning curve for the Python language as well as for the NL Toolkit. The lab materials provided to the students consisted of installation instructions for Python and NL Toolkit and a number of example programs that combined programming

snippets from the NL Toolkit tutorials to process text through the NLP phases of tokenization, POS tagging and the construction of frequency distributions over the POS tagged text. During the lab session, some of the example programs were worked through as a group with the goal of enabling the students to become competent in Python and to introduce them to the NL Toolkit tutorials that had additional materials. The NL Toolkit tutorials are extensive on the lower levels of NL processing (e.g. lexical and syntactic) and students with some programming background were able to utilize them.

As part of their first task, the student teams were asked to select a subset of the Enron emails to work with. The entire Enron email directories were placed on a server for the teams to look at in making their selections. The teams also used information about the Enron employees as described in a paper by Corrada-Emmanuel (2005). Some student teams elected to work with different email topic folders for one person, while others chose a few email folders each from a small number of people (2-5). Their selected emails first needed to be processed to text using programs adapted from Corrada-Emmanuel. For the most part, the sub-corpora choices of the student teams worked out well in terms of size and content. Several hundred emails turned out to be a good size, providing enough data to experience the challenges of long processing times and to appreciate why NLP is useful in processing large amounts of data, without being unduly overwhelmed. Initially, one team chose all the emails from several people. The number of email files involved was several thousand and it took several hours to unzip those directories, let alone process them, and they subsequently reduced the number of files for their analysis.

The first task was to analyze the chosen emails based solely on lexical level information, namely words with POS tags. NL Toolkit provides libraries for tokenization where the user can define the tokens through regular expressions, and the students used these to tailor the tokenization of their emails. The Toolkit also provides a regular expression POS tagger as well as n-gram taggers, and the students used these in combination for their POS tagging. Students experimented with the Brown corpus and a part of the Penn Treebank corpus,

provided by NL Toolkit to train the POS taggers, and compared the results.

Building on the first task, the second task extended the analysis of the chosen emails to phrases from the text. Again, NL Toolkit provides a library for chunk parsing where regular expressions can be used to specify patterns of words with POS tags either to be included or excluded from phrases. Since chunk parsing depends on POS tagging, there was a need for a larger training corpus. A research center within the Information School has a license for Penn Treebank, and provided additional Penn Treebank files for the class to use for that purpose. Most teams used regular expressions to bracket proper names, minimal noun phrases, and verb phrases. One team used these to group maximal noun phrases, and another team used regular expressions to find patterns of communication verbs for use in social network analysis.

In retrospect, it was found that the chunk parsing did not take the teams far enough in NLP analysis of text. Experience in teaching using the NL Toolkit suggests that use of the syntactic parsing libraries to find more complex structures in the text would have provided more depth of analysis. Students also suggested that they would have liked to incorporate semantic level capabilities, such as the use of WordNet to find conceptual groupings via synonym recognition. The next offering of the course will include these improvements.

Using the NL Toolkit for NL processing worked out well overall and enabled the students to observe and appreciate details of the processing steps without having to write a program for every algorithm themselves. The tutorials are good, both at explaining concepts and providing programming examples. There were a few places where some data structure details did not seem to be sufficiently documented, either in the tutorials or in the API. This was true for the recently added Brill POS tagger, and is likely due to its recency of addition to the toolkit. However for the most part, the coverage of the documentation is impressive.

6 Evaluation

Multiple types of evaluation are associated with the course. First, the typical evaluation of the students by the professor (here, 2 professors) was done on multiple dimensions that contributed proportionately to the student's final grade as follows:

- In-Class group exercises 20%
- NLToolkit Team Assignments 35%
- NLP Application Team Poster & 35%
  Presentations
- Contributions to class discussion 10%
  (both quality and quantity)

Additionally, each team member evaluated each of their fellow team members as well as themselves. This was done for both of the teams in which a student participated. For each team member, the questions covered: the role or tasks of the student on the project; an overall performance rating from 1 for POOR to 4 for EXCELLENT; the rationale for this score, and finally; what the student could have done to improve their contribution. Knowledge of this end-of-semester team self-evaluation tended to ensure that students were active team contributors.

The professor was also evaluated by the students. And while there are quantitative scores that are used by the university for comparison across faculty and to track individual faculty improvements over time, the most useful feature of the student evaluations is the set of open-ended questions concerning what worked well in the course, what didn't work well, and what could be done to improve the course. Over the years of teaching this course, these comments (plus the mid-term evaluations) have been most instructive in efforts to find ways to improve the course. Frequently the suggestions are very practical and easy to implement, such as showing a chart with the distribution of grades on each assignment when they are returned so that the students know where they stand relative to the class as grading is on a scale of 1 to 10.

7. Indicators of Success

Finally, how is the success of this course measured in the longer term? For this, success is measured by: whether students elect to do continued work in NLP, either in the context of further courses in which NLP is utilized, such as Information Retrieval or Text Mining; whether the masters (and undergraduate) students decide to pursue an advanced degree based on the excitement engendered and knowledge gained from the NLP course; or whether PhD students elect to do continued re-

search either in the school's Center for Natural Language Processing or as part of their dissertation. For students in a terminal degree program, success is reflected by their seeking and obtaining jobs that utilize the NLP they have learned in the course and that has provided them with a solid, broad basis on which to build. For several of the undergraduate computer science students in the course, their NLP experience has given them an added dimension of specialization and competitive advantage in a tight hiring market.

An additional measure of success was the request by the doctoral students in the home school for a PhD level seminar course to build on the NLP course. This course is entitled Content Analysis Research Using Natural Language Processing and will enable PhD students doing social science research on large textual data sets to explore and apply the NLP tools that are developed within the school, as well as to understand how these NLP tools can be successfully interleaved with commercial content analysis tools to support rich exploration of their data. As is the current course, this seminar will be open to PhD students from all schools across campus and already has enrollees from public policy, communications, and management, as well as information science.

8. Summary

While it might appear that a disproportionate amount of thought and attention is given to the more human and social aspects of designing and conducting this course, experience shows that such attention is the key to the success of this diverse body of students in learning and understanding the content of the course. Furthermore, given the great diversity in class-level and disciplinary background of students, this attention to structuring the course has paid off in the multiple ways exemplified above. While it is obvious that a course for computer-science majors alone would be designed quite differently, it would not provide the enriched understanding of the field of NLP and its application value that is possible with the contributions by the variety of disciplines brought together in this course.

References

Loper, E. & Bird, S., 2002. NLTK, the Natural Language Toolkit. In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics.

Corrada-Emmanuel, A. McCallum, A., Smyth, P., Steyvers, M. & Chemudugunta, C., 2005. Social Network Analysis and Topic Discovery for the Enron Email Dataset. In Proceedings of the Workshop on Link Analysis, Counterterrorism and Security at 2005 SIAM International Conference in Data Mining.