

# Building a Flexible, Collaborative, Intensive Master’s Program in Computational Linguistics

**Emily M. Bender**

University of Washington  
Department of Linguistics  
Box 354340  
Seattle WA 98195-4340  
ebender@u.washington.edu

**Fei Xia**

University of Washington  
Department of Linguistics  
Box 354340  
Seattle WA 98195-4340  
fxia@u.washington.edu

**Erik Bansleben**

University of Washington  
Educational Outreach  
UW Tower, Box 359485  
4333 Brooklyn Ave., NE  
Seattle, WA 98105-9485  
ebansleben@extn.washington.edu

## Abstract

We present the design of a professional master’s program in Computational Linguistics. This program can be completed in one-year of full-time study, or two-three years of part-time study. Originally designed for CS professionals looking for additional training, the program has evolved in flexibility to accommodate students from more diverse backgrounds and with more diverse goals.

## 1 Introduction

In the past two decades, there has been tremendous progress in natural language processing and various undergraduate/graduate programs in language technology have been established around the world (Koit et al., 2002; Frederking et al., 2002; Dale et al., 2002; Uszkoreit et al., 2005; Pilon et al., 2005).

This paper introduces the University of Washington’s Professional Masters Program in Computational Linguistics (CLMA)—one of the largest programs of its kind in the United States—and highlights unique features that are key to its success. The CLMA program is currently operating in its third year as a fee-based degree program managed jointly by the Department of Linguistics and the Educational Outreach arm of the University. The program is distinguished by its programmatic focus, its flexibility, its format and delivery as well as in the partnerships that are an integral part of this degree.

This paper highlights how these features of our program contribute to effective teaching in our interdisciplinary field as well as making the program

relevant to both working professionals and students on the research track. We provide a brief program overview highlighting the people and partnerships involved, course design, practicum and research options, and dealing with student diversity. We then reflect on how we have approached the challenges of setting up the program and our future plans.

## 2 Program Overview

With working professionals who wanted to return to school to retool for a career change in mind, we designed a curriculum that can be completed in 12 months of intensive full-time study. In this way, students can complete the degree without leaving the working world for too long. In addition, the compactness of the program means that even with part-time study (one-two courses per quarter), the program can be completed within a reasonable time horizon (two-three years). Once the program got underway, we found that we also had strong interest from Linguistics students. The flexibility of the part-time option has allowed us to develop a two-year schedule which accommodates students who need time to get up to speed with key CS concepts.

The curriculum is designed around hands-on and collaborative work which prepares students for industry jobs. At the same time, the courses are structured around fundamental building blocks rather than applications in order to teach students to think like computational linguists, and to provide them with an educational foundation which will remain relevant for years to come.

This section gives an overview of the CLMA program, including its structure and participants.

## 2.1 Program Structure

The CLMA program comprises three quarters (nine courses) of coursework and a summer master's project, which can take the form of an internship or a master's thesis (§3.5). The courses, described in more detail in §3 below, include three courses in Linguistics and six in Computational Linguistics.

The program also offers a series of talks by computational linguists working at local companies, an informal seminar of the computational Linguistics lab group (which includes PhD students and focuses on research methodology), and career development services as we work with students to place them in internships and post-degree jobs.

## 2.2 The new certificate program

This summer, the program is branching out with a new Certificate in Natural Language Technology. This three-course Certificate includes two NLP courses from the Masters degree and an introductory course titled "Computational Linguistics Foundations" which serves both this Certificate audience and acts as a refresher course for some degree students. It reinforces the concepts from Linguistics, CS and statistics that students rely on heavily in the core course sequence (see §4). The Certificate is an alternate course of study for those students wanting to study a single topic in depth but who are not yet ready to commit to the entire degree.

## 2.3 Synchronous online and in-person courses

As part of the certificate summer launch, we will be offering a selection of courses in a synchronous online and in-person format, streaming the content from the classroom to a live remote audience. This will allow us to extend the reach of the program and make study in Computational Linguistics available to students who otherwise would not have access to instruction in this field without relocating.

In the context of current globalization trends, the need for online and distance education is growing (Zondiros, 2008), and indeed we hope that our audience will extend beyond North America. At the same time, we agree with Neal and Miller's (2004) position that even with remote participants, the classroom remains a key part of the educational experience. We have thus adopted an approach that

allows students to be part of a virtual classroom where they can engage with other students while still working from a remote location. This surmounts the hurdle of more traditional distance or online education that is primarily text-based and asynchronous.

In a pilot of an online course offering in Autumn 2007 (with Intro to Syntax), we found that most of the pieces were already in place for taking our courses online: Course materials are already disseminated through websites, student programming work is done on a server cluster that is always accessed remotely, and most of the discussion outside of class happens on electronic discussion boards.

## 2.4 Faculty and Staff

The CLMA program is taught by a group of instructors who combine academic knowledge and practical expertise in the field. The program budget supports two faculty positions, one tenure-track (and guaranteed by the College of Arts and Sciences), and one two-year visiting position.<sup>1</sup> Each of these faculty teach two of the core NLP courses described in §3.1 below and one seminar each year. In addition, they share the work of supervising MA theses and internships over the summer. In recognition of this supervising load, their schedules are arranged so that they each have one non-summer quarter off from teaching. A third faculty member in Computational Linguistics teaches three graduate-level courses in Computational Linguistics per year, and takes on one-two MA students from each CLMA cohort.

The program also includes a part-time administrator and a technical specialist within the Department of Linguistics. In addition, the program includes affiliated instructors and guest lecturers, ranging from faculty members of other departments such as CS and Statistics to researchers from industry.

## 2.5 Students

A strength of the program is its emphasis on student diversity and allowance for individualized student needs. The program allows for both part-time and full-time enrollment and includes both recent college graduates as well older, more non-traditional students who work in industry. We have students from throughout the US, as well as from

<sup>1</sup>To be converted to tenure-track in the future, once the program has a longer track-record.

Canada, China, Germany, India, Japan, and Russia. Two-year course schedules allow students to begin CLMA course work while simultaneously taking CS and statistics courses in the first year, increasing the diversity of backgrounds of our students. The program is starting to deliver several courses online (see §2.3) which provides additional flexibility to local students while also reaching a wider national and international audience. Lastly, the program seeks to foster both research and industry interests by providing both thesis and internship options.

## 2.6 Advisory board

The program maintains an advisory board composed of significant industry researchers and practitioners, including representatives from AT&T, Boeing, Google, IBM, Microsoft, Nuance, PARC, and Yahoo!, and faculty from several departments at the University. This board was instrumental in developing the original program focus and curriculum outline, as well as providing input from the perspective of employers as the University decided whether or not to launch the program. It continues to be engaged in guiding the program's content, providing internship opportunities for students, and keeping the content relevant to current industry trends.

## 2.7 Support from Educational Outreach

Another element of success is a centralized infrastructure of administration and support available through the University's Educational Outreach division (UWEO) which manages the CLMA program, among more than 30 degree offerings.

UWEO provides many benefits, including considerable institutional knowledge in starting a new degree program, providing methods of separating fee-based revenue from that of state-run programs, marketing expertise, fiscal management, registration services and more. As the outreach arm of the University, UWEO works closely with non-traditional students and is able to leverage its industry contacts to assist serving this community most effectively.

Lastly, partnering with UWEO also serves as a method of risk management for all new degree programs. As a state school, the University may have difficulty in getting state approval and funding for new degree programs unless initial need and demand can be demonstrated persuasively. UWEO can as-

sume a certain degree of risk during the start-up phase of new programs allowing for additional flexibility and time to reach financial viability.

## 3 Curriculum Design

The program curriculum was designed according to the following principles: (1) we should provide students with an educational foundation that is relevant in the long term; (2) we should emphasize hands-on coursework to provide depth of understanding as well as practical experience students can point to when looking for a job; and (3) we should highlight unifying themes in a diverse set of subject matter.

The courses were designed by taking an inventory of the applications and research areas that comprise Computational Linguistics, and then breaking them down into subtasks. These subtasks were then grouped by similarity into coherent courses, and the courses into core and elective sets. Three topics resisted classification into any particular course: ambiguity resolution, evaluation, and multilingualism. These became our cross-cutting themes which are highlighted throughout all of the courses. In addition to understanding each subtask, working computational linguists need to know how to combine the stages of linguistic processing into end-to-end systems. For this reason, we include a capstone "Systems and Applications" course in which students work in groups to create an NLP application.

Key to making the Computational Linguistics curriculum fit into one calendar year was deciding not to include the course "Intro to Computational Linguistics." Such a course serves to expose students to a broad range of topics in the field and get them interested in exploring further. CLMA students are already interested in further studies in Computational Linguistics, and will be exposed to a broad range of topics throughout the curriculum. However, we did still want to give the students an overview of the field so that they could see how the rest of their studies will fit in. This is done through a two-day orientation at the start of each year. The orientation also introduces the three cross-cutting themes mentioned above, gives the students a chance to get to know each other and the CLMA faculty, and provides practical information about the university such as libraries, computing lab facilities, etc.

### 3.1 Required courses

There are six required courses: The first two are Linguistics courses, and the remaining four form the NLP core courses. Among the four NLP courses, Ling 572 should be taken after Ling 570, and Ling 573 should be taken after Ling 570, 571, and 572.

**Ling 450 (Intro to Linguistic Phonetics):** Introduction to the articulatory and acoustic correlates of phonological features. Issues covered include the mapping of dynamic events to static representations, phonetic evidence for phonological description, universal constraints on phonological structure, and implications of psychological speech-sound categorization for phonological theory.

**Ling 566 (Intro to Syntax for Computational Linguistics):** Introduction to syntactic analysis and concepts (e.g., constituent structure, the syntax-semantics interface, and long-distance dependencies). Emphasis is placed on formally precise encoding of linguistic hypotheses and designing grammars so that they can be scaled up for practical applications. Through the course we progressively build up a consistent grammar for a fragment of English. Problem sets introduce data and phenomena from other languages.

**Ling 570 (Shallow Processing Techniques for NLP):** Techniques and algorithms for associating relatively surface-level structures and information with natural language data, including tasks such as tokenization, POS tagging, morphological analysis, language modeling, named entity recognition, shallow parsing, and word sense disambiguation.

**Ling 571 (Deep Processing Techniques for NLP):** Techniques and algorithms for associating deep or elaborated linguistic structures with natural language data (e.g., parsing, semantics, and discourse) and for associating natural language strings with input semantic representations (generation).

**Ling 572 (Advanced Statistical Methods in NLP):** Major machine learning algorithms for NLP, including Decision Tree, Naive Bayes, kNN, Maximum Entropy, Support Vector Machine, Transformation-Based Learning, and the like. Students implement many of these algorithms and use them to solve classification and sequence labeling problems.

**Ling 573 (NLP Systems and Applications):** Design and implementation of coherent systems for practical applications, with topics varying year to year. Sample topics: machine translation, question answering, information retrieval, information extraction, dialogue systems, and spell/grammar checking. In 2006, the students collectively built a question answering system, which was further developed into a submission to the TREC competition (Jinguji et al., 2006). This year's class is developing a chatbot to submit to the Loebner Prize competition, an implementation of the Turing Test.

Among the required courses, Ling 566 was created a year before the CLMA program started, and has been taught four times. Ling 450 is an established course from our Linguistics curriculum. Ling 570-573 were newly created for this program, and have each been taught three times now. We have put much effort in improving course design, as discussed in (Xia, 2008).

### 3.2 The prerequisites for the required courses

In order to cover the range of methodologies and tasks that our program does in its core sequence, we need to set as a prerequisite the ability to program, including knowledge of data structures and algorithms, and expertise in C++ or Java.<sup>2</sup> Another prerequisite is a college-level course in probability and statistics. Without such knowledge, it is all but impossible to discuss the sophisticated statistical models covered in the core NLP courses. For the two Linguistics required courses, the only prerequisite is a college-level introductory course in Linguistics or equivalent. Because our students have very diverse backgrounds, we have tried several approaches to help the students meet all these prerequisites, which will be discussed in §4.

### 3.3 Elective courses

All students must take three electives, including one in Linguistics, one in Computational Linguistics, and one more in Computational Linguistics or a related field. The Linguistics electives are drawn from the broad range of graduate-level Linguistics courses offered in the department. The related fields

<sup>2</sup>Knowing Perl or Python is recommended but not required, as we believe that good C++ or Java programmers can learn Perl or Python quickly.

electives include courses in CS and Electrical Engineering on topics such as Machine Learning, Graphical Models, Artificial Intelligence, and Human-Computer Interaction as well as courses in the Information School on topics such as Information Retrieval. We maintain a list of pre-approved courses, which grows as students find additional courses of interest and petition to have them approved.

The annual elective offerings in Computational Linguistics include Multilingual Grammar Engineering, as well as seminars taught by the Computational Linguistics faculty and by guest experts (including researchers in local industry), covering new topics each year. Recent topics include: Corpus Management, Development and Use, Text-to-Speech, Multimodal Interfaces, Lexical Acquisition for Precision Grammars, Semi-supervised and Un-supervised Learning for NLP, and Information Extraction from Heterogeneous Resources. There are four-five such seminars per year, three from the Computational Linguistics faculty and one-two from guest experts.

We ask students to present their choices of electives for approval, and require that they articulate reasons why their range of choices constitutes a coherent course of study.

### 3.4 Hands on, interactive courses

All of the courses in the curriculum are hands-on, emphasizing learning through doing as well as collaboration between the students. Theoretical concepts introduced in lecture are put into practice with problem sets (e.g., in Intro to Syntax), programming assignments (in the core sequence) and opened-ended projects (in the Systems and Applications course and the seminars). Student collaboration is promoted through group projects as well as active online discussion boards where students and faculty together solve problems as they arise.

### 3.5 The master's project

In addition to nine courses, the students need to complete a master's project, either through an internship or through completing a master's thesis.

**The internship option:** Internships counting towards the MA degree must be relevant to Computational Linguistics or human language technology

more broadly. Students develop a pre-internship proposal, including a statement of the area of interest and proposed contributions, a discussion of why the company targeted is a relevant place to do this work, and a list of relevant references. Once the students have been offered and accepted an internship, they write a literature review on existing approaches to the task in question.

At the end of the internship, students write a self-evaluation which they present to the internship supervisor for approval and then to the faculty advisor. In addition, we require a confidential, written evaluation from the intern's supervisor which references the self-evaluation. If this evaluation does not indicate satisfactory work, the internship will not count.

Students also write a post-internship report, including a description of the activities undertaken during the internship and their results, a discussion of how the program course work related to and/or prepared the student for the internship work, and a second version of the literature review. We expect the second review to be different from the initial version in incorporating the additional perspective gained in the course of the internship as well as any additional key papers that the student discovered in the course of internship work.

**The thesis option:** This option is recommended for students who wish to petition for admission to the Department's PhD program, and encouraged for students who wish to apply to other PhD programs in the near future. An MA thesis typically involves the implementation of working systems (or extensions or experimental evaluations thereof). In some cases, they may provide theoretical contributions instead. MA theses require a thorough literature review, are typically longer (30-50 pages), and represent the kind of research which could be presented at major conferences in our field.

**The milestones:** While the internship and a significant portion of the thesis work are conducted in the summer for full-time students, we start monthly graduation planning meetings as early as the preceding October to help students decide which option they should take. For those seeking internships, we will help them identify the companies that match their interests and make the contact if possible.

Students who choose the thesis option are required to turn in an initial thesis proposal that includes a thesis topic, a summary of major existing approaches, the students' proposed approach, and a plan for evaluation. With the feedback from the faculty, the students will revise their proposals several times before finalizing the thesis topic. Students are encouraged to take elective courses relevant to their topic. Because the amount of research is required for a good master's thesis, we expect students with this option to take one or two additional quarters to finish than the ones who choose the internship option.

## 4 Challenges

In this section, we address several challenges that we encountered while establishing the new program.

Students enrolling in our program have varied backgrounds in Linguistics, CS and other undergraduate majors. In addition, some students come to us straight from undergraduate studies, while others are re-entry students. To better prepare students for the program, starting this year we offer an eight-week summer refresher course, which reinforces the most important skills from contributing disciplines to prepare students for the CLMA core courses. The course covers the following topics: (1) formal grammars and formal languages, (2) finite-state automata and transducers, (3) review of main concepts from probability and statistics, (4) review of major data structures and algorithms, and (5) using Unix and computer clusters.

After students are admitted to the program, they are asked to take an online placement test to identify the areas that they need to strengthen before entering the program. They can then choose to take the summer course or study on their own.

While some of our students, typically fresh out of college or stepping out of the workforce for re-training to switch careers, are eager to complete the degree in one year, others wish to complete the program while continuing their current employment or simply need more time. We offer various options to accommodate different needs:

**Part-time vs. full-time** Students can complete the program in one year, taking three classes each quarter and completing the master's project in the summer. At this pace, the program is very intense.

The program also offers part-time options, allowing students to take courses one or two at a time. This works well for students who are currently employed and also leaves time for students coming from a Linguistics background to take CS and Statistics courses before approaching the Computational Linguistics core sequence. While full-time students must start in Autumn quarter, part-time students can start in any academic quarter.

**Curriculum flexibility** Students who come to us with an extensive background in Linguistics (e.g., a prior MA), can waive one or more of the Linguistics requirements, giving them more time in their schedule for Computational Linguistics or related fields courses, such as CS.

**Program options** Our courses are open to qualified students for single-course enrollment, allowing people who don't have the time or financial resources to commit to the whole master's program to benefit from the course offerings. In addition, the three-course certificate provides more continuity than single-course enrollment (though less than the full master's program) as well as the recognition of a certificate. In either case, graduate non-matriculated status allows such students to apply their coursework to the master's program at a later date.

**Master's project options** In providing for both internships and master's theses, the program can accommodate students seeking training for positions in industry as well as those seeking to continue graduate studies. In the former case, the practical experience of an internship together with the industry connections it can provide are most valuable. In the latter case, a chance to do independent research is more appropriate. Students who spread the program out over more than one year can do internships in the summer between years one and two in addition to the master's project (internship or thesis) in the second summer. Finally, the "internship option" can also be fulfilled by ordinary full-time employment: when students begin full-time positions in the summer after they complete coursework or apply the knowledge gained in the master's program to new projects at their current places of employment.

**In class or online** By offering our courses in a hybrid, synchronous in-person and online format, we

add the flexibility to attend our program from anywhere in the world while still benefiting from the same courses, lectures, online discussions and collaborative work. The online option is also beneficial to local students, allowing them to tune in, for example, when home caring for a sick child, to review lectures previously attended, to attend online on days without other on-campus obligations and to avoid the commute. In the 2008-2009 school year, three of our courses will be offered in this format, and we plan to extend the offerings going forward.

## 5 Outcomes

### 5.1 Enrollment, graduation rate and placement

In years 1-3, 70 students have enrolled, and about 30 of them enrolled as full-time students.<sup>3</sup> To date 13 have completed the program, and at least nine of them are currently in jobs related to Computational Linguistics. Another 12 are projected to graduate this year. Out of these 25 students, 15 chose the internship option and 10 chose the thesis option. We have placed students in internships with companies such as Amazon.com, Google, Microsoft, PARC, Tegic (Nuance), and VoiceBox, and have graduates working at companies such as Cataphora, Cisco, Google, InXight, Tegic (Nuance), and Voice-Box. Among the 10 students who took the thesis option, four received RAs from CLMA faculty's research grants, and at least two will enroll in our Ph.D. program after receiving their MAs.

Recent internal research completed by UWE0 identified a total of 34 CL programs, 23 in the US and 11 in Western Europe. These programs vary from named degrees in Computational Linguistics or a similar variant, to concentrations in other degrees and to loose courses of study. It appears that there is one other university in the US that has enrollment as high or higher than our own, but all other programs typically have at least 50% fewer students enrolling as of 2007. Given that this program is only in its third year, we consider this level of high comparative enrollment a strong measure of success. Additionally, during this 3 year period, there has been an upward trend in applications which may be a reflection

<sup>3</sup>Some of them later switched to part-time status due to various reasons (e.g., the intensity of the program, financial consideration).

on the growth and awareness of the discipline, but may also be a reflection on the growing reputation of the program.

### 5.2 Student feedback

We sent an anonymous survey to all alumni and current students (N=70) asking them about the effectiveness of the overall CLMA program, individual courses, the curriculum, success in getting a job as well as for some qualitative feedback about the strengths and weaknesses of the program. We received 31 responses (44% response rate). For the sake of brevity, we will provide a selection of questions and categorize the results as follows: positive (1=very well, 2=fairly well), neutral (3=so so); negative (4=not very well, 5=poorly).

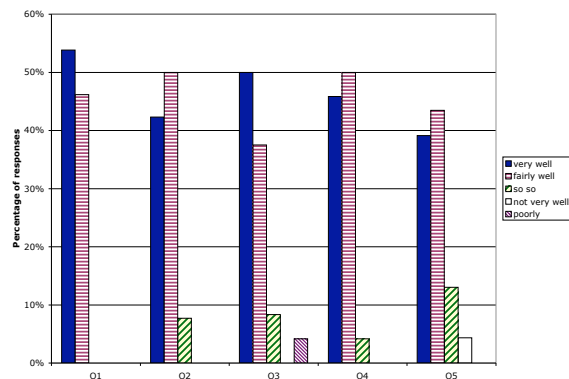


Figure 1: Student and alumni responses

As shown in Figure 1, the responses were overwhelmingly positive. The first four questions ask how well the program as a whole helped the students achieve the goals of learning to think like a computational linguist (Q1), understanding the state of the art in Computational Linguistics (Q2), understanding the potential contributions of both machine learning and knowledge engineering (Q3), and preparation for a job in industry (Q4). The fifth question asks how helpful the CLMA experience has been/will be in finding a job.<sup>4</sup> There were a number of other questions, but the results are all very similar to the ones above. These same questions were also asked with respect to individual courses. The results were again similar, although slightly lower. Positive responses were in the range of 80%-95%, neutral in the range

<sup>4</sup>Each of these questions was answered by 24-27 students.

of 5-20% and negative responses were usually limited to no more than 5%. For the question of how well the program has prepared students for their current job (alumni only, N=5), 100% answered positively. For the question about how important the program was in attaining their current job, again 100% felt that the program was crucial.

We also received valuable qualitative feedback on this survey. The majority of students (67%) felt that the program was very intense, but very much worthwhile. The faculty consistently receives high praise; students enjoy the small hard-working community; and comments indicate that the coursework is relevant for their future career. When asked about suggestions for improvement, students provided a number of logistical suggestions, would like to see some degree of student mentoring, and seek to find ways to reduce the intensity of the program, especially for part-time students who are working. It is clear, though, from the overall survey results, that students feel very positive about the program as a whole, and its relevance for their professional future.

While we at first thought the program to be primarily a one-year program, the intensity of the curriculum has resulted in a number of students taking longer than one year to complete the program which has impacted the number of students who have thus far completed. Consequently, we will consider student feedback from the survey which—in conjunction with the new preparatory course—should lead us to find methods of reducing the intensity but maintaining the high quality.

## **6 Conclusion and future directions**

### **6.1 Lessons learned**

In starting this program, we had the privilege of designing the curriculum first and then hiring faculty to teach the required courses. We worked closely with our advisory board to develop a course of study well-suited to training students for industry jobs, while also striving to design a program that will remain relevant to graduates many years down the line.

Financial support from UWEO allowed us to jump in with both feet, offering the full curriculum from year one. This was critical in attracting a strong and reasonably large student body. It also provided the freedom to design a curriculum that goes in-

depth into Computational Linguistics.

Other facets of our curriculum which contribute to its success include: (1) We combine in-depth exploration of particular topics with cross-cutting themes that tie the courses together. (2) The courses emphasize hands-on work, providing immediate motivation to delve deeper into the theoretical concepts presented. (3) The program combines high intensity with high levels of support: We ask the students to attempt real-world scale projects and then assist them in achieving these goals through providing software to work from, offering high levels of online interaction to answer questions, and facilitating collaboration. By working together, the students can build more interesting systems than anyone could alone, and therefore explore a broader territory. In addition, collaborative projects help students benefit from each other's diverse backgrounds.

At the same time, we've found providing multiple ways of completing program requirements to be key to allowing students from different backgrounds to succeed. Exceptional students coming from Linguistics can get up to speed quickly enough to complete the program on a full-time schedule (and some have), but many others benefit from being able to take it more slowly, as do some students from a CS background. We also find that having expertise in Linguistics among the students significantly benefits the overall cohort.

### **6.2 Future directions**

In the near future, we plan to expand our online offerings, which directly expands our audience and benefits local students as described above. We have found connecting course work to faculty research and/or external competitions such as TREC and the Loebner Prize to be extremely motivating and rewarding for students, and plan to seek more opportunities for doing so. We are also expanding out interdisciplinary reach within the university. The TREC submission was done jointly with faculty from the Information School. This year's guest faculty course will be offered jointly with a course in the School of Art on interface design. In pursuing all of these directions, we will benefit from input from our advisory board as well as feedback from current students and alumni.



## References

- Robert Dale, Diego Molla Aliod, and Rolf Schwit-  
ter. 2002. Evangelising Language Technology: A  
Practically-Focussed Undergraduate Program. In *Pro-  
ceedings of the First ACL Workshop on Effective Tools  
and Methodologies for Teaching NLP and CL*.
- Robert Frederking, Eric H. Nyberg, Teruko Mitamura,  
and Jaime G. Carbonell. 2002. Design and Evolution  
of a Language Technologies Curriculum. In *Proceed-  
ings of the First ACL Workshop on Effective Tools and  
Methodologies for Teaching NLP and CL*.
- Dan Jinguji, William D. Lewis, Efthimis N. Efthimiadis,  
Joshua Minor, Albert Bertram, Shauna Eggers, Joshua  
Johanson, Brian Nisonger, Ping Yu, and Zhengbo  
Zhou. 2006. The University of Washington's UW-  
CLMAQA system. In *Proceedings of the Text Re-  
trieval Conference (TREC) 2006*, Gaithersburg, Mary-  
land.
- Mare Koit, Tiit Roosmaa, and Haldur Oim. 2002. Teach-  
ing Computational Linguistics at the University of  
Tartu: Experience, Perspectives and Challenges. In  
*Proceedings of the First ACL Workshop on Effective  
Tools and Methodologies for Teaching NLP and CL*.
- Diane Neal, Lisa nad Miller. 2004. Distance educa-  
tion. In Robert W. Proctor and Kim-Phuong L. Vu,  
editors, *Handbook of Human Factors in Web Design*.  
Lawrence Erlbaum Associates.
- Suléne Pilon, Gerhard B Van Huyssteen, and Bertus  
Van Rooy. 2005. Teaching language technology at the  
North-West University. In *Proceedings of the Second  
ACL Workshop on Effective Tools and Methodologies  
for Teaching NLP and CL*.
- Hans Uszkoreit, Valia Kordoni, Vladislav Kubon,  
Michael Rosner, and Sabine Kirchmeier-Andersen.  
2005. Language technology from a European perspec-  
tive. In *Proceedings of the Second ACL Workshop on  
Effective Tools and Methodologies for Teaching NLP  
and CL*.
- Fei Xia. 2008. The evolution of a statistical nlp course.  
In *Proceedings of the Third ACL Workshop on Effec-  
tive Tools and Methodologies for Teaching NLP and  
CL*, Columbus, Ohio, June.
- Dimitris Zondiros. 2008. Online, distance education  
and globalization: Its impact on educational access,  
inequality and exclusion. *European Journal of Open,  
Distance and E-Learning*, Volume I.