Teaching Machine Translation in a Graduate Language Technologies Program

Teruko Mitamura, Eric Nyberg, Robert Frederking
Language Technologies Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
{teruko, ehn, ref}@cs.cmu.edu

Abstract

This paper describes a graduate-level machine translation (MT) course taught at the Language Technologies Institute at Carnegie Mellon University. Most of the students in the course have a background in computer science. We discuss what we teach (the course syllabus), and how we teach it (lectures, homeworks, and projects). The course has evolved steadily over the past several years to incorporate refinements in the set of course topics, how they are taught, and how students "learn by doing". The course syllabus has also evolved in response to changes in the field of MT and the role that MT plays in various social contexts.

1 Introduction

The Language Technologies Institute (LTI) is a unit within the School of Computer Science at Carnegie Mellon University. The LTI has offered MS and Ph.D. degrees in language technology since 1996[1]. The core LTI curriculum consists of four focus areas (Linguistics, Computer Science, Statistical/Learning and Task Orientation); Ph.D. students are required to take at least one course from each of the four focus areas. The Task Orientation focus, which also includes Information Retrieval, Speech Recognition, and Software Engineering, has always included a Machine Translation course. The MT class typically has an enrollment of about 15 students. The curriculum also contains an MT Lab course, where students complete hands-on exercises related to the MT course lecture material (see Section 5).

The role of MT in teaching varies, depending on the nature of the students and the goals of the instruction (Somers, 2001). The students may include translators, second language learners, or university students with different academic orientations (computer science, computational linguistics, etc.). The goals of instruction might focus on the use of existing MT tools in particular applications (e.g., teaching job skills to translators), the use of MT in second-language learning, or how to create new MT systems that surpass existing approaches.

The students who take our course are generally M.S. or Ph.D. students in the Language Technologies Institute with a background in computer science or linguistics. All students are expected to write programs while taking courses at LTI, and students who lack a programming background are expected to take appropriate preparatory courses. The goals of the MT course are primarily technical, and focus on teaching students how to develop new MT systems. We also attempt to infuse the students with an appreciation of the business issues surrounding successful deployment of MT systems. We do not, however, spend much time teaching specific MT tools or commercial software to the students (although students may elect to evaluate a commercial tool as part of a term project).

In the remainder of the paper, we describe the evolution and current status of the course, and present some of the challenges we have encountered while teaching the course.

[1] The LTI was created as an expansion of the earlier Center for Machine Translation (CMT), which existed at CMU from 1986 to 1996.

2 Objectives

There are two sets of objectives for the course: specific objectives related to the MT subject area, and general objectives associated with the graduate programs. The main objectives within the MT subject area include:

- Obtain a basic understanding of MT systems and MT-related issues;
- Learn about the theory of MT and approaches to MT;
- Learn about basic techniques for MT development, in preparation for the MT Lab course and real-world MT system project development;
- Obtain in-depth knowledge of one current topic in MT, or perform an analysis of a given MT problem, matching it with the most suitable techniques.

The general objectives include learning how to find an interesting research topic, learning to conduct a research investigation, and learning to organize and present research results. Students are required to give final presentations to the entire class, and in-class discussion of individual research projects is an important educational experience in the course. Through their term project work, students begin to learn how to become MT independent researchers.

3 What We Teach

There are no formal prerequisites for the MT course, but students are encouraged to take two other LTI courses (“Grammars and Lexicons”, “Algorithms for NLP”) before they enroll in the MT course. The course reading materials consist of a compilation of articles, chapters and papers from various textbooks, technical reports, and published papers, along with pointers to relevant web sites and conference proceedings. Basic reading on MT is taken from various MT-related texts, such as Hutchins and Somers (1992), Arnold et. al. (1994), and Trujillo (1999). Readings on MT history are selected from Hutchins (1986) and Hutchins (2000). A chapter from Manning and Schütze (1999) is used for Statistical MT. Goodman and Nirenburg (1991) and Dorr (1993) are used for background on linguistic knowledge development. We also provide relevant technical papers from journals and conferences to augment the lectures described below. The course is divided into the following sections:

Introduction to MT: This section provides basic background on MT, including history, fundamental approaches, and examples of MT systems.

Modern Theory and Approaches for MT: This section includes separate lectures on various approaches to MT, including Direct and Transfer Methods, Interlingual MT, Example-Based MT, Statistical MT and Multi-Engine MT. Technical papers provided to the students include Mitamura et. al. (1993), Carbonell et. al. (1992), Brown, R. (1996), Collins et. al. (1996), Brown, P. et. al. (1990), Berger et.al. (1996), Frederking et. al. (2000), and Brown and Frederking (1995).

MT System Development: This section provides a software engineering perspective on MT development, with an in-depth examination of how to build a working MT system. We present various software engineering activities (e.g., Domain Analysis, Requirements Specification, MT Code Development, MT Knowledge Development), primarily illustrated with examples from rule-based, interlingual MT applications. We also discuss some of the modular software architectures that are used for analysis and generation (e.g. Reiter and Dale, 1995). For MT knowledge development, we use the KANT system (Mitamura et. al, 1993) to introduce a grammar formalism and transfer rules for analysis and generation. This section includes a homework assignment where each student builds a small MT system that translates from English to a language that he or she is familiar with.

Topics in MT: After presenting the basics of MT systems, the course covers additional topics in more detail:

- **Ambiguity and Ambiguity Resolution:** Various ambiguity issues are discussed, including lexical, structural and semantic ambiguities which cause problems for MT. Various ambiguity resolution methods (e.g., word sense disambiguation, structural disambiguation) are introduced. Different aspects of ambiguity resolution in the KANT system are covered in a case study (Mitamura, et. al. 1999; Baker, et. al. 1994).

- **Controlled Language (CL) Input/Output:** This lecture covers the definition of CL, the goals of CL, and different types of CLs. The history of CL is presented, followed by a discussion of the issues in design, development, and deployment of CL vocabulary and grammar for MT. We also discuss the success criteria for deploying CL in a real-world context (Nyberg, et al. 2003).

- **MT Evaluation:** This lecture introduces commercial goals and research goals in MT evaluation, placed in the context of the history of MT evaluation. Traditional evaluation approaches (e.g. the DARPA MT evaluations in the '90s) and more recent evaluation measures (IBM's BLEU Metric; Papineni, et. al. 2002) are presented. We also discuss the relationship between MT quality and associated postediting costs, and point out that evaluation in industry is often based purely on cost savings rather than an independent quality measure. The other factors related to MT system selection for a specific business solution are addressed in a separate lecture on the Business Case for MT (see below).

- **MT Workflow and Human Factors:** In this lecture we discuss the relationship between an MT system and its context of use, whether by an organization (workflow), by an individual (human factors), or within other software systems (embedded MT systems). The human factors lecture uses two optional readings, a standard human factors reference (Nielsen 1993) and the online Apple Human Interface Guidelines text[2].

- **Business Case for MT:** In this lecture we focus on how to judge the utility of MT systems from a business perspective. Depending on the characteristics of the problem to be solved (language pair(s), time to market, domain, translation volume, hardware/software integration, etc.) what is considered an effective MT solution may vary widely from customer to customer. Students learn how to calculate return on investment (ROI) based on business parameters extracted from customer interviews and estimates of how MT will speed up the translation process.

- **Commercial MT Systems:** This lecture presents commercial MT systems. We begin by discussing customer requirements issues. We then describe the range of MT development options, from fully custom systems, through customizable systems, to straight off-the-shelf systems with no customization. We describe six specific systems in some detail, covering a variety in terms of complexity and age.

- **Speech-to-Speech MT System Development:** We base this lecture largely on local research work in Speech-to-Speech MT, both the interlingua-based (Levin et al, 1998; Lavie 1996; Levin et al, 2000; Lavie et al, 2002) and EBMT-based (Frederking et al, 2000; Frederking et al, 2002) lines of research. We describe the effects of spoken input and speech recognition technology on the design, use, and evaluation of MT systems.

Term Project Presentation and Discussion: The last section of the class is devoted to student presentations of their term projects and group discussion of each project. The class presentation is made before the final written report is due, so that the students can refine their report based on feedback and ideas from other students and the course faculty.

4 How We Teach

The current MT course combines a series of lecture presentations (see topics in previous section) with homework assignments, examinations, and a multi-faceted term project. These activities are discussed in more detail in this section.

The homework assignments typically include a mixture of linguistic analysis (e.g. on lexical mis-

[2] `http://devworld.apple.com/techpubs/mac/HIGuidelines/HIGuidelines-15.html`

matches in a language pair) and hands-on creation and evaluation of MT systems. During 2003, the course included the following homework exercises:

- **Exploring MT systems on the web**. Students are asked to locate a set of MT systems or translation services available on the web. Students must a) categorize them according to important attributes such as type of service, application domain, languages, integration requirements, costing, etc.; b) Perform a comparative evaluation of two MT systems on the same language pair.

- **Building a simple transfer-based MT system**. Using the CMU Generalized LR Parser (Tomita, et al.,1988), Transformation Kit and Genkit (Tomita and Nyberg, 1988), students write a simple transfer-based MT system for two language pairs: English to English paraphrasing, and an additional target language chosen by the student. The same analysis grammar is used for both language pairs. The final deliverable includes an analysis of the system's performance and issues of transfer with the chosen language.

The course also includes two closed-book, in-class examinations (a midterm and a final). The exams contain primarily essay questions, with occasional questions that require simple mathematical calculations (e.g., a return on investment scenario for MT deployment).

The capstone activity in the course is the term project, which includes the following activities:

- **Directed Research**. Students work with their term project advisor to define a set of research activities, discuss possible approaches, and construct a research plan for the semester.

- **Written Report**. Each student must write and submit a final report detailing the research and development (if any) they accomplished during the course. If the project involved empirical analysis (e.g. of MT system performance), the experimental design and results are included. Students may also elect to do a survey of the recent literature in a particular area of MT research (e.g. problems of transfer for a specific language pair).

- **Public Presentation and Discussion**. Students present their project and results in a public presentation at the end of the semester. Each project is discussed by the instructors and other students in a discussion period that follows the presentation. Students receive feedback on their results and research methodology, and gain experience with public speaking.

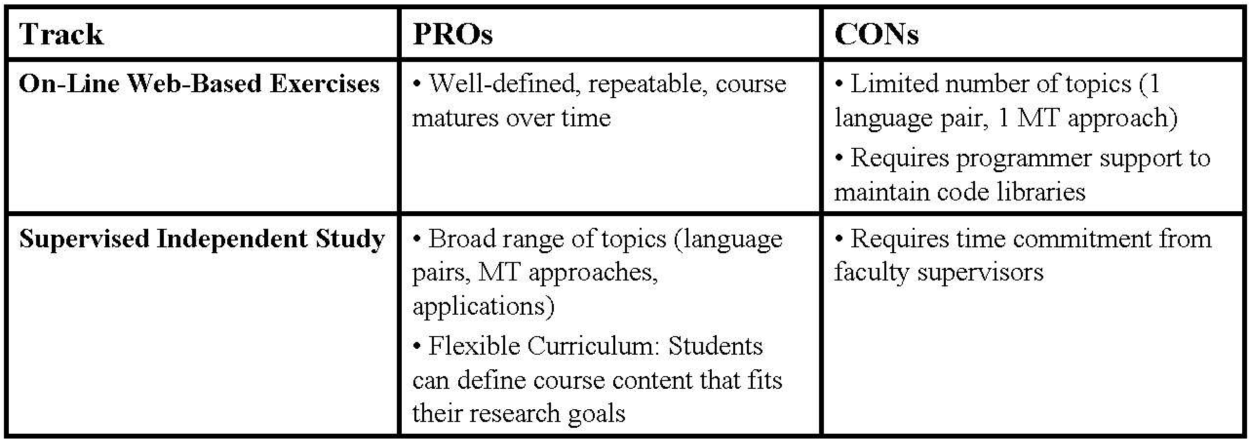
Examples of recent term projects include: a) "An Evaluation of the BLEU and NIST Metrics for Automatic Evaluation of MT" (comparative analysis of proposed standards); "English-Portuguese Translation Using a Transfer-rule Based System" (a solution for a particular set of transfer problems); "Current Research Projects on Translating to Asian Languages" (a survey of recent work); "Lattice Input for Speech-to-Speech MT" (technical project on extending an existing system); "A Framework for Pay-Per-Translation With On-Board EBMT System" (exploratory prototype).

Although traditional assessments such as homeworks and exams are still a fundamental part of teaching the MT course, we have tried to evolve the course in the direction of "learning by doing". Students gain experience with "real world MT", either by building MT solutions or evaluating existing MT research and development efforts. We feel that hands-on experience is essential if the course is to adequately prepare the student to work with MT technologies after they complete their degree.

5 MT Lab

In order to complement the MT lecture course with hands-on, in-depth exploration of various MT paradigms, we created a companion course, title 'Self-Paced Lab in MT Algorithms'. The lab course has two options or 'tracks' that may be selected by the students:

- *On-line Course Materials and Exercises*. Students may choose to complete a web-based course which includes exercises on parsing, generation, and semantic interpretation in a KBMT framework. Students are given sample code libraries and partially implemented systems to work with, so that the amount of coding required per student is appropriate. Some programming support effort is required to keep the



Track	PROs	CONs
On-Line Web-Based Exercises	• Well-defined, repeatable, course matures over time	• Limited number of topics (1 language pair, 1 MT approach) • Requires programmer support to maintain code libraries
Supervised Independent Study	• Broad range of topics (language pairs, MT approaches, applications) • Flexible Curriculum: Students can define course content that fits their research goals	• Requires time commitment from faculty supervisors

Figure 1: **The PROs and CONs of the two MT Lab tracks.**

code libraries running on platforms accessible to the students.

- *Supervised Independent Study*. With pre-approval from the course instructor, students may also define their own hands-on project, and select a project advisor from among the program faculty. Projects may explore a wide range of MT approaches (Statistical MT, EBMT, Multi-Engine MT, etc.), application areas (speech MT, mobile MT, web MT), and language pairs. Additional support comes from advisor's research team, if required. At the end of the course, students present their project in a presentation/demo for the entire class, which promotes sharing of research results.

When the lab course was first taught in the late 90's, only the on-line web-based track was available. More recently, students have been actively encouraged to propose independent project topics. The result is that the lab course content can be tailored to better meet the research goals of individual students. Students choosing the independent study track work closely with a faculty advisor, typically in the advisor's area of expertise. The incremental cost of adding this track is the time spent by the faculty members who contribute their time to the course as project supervisors. The PROs and CONs of the two tracks are summarized in Figure 1.

In a department with limited resources for teaching MT, it may be more feasible to base a lab course on pre-defined, web-based exercises with supporting code. Even when an independent study option is available, some students may still select the web-based option when they chiefly desire a highly structured, guided course. Nevertheless, it has been our experience that students prefer the more flexible approach of the supervised independent study. The additional motivation students feel for a self-defined project can often lead to a greater degree of accomplishment during the course. The problems and challenges they face in a self-defined project may be more difficult than those present in the pre-defined exercises, but afford a more realistic exposure to the realities of MT research and development.

6 Discussion

In this section, we present some of the issues we have experienced in teaching our MT course. Since most of the students at LTI have a computer science background, our course is designed for students who are able to work with existing tools (e.g. parsers, generators, etc.) effectively after a brief introduction. However, since we only recommend two courses prior to the MT course (see Section 3) and do not require them as absolute prerequisites, we sometimes teach students with a different background (e.g. linguistics) or levels or preparation (e.g. motivated undergraduates). This is a challenge for the instructors, and we sometimes find it necessary to offer special help sessions outside the lectures. Nevertheless, we have found that most students that are well-motivated and apply themselves are able to pass the course without difficulty.

The course lectures include various focused topics in MT; among them, the lecture on "Business

Case for MT" is unusual for our students, because it focuses on business issues rather than technology. Nevertheless, we feel it is important to make students aware of business perspectives on MT as part of a general education on MT as a practical area of application.

The most interesting part of the course is the term project, which covers a wide range of topics due to the varied interests of the individual students. Sometimes, MT is embedded inside a larger system to provide a multilingual capability (e.g. "Keyword Selection for Multi-Lingual Question Answering"). Some of the embedded applications of MT are often novel and sometimes unique; for example, a recent project explored "Intra-Language Matching of Proteins", where MT techniques were adapted to protein analysis. Since more recent applications of MT go far beyond technical document translation, we have adjusted the term project requirements to accept a wide variety of MT-related topics.

To date, the course has not given much emphasis to tools, environments and techniques for machine-aided human translation (e.g. postediting, translation memory, multilingual text processing, etc.). We plan to add material on these topics in the future. In general, we find it necessary to adjust the materials and topics in the course as the MT field evolves, language technologies improve, and new technologies emerge. Although our current MT course represents 7 years of development and refinement, we assume that the course will continue to evolve due to the changing nature of the field.

Bibliography

Arnold, D, L. Balkan, R.L. Humphreys, S. Meijer, L.Sadler (1994). "Machine Translation - An Introductory Guide", NCC Backwell Ltd., Oxford, UK. http://www.essex.ac.uk/linguistics/clmt/MTbook/HTML/book.html

Baker, K., A. Franz, P. Jordan, T. Mitamura and E. Nyberg (1994). "Coping With Ambiguity in a Large-Scale Machine Translation System" *Proceedings of COLING-94*, Kyoto, Japan.

Berger, A., S.A. Della Pietra, V.J. Della Pietra (1996). "A Maximum Entropy Approach to Natural Language Processing" *Computational Linguistics,* Vol. 22. Number 1.

Brown, R. (1996). "Example-Based Machine Translation in the Pangloss System". *Proceedings of COLING-96,* pp. 169-174, Copenhagen, Denmark.

Brown, R. and R. Frederking (1995). "Applying Statistical English Language Modeling to Symbolic Machine Translation". *Proceedings of TMI-95,* pp. 221-239, Leuven, Belgium.

Brown, P., J. Cocke, S.A.Della Pietra, V.J. Della Pietra, F. Jelinek, J. Lafferty, R. Mercer, and P. Roossin (1990). "A Statistical Approach to Machine Translation" *Computational Linguistics,* Vol. 16. Number 2.

Carbonell, J., T. Mitamura and E. Nyberg (1992). "The KANT Perspective: A Critique of Pure Transfer (and Pure Interlingua, Pure Statistics,...)", *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation,* TMI-92, 225-235, Montreal, Canada.

Collins, B, P. Cunningham, T. Veale (1996). "An Example-Based Approach to Machine Translation." In Proceedings of AMTA-96, pp. 1-13, Montreal, Canada.

Dorr, B. J. (1993). "Machine Translation: A View from the Lexicon", Cambridge, MA: MIT.

Frederking, R., A. Rudnicky, C. Hogan, K. Lenzo (2000). "Interactive Speech Translation in the DIPLOMAT Project". *Machine Translation*, Vol. 15. Number 1-2. Special Issue on Spoken Language Translation.

Frederking, R., A. Black, R. Brown, J. Moody, E. Steinbrecher (2002). "Field Testing the Tongues Speech-to-Speech Machine Translation System" *Proceedings of LREC 2002.*

Goodman, K. and S. Nirenburg (eds.) (1991) "The KBMT Project: A Case Study in Knowledge-Based Machine Translation", Morgan Kaufmann Publishers, San Mateo.

Hutchins, John W. and Harold L. Somers, (1992) "An Introduction to Machine Translation", Academic Press, San Diego.

Hutchins, John W. (ed.) (2000), "Early Years in Machine Translation", John Benjamins.

Hutchins, John W. (1986) "Machine Translation: Past, Present, Future", Ellis Horwood Limited, England.

Lavie, A. (1996) "GLR*: A Robust Grammar-Focused Parser for Spontaneously Spoken Language." PhD Dissertation. Technical Report CMU-CS-96-126, Carnegie Mellon University, Pittsburgh, PA.

Lavie, A., F. Metze, R. Cattoni, E. Costantini (2002). "A Multi-Perspective Evaluation of the NESPOLE! Speech-to-Speech Translation System." In Proceedings of Speech-to-Speech Translation: Algorithms and Systems Workshop at ACL-02, Philadelphia, PA.

Levin, L., D. Gates, A. Lavie, and A. Waibel (1998). "An Interlingua Based on Domain Actions for Machine Translation of Task-Oriented Dialogues." In Proceedings of ICSLP-98, pp. 1155-1158, Sydney, Australia.

Levin, L., D. Gates, A. Lavie, F. Pianesi, D. Wallace, T. Watanabe, and M. Woszczyna (2000). "Evaluation of a Practical Interlingua for Task-Oriented Dialogue." In *Proceedings of Workshop on Applied Interlinguas: Practical Applications of Interlingual Approaches to NLP*, Seattle.

Manning, C. and H. Schütze (1999). "Foundations of Statistical NLP", MIT Press. Cambridge, Massachusetts.

Mitamura, T., E. Nyberg, E. Torrejón and B. Igo (1999). "Multiple Strategies for Automatic Disambiguation in Technical Translation" it Proceedings of TMI-99, Chester, UK.

Mitamura, T., E. Nyberg and J. Carbonell (1993). "An Efficient Interlingua Translation System for Multilingual Document Production", in Nirenburg ed. *Progress in Machine Translation*, Amsterdam, IOS Press. Originally published in *Proceedings of Machine Translation Summit III*, 1991, Washington, DC.

Nielsen, Jakob (1993). "Usability Engineering," AP Professional, Boston, MA.

Nyberg, E., T. Mitamura and Huijsen (2003). "Controlled Language for Authoring and Translation" in *Computers and Translation*. Harold Somers (ed.) Benjamins Translation Library.

Papineni, K., S. Roukos, T. Word and W.J. Zhu (2002). "BLEU: a Method for Automatic Evaluation of Machine Translation", In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia.

Reiter, Ehud and Robert Dale (1995). "Building Applied Natural Language Generation Systems," *Natural Language Engineering*, 1(1).

Somers, H. (2001). "Three perspectives on MT in the classroom", MT Summit VIII Workshop on Teaching Machine Translation, Santiago de Compostela.

Tomita, M.(ed.), T. Mitamura, H. Musha and M. Kee (1988). "The Generalized LR Parser/Compiler: Version 8.1 User's Guide", CMU-CMT-88-MEMO. Technical Memo. CMU, Pittsburgh.

Tomita, M. and E. Nyberg (1988). "Generation Kit and Transformation Kit: Version 3.2, User's Mannual", CMU-CMT-88-MEMO. Technical Memo. CMU, Pittsburgh.

Trujillo, Arturo (1999) "Translation Engines: Techniques for Machine Translation", Springer-Verlag Series on Applied Computing.