ABSTRACT
In this paper we discuss transforming the newly created Capstone course in Computer Information Systems (CIS) at the University of New Hampshire at Manchester (UNHM) to mentor students in developing problem solving skills by immersing them in a real world research environment. UNHM is a commuter college, representing a non-traditional educational setting with a majority of students holding day jobs thus having limited free time. We use Automatic Speech Recognition (ASR) as the field of research, a cutting edge, complex and challenging technology that has a very appealing hands on nature where students can see tangible results of their work. Students learn to decompose problems, find solutions, self organize and establish leadership roles based on their skill sets and interests. This gives a unique opportunity to observe and guide the learning process that students use to solve a challenging research problem from that of a traditional external project oriented Capstone course where only milestones and end goals are seen. By exposing students to research, the goal of this project is to pique their interest in science and furthering their education beyond the undergraduate experience.

Categories and Subject Descriptors
K.3.2 [Computers and Education]: Computer and Information Science Education – computer science education, curriculum, and literacy.

General Terms
Experimentation.

Keywords
Capstone, research, speech recognition.

1. INTRODUCTION
The newly created Capstone course in Computer Information Systems (CIS) at UNHM, which started in spring 2011, offers students a faculty-guided project. The course exposes students to the rigors of speech recognition, giving them the opportunity to engage in solving real world problems, gaining invaluable experience along the way. We combine the Capstone course with an active research agenda in speech and plan to use it as a pilot program for a National Science Foundation (NFS) Transforming Undergraduate Education in Science, Technology, Engineering and Mathematics (TUES) [1] grant proposal to enhance students’ comprehension of scientific principles.

The purpose of the NFS TUES program is to improve the quality of science, technology, engineering and mathematics education at the undergraduate level. Since one way to excite students about science is to involve them in direct scientific discovery, using speech provides a rich area that encompasses all those aspects required by the TUES program. For a wide area of technological fields, speech can be applied to solve real world problems. Both its solution and applications are engineering problems that rely heavily on mathematics. To create a viable solution, good scientific principals need to be applied.

1.1 Inaugural Course
Having only 16 weeks in a one-semester course requires a staged approach. The inaugural project focuses on creating the platform and processes to enable active research. Specifically, students learn with the Carnegie Mellon University Sphinx Open Source Speech Recognition Toolkit [2], building tools to use the system on a set of servers running Linux. The eventual goal is to generate robust models needed in speech recognition to decode audio into text and to capture the techniques in creating these models by crafting tools to ease that process. Each subsequent course will expand on the previous, enabling more sophisticated research as the program develops.

Additionally, creating an experiment paradigm along with all tools required to generate a viable system is an important exercise that exposes students to all facets of modern research. Not only will the aim be to capture the techniques used to create the models, but to track and document performance results of those models for future comparison. This is a critical element in any experimentation-based field.

1.2 Future Direction
As the Capstone course matures the aim is to partner with small high tech firms to work on real world speech projects. Think of it as a low cost alternative to an expensive speech-consulting firm. Students would be key resources in solving problems towards specific project goals and would allow partners, both external as well as within UNHM a viable and cost effective solution to speech related problems.
Close ties with industry will enable placement opportunities for graduating seniors with high tech companies specializing in the field of speech. For non-seniors, the possibility for summer internships in industry is a very likely prospect, and a summer program as a follow up on the Capstone course will also be created to allow interested students a more in-depth and focused experience.

2. UNH Manchester
The University of New Hampshire at Manchester is one of six colleges of the University of New Hampshire (UNH) and the only one on an urban campus in Manchester, the largest city in the state of New Hampshire. UNHM has around 900 undergraduate students compared to the 9,000 students that make up the main campus of UNH in Durham. UNHM is a non-residential commuter college with a more diverse student population when compared to students from the main campus. Two-thirds of the students receive financial aid and on average are three or more years older than traditional college age students.

The large majority of UNHM students attend school full-time while working, approximately 20 or more hours per week. Full time student on average carry a course load of 16 credit hours, i.e. four courses, but can take a maximum of 20 credit hours in a semester.

2.1 Time Requirement
One critical aspect of succeeding at UNHM is for students to manage time wisely. Many students work between and around classes. Some students attempt to schedule multiple classes in a day. It should be noted that a majority of classes fall into a single 3-hour time slot, so loading up three classes in one day potentially creates 9 hours of lecture for a student.

Even taking two classes in a day is almost equivalent to a full workday in lectures alone. Add to that about 10 hours of outside class time per course, a 4 course average load, plus 20 or more hours of work, and a student is looking at close to 80 hours of time their mind is engaged in some intellectually challenging activity.

Formulating a class that engages students can be somewhat difficult. These time requirements need to be kept in mind during any course, but especially one that challenges students to learn in a new way, out of their element. One-on-one face time with faculty is a key element to alleviate potential problems students face and is one mechanism that is heavily promoted to students.

3. SPEECH AND EXPERIMENTATION
To put the course into context, we give a brief description of speech recognition technology as well as the experiment paradigm that was put forth to the students. This paper is not the proper forum to discuss speech recognition in depth and thus references are provided for further reading [3][4].

Simply put, speech recognition is the process of taking recorded audio of a conversation and transforming it into text of what was said.

The basic solution is to first convert the recorded speech to phonetic units, or phonemes, which we then map to words via a dictionary of phonetic spellings. Concatenating the words together forms the sentence of what was said. It is a simple process to describe but one that is difficult to implement, primarily due to variability in not only how the phoneme is pronounced but also in the environment it was spoken in, as well as the speaker, dialect and language itself.

The key is to recognize phonemes, and for this we use a mathematically representation, called a Hidden Markov Model (HMM) to account for each phoneme and deal with its many forms of variability [5]. A common topology for an HMM in speech recognition is a left to right model with self loops and skips as shown below.

![Speech Features](image)

The acoustic model, as it is called, transitions from beginning to end as it aligns its states to the best possible sequence of features of the audio signal. A perfect alignment would match exactly a single phoneme that represents that audio signal. In practice, many errorful alignments are found, with the optimal match depending on the overall alignment of the entire audio signal, which is comprised of many phonemes. Although the number of possible alignments can grow quickly, pruning based on various criteria reduces the complexity of the searchable space.

3.1 Building and Using Models

There are primarily two tasks in speech recognition: building models from large amounts of data, and using those models to come up with the desired answer of what someone was recorded as saying. We refer to the former as “training” and the latter as “decoding.”

3.1.1 Training
Training uses a collection of recorded audio data and a set of associated text transcriptions, called “truth” to build an individual model for each phoneme. This is achieved through a method of alignment using numeric features extracted from audio to the models given in truth. Since truth is transcribed text and that text is expanded via the dictionary of phonetic spellings to a sequence of phonemes, we simply work to optimally align all features to their respective phonetic models [6].

It’s a simple process to describe but enormously time consuming and data intensive. For any reasonably robust system it could take many weeks for training to complete while running on multiple machines. It should be noted that setting up training is far from a trivial task, even for a veteran speech researcher.

3.1.2 Decoding
Decoding is a much simpler step and generally one wants to perform this in real time. Similar to training, the task is to align a set of features from audio input to a set of possible models. However, there is no truth to guide which models to use; therefore every combination of models is possible. There are reasonable techniques to make this work in real time and this task only requires the models that were built in training. Decoding is
therefore a much more straightforward and easier to understand task [7].

3.2 Types of Research
There are two types of research paths taken in speech recognition: improving the performance of the system, which we term as “theory,” and developing new and unique ways to use an existing system to solve real world problems, which we term as “applied.” Working on pure theory is beyond the scope of an undergraduate program, regardless of the type of course. However, applied research is just as challenging and just as engaging.

A possible applied research project, for instance, would be to transform Sphinx generated acoustic models into ones compatible with Google’s Android recognizer. This could be viewed as simply solving an interface incompatibility, but in reality, the complexity of a speech system forces anyone tackling this problem to have a deeper understanding than merely looking at data formats. Understanding what the topology of models in each system are, what methods were used to train them and what tradeoffs were made, are critical to achieve a set of workable models among the two speech systems. In fact, this is a project that future Capstone students will tackle.

3.3 Experiment Paradigm
In both applied and theoretical research, an often overlooked step is building a mechanism to record experimentation. Even at professional organizations, all too often a simple email repository is used to manage a history of trials and their associated results. This simply isn’t enough to ensure a clear and concise audit of what has been done. In experiment-based research, it is crucial to be able to compare what you are doing now to what you or someone else did in the past. It is therefore important to be rigorous in capturing a detailed record.

Creating a database that keeps a historical record of experiments and asserting an accounting system on them will help ensure that any new result can be put into proper context. Serially numbering experiments to identify them is a simple yet powerful mechanism to ensure traceability of work done. These are important aspects of not only speech research, but of any type of research, and will be interwoven within the Capstone course throughout.

4. COURSE OUTLINE
Similar to many Capstone courses, this course is designed as a self guided project with a faculty advisor overseeing student progress. Students are required to formulate a proposal where they plan their semester work and write a final report detailing the project and results attained. The advisor oversees the project, guides and mentors students and insures progress is on track during the semester. Where it differs from most senior project courses is that the focus is on research and the project is internal and in the advisor’s field of interest.

Specifically, the project focused on applying the Sphinx speech recognizer to generate a set of baseline models. Students were tasked with learning about Sphinx through online resources, followed by installing and configuring the latest release on a queue of machines running Linux.

Students were also introduced to a large data set, the Language Data Consortium (LDC) Switchboard corpus, comprised of 100 hours of recorded audio along with an accompanying set of annotated transcriptions matching the audio. Additional data sources, such as a phonetic dictionary and language modeling data critical to the process, had to also be located from various online sources which students had to go out and find.

The ultimate goal is to develop infrastructure for doing robust speech research and ease the knowledge transfer between student groups across different semesters. Documentation and development of tools to simplify these processes would be crucial outcomes in the success of the project.

4.1 Self Organizing
One important element in the makeup of the course is for students to take the lead in seeking out roles they are both interested in and feel best able to perform. There were two distinct stages of the initial project startup, system setup followed by installation and configuration, which presented students with opportunities to self organize.

4.1.1 Initial System Setup
Before any speech related tasks could begin, a group of 10 dual processor Dell PowerEdge servers needed to be configured. One machine, the lone PowerEdge 2650, was chosen as the file server and configured as the data store for the Switchboard corpus. The file server was connected to the UNH network for access to the Internet. An additional set of 9 queue machines, all Dell PowerEdge 1750’s, were configured on their own internal switched network with a single connection to a second network card on the file server. The queue machines do not have independent access to the Internet but could access it by bridging through the file server. This was done both as a cost cutting measure and also to avoid unduly congesting the outside network during large data traffic within the internal queue. It also required double logins from the outside world to the queue machines by first accessing the file server and then each individual queue machine.

4.1.2 Installation and Configuration
All machines run Linux and each of the 9 queue machines have a full installation of the Sphinx speech recognition system, language modeling tools and scoring tools along with a full complement of necessary dependencies to successfully install and run those systems.

Additionally, network file server, Kerberos login service and source code revision control software were added to the file server to allow sharing of data, user access and a source repository for software tools written. Finally, configuring data directories, user accounts and group permissions rounded out final tasks required to get the hardware and software systems up and running.

4.1.3 Teams and Leadership
Both during initial setup and during installation and configuration, teams formed to handle the various tasks. Leaders for these teams were not directly chosen but emerged as the class progressed. Leadership was a loosely defined role of organizing resources and those students eager to accomplish a specific task were easily drawn to that responsibility.

It is important to note that team leaders held no significant role of authority over their peers. Their job was to reach consensus among group members and negotiate a timeline. Although no approval was required by the faculty advisor, it was strongly recommended to allow for cursory review of the final schedule.
4.2 Risks and Reward
Since motivation is a key factor to student participation in the project, the relationship of risks versus reward was important to understand in order to ensure that best possible outcomes were achieved. Because individual work ethic and achievement needed to be maintained, we applied both peer reviews as well as tangible project goals in determining each student’s final grade.

4.2.1 Grading
Class grading was split evenly between peer review and tangible results. Half of each student’s grade was determined by reviews received by fellow class members. Each student had the opportunity to rate fellow students, ranging from ones that excelled in the program to ones that did not contribute to the project. A student could also choose to skip rating individuals that they did not work with directly, although this option should have been used as a last resort.

Along with this peer rating, students also had to write a brief description in defense of each rating given. These were reviewed by the advising faculty member and anonymously available for each student to view upon request. Final peer review grades were determined by averaging all ratings a student received from their peers and accounting for any mitigating circumstances that arose. Care was taken to avoid bias through personal conflicts by ensuring the rating given and description in its defense were purely based on objective measures.

Tangible results were rated by the faculty advisor and divided into quality of both the proposal and final report for the project, each worth 20% and shared as a group grade among the class, and individual student communication on progress, worth 10%. That 10% was divided among weekly log entries through the entirety of the course, enabling the faculty advisor to keep track of what each student was doing and allowed for assistance when a particular student went off track.

4.3 Project Goals
Initial project goals included generating a complete and robust set of baseline Switchboard acoustic models with accompanying language models. It was critical to document the process of using the system, create a database to capture experimental results and develop tools to automate as many of the tasks required to re-create those models and allow different configurations, perhaps with a different data set or different set of input parameters.

4.3.1 Goal Assessment
An important aspect of this course is for students to come to the realization that continually reassessing those goals is an important part of project management. Team leaders were tasked with assessing the state of the project as it related to its goals and adjust schedules accordingly. At times this required renegotiating new time lines with team members as well as communicating with other teams. Though no specific schedule was applied to this process, it became apparent at an early stage that this readjustment would become an integral part of scheduling.

5. COURSE OBJECTIVES
As stated in the introduction, the course objectives are to excite students about Science, Technology, Engineering and Mathematics (STEM) disciplines. We use speech recognition both because it is a cutting edge technology that exposes students to skill sets that will be useful in their careers and because of its hands-on nature where students can easily see the results of their efforts when building a system that recognizes what was said.

One interesting element of speech is its complexity. It has a high learning curve to not only understand how to configure and run an actual system, but understanding the underlying technology requires a solid mathematical background. It’s important to address this complexity with students and guide them in ways to solve problems.

What sets this course apart from many undergraduate research focused Capstone programs [8][9][10] primarily is its team approach to research. Students work toward a common goal, and though they contribute both individually and as members of specialized teams, success is dependent on the class as a whole achieving measurable results. The aim is to immerse students in research within a team setting where they learn to work, communicate, and solve problems together in a collaborative environment. It is developing these skills that will be among the most important lessons students learn through the experience.

5.1 Mentoring
An important component in creating a successful course is insuring that students work in a non-combative environment. Both working in a complex technological field and engaging in research, by nature an unstructured environment, can be a recipe for failure unless a framework is set up at the start to allow students to easily find help.

A close mentoring relationship between faculty advisor and student is important and needs to be nurtured from the outset. This is especially true in the type of environment that students face at UNHM. Students seldom have much free time with both internal and external demands on their schedule so it is critical to streamline communication channels and reduce unnecessary overhead whenever possible. Some examples of this include: scheduling office hours immediately before and after class, and having online office hours with a guarantee of 24 hour turnaround for questions so that students don’t need to make additional trips to campus to avoid a lengthy, time consuming round trip commute.

6. OBSERVATIONS
This paper is about lessons learned from the first instance of an undergraduate research course in speech recognition at UNHM. These lessons came through both successes and failures. Some seem obvious after the fact, but it never the less was as much of a learning experience for the advising faculty member as it was for the students.

The class had eight students enrolled, split evenly between majors from Computer Information Systems (CIS) and Engineering Technology with a concentration on Computer Technology (ET-CT), the latter being similar to a Computer Engineering program at other institutions. CIS majors focus on standard IT topics with courses in database systems, web authoring, networking and operating systems. There are two required programming courses; an introductory one and one that focuses on web development technology. All ET-CT students matriculate from the community college system and spend their junior and senior years at UNHM focusing on electrical engineering and computer topics. The computer specific courses are an additional programming course covering data structures, an elective for CIS students, as well as a course in programming languages.
6.1 Starting Out
At the onset of the course students had to break into teams based on their interest and skill sets. There were both hardware needs and software needs. As previously discussed, hardware needs required configuring 10 Dell PowerEdge computers as a cluster of queue machines isolated from the main network with one of those machines acting as gateway to the outside world and providing file service to the rest. Students with Linux experience volunteered to set up the course’s hardware.

Software needs came in two forms, software that was apparent, such as CMU’s Sphinx, and software that was yet to be determined, such as a good scripting language to help build support tools. Additionally, infrastructure software, such as a database system to store and retrieve documented experiments and their results also needed to be found or developed. Teams fell along those three lines.

At this stage of the project, things seemed to go smoothly, primarily because students were comfortable solving directed tasks such as this. Configure hardware and locating, downloading and installing software, these were tasks that a 3rd or 4th year student in CIS should be comfortable doing and indeed, students took to them easily.

Students struggled when it came time to configure the CMU Sphinx speech recognizer to start running various scenarios. A speech recognizer is not an out of the box turnkey system that one can run a simple set of commands to achieve some result. To alleviate this, weekly seminars were given for both a theoretical and practical overview of how a speech system creates and stores models and how it uses them.

6.2 Course Adjustments
A research-focused course tends to be less structured and somewhat open ended compared to a more standard lecture format. A fundamental problem in an open-ended course is time management by students. It was quickly apparent that for various reasons students put off work on assigned tasks until the day before the course met.

Initial status updates were done in document format and students were advised to bring copies to weekly meetings to go over tasks. This caused a significant bottleneck in advancing the course goals as issues were held off until the class had a chance to meet.

Primarily, as students got stuck on problems or approached them incorrectly, no time for correction was available until the end of the week. In a true research group, over a longer period of time, say a year, there is ample opportunity to meet weekly and make corrections, but in a one-semester course, a few missteps could cost the overall project some of its major milestones.

In order to insure students spent more than once a week working and thinking about the course, a minimum of 4 weekly status updates, on separate days, was added as a new requirement. Half of those could simply be reading other student’s logs, but the other half should reflect significant work done. A media wiki page was used to host the status logs and the history feature could easily track whether each student met the requirements of 4 separate activity periods.

6.3 Student Apprehension
Even though students from both majors had extensive experience in project-focused courses built into the curriculum, most of these courses had a specific topic and a specific goal. Perhaps a database course with a real world client requiring online access to various types of data, or a software engineering course requiring the design of a geographical mapping client that access an automatically generated database.

A research project gives undergraduate students their first opportunity to experience a unique environment with loosely defined goals [11]. This causes quite a bit of apprehension among students as concerns arise about how to accomplish tasks that aren’t well defined or clearly laid out. To alleviate this, weekly one-on-one office sessions were introduced where students could seek help working through some of the more complex issues involved in understanding the technology.

Although the system provided ample documentation, it was written for a target audience that understood the fundamentals of speech recognition. A student with only a cursory understanding through a handful of seminars would still be expected to struggle. Making this realization clear further relieved stress students felt and helped move the learning curve of the course along.

6.4 Student Frustration
Being immersed in a highly challenging field for an entire semester can become very distressing. Compounding this with a general perception of lack of tangible progress or even failure can distort the efforts of the course in having students gain valuable insight on such a real world tasks. This became apparent early on in individual log entries where, under the appropriately entitled “concerns” section, students would articulate their frustrations of not being able to easily find solutions to their set of tasks.

Although the goal is to immerse students in real world experiences and then assist them in overcoming obstacles, it is important that this is done in a measured approach. The danger lies in losing students because they feel a sense of hopelessness at the impossibility of achieving an uncertain task. It is thus important to closely monitor student progress and insert oneself when it is apparent that a major road block has been hit.

At those times the opportunity for students to learn is tremendous. It is important not to give the answer to the student but help find it with them. Simply stating that they need to find it elsewhere is not enough, a lesson learned from some early exchanges. Having to sit there and demonstrate this task in a one-on-one setting seemed the most helpful mechanism to achieve the desired goal of jump-starting a stalled student.

6.4.1 What is Research?
A common complaint among students was a perception that they weren’t doing research. Where were the new discoveries or the adventure in finding them? The reality of research is that it can at times seem mundane and filled with too many steps. It is of course the goal that a researcher keeps in mind and the possible result of achieving that goal. However, it was a legitimate question that arose frequently and needed to be addressed.

One interesting aspect of the research question is the mislabeling of the term itself. Many of the students would consistently label the work they were doing as research, but most of it fell into categories of searching for an easily found answer. For instance, if you are researching what the best midsize car is, it is not really doing research. The answer is easily reproducible and depends on the criteria of what the word “best” means to you. It is certainly a good exercise in techniques needed to solve problems but hardly something one could consider fitting under the criteria of
Scientific research. Highlighting this point didn’t necessarily resolve the issue but it did start a dialog on the differences.

6.4.2 Cutting Edge
What set the work this class was doing apart from experiences that students had in other courses at UNHM was the technology. The complexity of a speech recognizer can be a barrier for entry into the field and students pretty quickly accepted that as they were faced with those complexities. What was an important element in motivating students was the cutting edge nature of the technology. Although one can argue that a compiler for an object oriented programming language in its entirety is a complex system, its use as well as what students are exposed to (i.e. a few command line options to configure it) is relatively simple.

For speech, this is not the case. Current speech recognizers are not designed to be used as turnkey systems. Unlike a compiler, there isn’t a single executable you can run that makes the entire system work. In fact, one of the primary tasks of this Capstone group is to eventually generate a close approximation of a turnkey system that is well documented and easily used.

The motivation for students, however, is less in the complexity and more in speech’s cutting edge nature. It’s easy to point at speech applications in our society; you simply need only to pull out a smart phone and you can download an application that has a speech-enabled feature. Enabling students the opportunity to learn how that works and the possibility of adding to that technology, perhaps developing their own, is something that excites students and allows them to take on the complex nature of speech.

6.5 Improvements
Numerous lessons have been learned from the first run of this course. A better mechanism to collaborate on documents, improved ways to share weekly status updates in real time and the introduction of seminars on speech to give students a more formal look at the technology were all important additions to improving the course. Another important aspect was clarifying what the Capstone course was really all about.

The inaugural course was not required in the CIS curriculum, although subsequent ones will be mandatory for all majors before graduating. This meant that students were recruited to take the class and perhaps had a misconception of what to expect. Most thought they would be taught about speech recognition as in a traditional class. It came as a surprise to many that most of the material they were expected to understand had to be self-taught. Communicating that point clearly took several iterations.

7. FUTURE PLANS
As the course matures there are positive upsides in several areas. First, students will continue to develop tools and document processes to ease the building and use of speech models. There will always be a learning curve with each course as a new set of students will struggle to learn the complexities of speech, but as each course builds on work of the previous one and mechanisms are fleshed out, that learning curve should shorten each year.

There is potential for interesting and relevant research that a group of undergraduate CIS students can accomplish. Documenting and automating the Sphinx speech systems along with perhaps bridging to other speech systems would be important contributions to the community. An important aspect of research is not only being on the leading edge, but improving systems and processes so that other researchers are able to push the envelope.

7.1 FOSS and HumIT
Since CMU Sphinx is Free and Open Source Software (FOSS), an opportunity of integrating with the open source software mentoring approach exists [12]. This would further deepen the knowledgebase, expanding the possible ways to learn, beyond the faculty advisor and allow students to be mentored by many experts in the field.

Specifically, being a Capstone course in CIS, the focus on IT issues in open source software is a natural fit and something that students are currently working on to get the speech recognition system running internally. By pushing these processes back to the online community, it would force students to adopt more rigorous practices as the community gate keepers would set a higher bar than even the faculty advisor. By that same token, the online community would see an improved Sphinx speech recognition system with better automation and more detailed documentation which would open speech research up to many more individuals.

8. ACKNOWLEDGMENTS
I want to thank my fellow colleagues for helping with this course. I’m extremely grateful to my students who struggled to grasp the difficult concepts in a hectic and challenging environment that is UNHM. My hope is that I have piqued their interest to further investigate the field of speech and explore avenues of research.

9. REFERENCES
[2] CMU Sphinx - Speech Recognition Toolkit, Carnegie Mellon University, School of Computer Science, Pittsburgh, PA,