European Masters Program in Language and Communication Technologies (LCT)

Module Handbook for Prospective Students

October, 2012

Contents

1 What is LCT?

Language and communication are so fundamental to human activity that it is not at all surprising to find that Language and Communication Technologies (LCT), that is, technologies that have spun off from the scientific study of these two areas, have an impact on all major areas of society, including health, education, finance, commerce, and travel. The evidence for this claim is as tangible as the nearest Google-equipped internet browser, a pervasive and indispensable piece of technology that lies properly on the intersection between language and communication.

The relevance of LCTs to our daily activities explains why they have been with us almost as long as computers. It is no accident that one of the first problems to be studied in the 1950s, shortly after the invention of computers, was an LCT problem, namely the machine translation of human languages. (A machine translation system is a program that takes as input text in one human language, say French, and automatically translates it into a text in another language, say Dutch.) The huge amounts of funding poured into machine translation testifies to the perceived importance of the field, right from the beginning.

Language and communication are both necessary for the transmission of information in human situations, and although both disciplines have been studied since the Greeks, it took until the middle of the 20th century for scholars to develop theories of language and communication based on scientific methods. In the case of language, it was Noam Chomsky who redefined the goal of linguistics as a quest for a formal description of language generating machinery, whilst Claude Shannon and Warren Weaver provided a mathematical theory that linked communication with information.

Inevitably, computers and related technologies have provided a physical and conceptual framework within which such scientific studies can be pursued. Indeed, this framework has been fruitful on a number of levels. For a start, it has given birth to a new discipline, known as Natural Language Processing (NLP), or Computational Linguistics. This discipline studies, from a computational perspective, all levels of language from the production of speech to the meanings of texts and dialogues. And over the past 40 years, NLP has produced an impressive computational infrastructure of resources, techniques, and tools for analysing sound structure (phonology), word structure (morphology), grammatical structure (syntax) and meaning structure (semantics). As well as being important for language-based applications, this computational infrastructure makes it possible to investigate the structure of human language and communication at a deeper scientific level than was ever previously possible.

Moreover, NLP fits in naturally with other branches of computer science, and in particular, with Artificial Intelligence (AI). From an AI perspective, language use is regarded as a manifestation of intelligent behaviour by an active agent. The emphasis in AI-based approaches to language and communication is on the computational infrastructure required to integrate linguistic performance into a general theory of intelligent agents that includes, for example, learning generalisations on the basis of particular experience, the ability to plan and reason about intentionally produced utterances, the design of utterances that will fulfil a particular set of goals. Such work tends to be highly interdisciplinary in nature, as it needs to draw on ideas from such fields as linguistics, cognitive psychology, and sociology. All in all, LCT, which draws on all these influences, is one of the most interesting and vibrant areas of research in computer science.

2 Why is LCT important?

But why is LCT important? There are three main reasons. First, LCT is opening the doors to deep scientific understanding of human language and communication. Second, LCT plays a role in the development of webbased technologies, a role that will become increasingly important in coming decades. Third, LCT will play a key role in the development and maintenance of multilingual infrastructure, and for this reason LCT has a special significance for Europe.

The science of human language and communication. The use of language is central to much of our activity — indeed, it is so central that the ability to use language is often viewed as essential to human identity. Nonetheless, until the last half the 20th century, there was little scientific understanding of how

human languages actually worked. Poets used language to great effect, and philosophers speculated about it, but there was little concrete scientific understanding of how it functioned.

With the benefit of hindsight, this is hardly surprising. The structure of human language and communication turns out to be intrinsically complex. We use language easily and naturally, yet when we examine it closely, it becomes apparent that the structure of human languages are highly intricate. Indeed, they are arguably the most complex naturally occurring abstract structures on the planet.

In the 1950s, the first accurate maps of these structures began to be drawn up. In particular, Noam Chomsky showed that human languages were inherently open-ended and infinite — and at the same time he devised the basic tool, the formal grammar, needed to tame this infinity. Chomsky's insights, which initially concerned grammatical structure, were extended by linguists and logicians to all other level of language. In particular, with the work of the logician Richard Montague in the 1970s, the first mathematical account of how sentences could carry meaning was obtained.

It is now becoming increasingly apparent that exploring the scientific paths opened by pioneers such as Chomsky and Montague will require intensive computational modelling. Language and communication both have a rich structure, and unaided introspection will not yield the deeper insights about it that we need. In particular, in the 1990s it became clear that statistical methods were needed to reveal important aspects of linguistic structure that had previously been overlooked; in effect this work built a bridge between the symbolic tradition initiated by Chomsky and the stochastic tradition initiated by Shannon and Weaver.

Modern LCT is based on this dual tradition of symbols and statistics. This means that nowadays research on language requires access to large databases of information about words and their properties, to large scale computational grammars, to computational tools for working with all levels of language, and to efficient inference systems for performing reasoning. By working computationally it is possible to get to grips with the deeper structure of natural languages, and in particular, to model the crucial interactions between the various levels of language and other cognitive faculties.

Applications of LCT. 25 years ago, information stored in computers had two predominant characteristics: it would have originated on paper and would have arrived at the computer via a process of transcription. Once transcribed, it was stored centrally on a mainframe computer, most likely to be stored in the form of a database.

Today the situation is very different. Most if not all of the information arrives electronically, so there is no need for transcription in the old sense. Second, most of the information used by companies and organizations is material in ordinary human languages (reports, customer emails, web pages, discussion papers, text, sound, video). It is not material held in traditional databases.

Now, when people see text, they do, by and large, understand its meaning. However, when computers see text, they get only character strings (and perhaps HTML tags). This is a major problem, since we'd like computer agents to see meanings and be able to process text intelligently.

Already applications with this kind of ability exist. Forty years ago, machine translation was a dream — nowadays it is a reality, as users of the Babelfish website can testify (see http://babelfish.altavista.com/). Of course, as the Babelfish example shows, the challenge is not just to create language-based applications, but to create robust and widely usable applications. Babelfish certainly handles simple texts with impressive ease, but longer texts, especially those containing idiomatic language, it often handles badly. Deeper insight into machine translation, the sort of insight provided by ongoing research into LCT, will be needed for the next generation of machine translation systems.

And with the exploding importance of the internet, it will become increasingly necessary to have not only better machine translation systems, but also to have other language-driven web-based technologies: systems that can search the internet to find accurate answers to questions posed to them in ordinary human languages, systems capable of handling sophisticated customer queries, and voice recognition systems capable of robustly handling different users without training. The techniques and tools pioneered by LCT researchers will play a key role in the development of these technologies.

Multilinguality. Machine translation is the most obvious example of the relevance of LCT to multilinguality, but it's not the only one. Multilinguality in the deepest sense means making easy access to information possible for all human languages. This means, for example, that intelligent question answering systems, and other languagebased technology, should be made available for as many languages as possible.

Many hurdles need to be overcome before this ideal can be achieved. The most basic hurdle is this: although LCT is now a highly developed subject, it has not been developed equally for all languages; in particular, there has been a heavy bias towards English. As we said earlier, research in LCT nowadays depend on access to large amounts of knowledge about words, to sophisticated computational grammars, and to detailed statistical information about language structure. But these resources are often poorly developed, even for major European languages (for example, there is currently no large scale computational grammar for French) and for many languages they do not exist at all.

Achieving genuine multilinguality is an activity that will require a great deal of practical and theoretical work. While a lot can be learned from the experience of developing tools and resources for English, the fact remains that different languages pose different challenges. For example, techniques for grammatical analysis that work well on English (which has a relatively fixed word order) often do not adapt straightforwardly to German (where the word order is much more flexible). Moreover, many statistical techniques which work well for English need to be heavily modified to cope with Romance languages. But if language-based technology is to be developed for languages other than English, this work needs to be carried out. One of the most important aspects of he European Masters in LCT is that it provides training in the skills that will be needed to do this.

3 The European Masters in LCT

The main goal o f the European Masters in LCT is to train new generation of research-capable language technologists, who are familiar with both the linguistic and computational underpinnings of modern LCT. Crucially, this training is offered in a flexible way that offers access both to different approaches, and to LCT work on a number of distinct European languages.

As you can see by examining the tables on the following pages, the European Masters in LCT provides training in core topics in the areas of Computer Science and Language Technology. But its structure also makes access to the respective strengths of individual participating institutions possible: there is room within the course specification for partners to use their discretion both in the choice and organisation of taught units and of project work. Moreover, between them the partners offer access to work on a variety of European languages: languages from the Romance, Slavic, and Germanic groups are all represented.
The course structure is divided along two axes: Computer Science/Language Technology and Core/Specialized.

A minimum number of ECTS credit points for each of the quadrants so formed is required, as is shown in Figure 3.

Area Specialized	**Core**	**Specialized**
Language Technology	24 credits	at least 4
Computer Science	24 credits	at least 4

3.1 The Core Courses

Figure 3.1 shows the structure of the core part of the European Masters in LCT. The three modules in the Language Technology part provide a firm basis for further work in LCT, and in particular for the wide variety of options offered in the advanced courses. The Foundational module introduces the two major paradigms used in modern LCT (that is, the symbolic paradigm which stems from the work of Chomsky, and the statistical paradigm which stems from the work of Shannon and Weaver). It also provides the needed background material in linguistics and related areas. The two other modules then focus on two different levels of LCT work. One concentrates on the computational treatment on the grammatical level: word structure (morphology) and sentence structure (syntax). The other concentrates on the level of meaning, covering such key topics as the syntax/semantics interface and the computational treatment of dialogue.

Similarly, the three modules in the Computer Science part provide a firm basis for later work. The chosen topics are fundamental to computer science, but in addition, they are also of importance to LCT. For example, nowadays XML is standardly used for encoding large-scale linguistic resources, logic and

the theory of computability are central to modern work on inference, and the theory of formal languages is the theoretical basis for modern work on parsing. All in all, these modules provide training in the fundamental tools needed to give a computational realisation of the ideas of LCT.

	Module	**Topics**	**Credits**
Language Technologies (24 ECTS)	Methodologies	Statistical methods, symbolic methods, cognition, corpus, text and speech, foundations of Linguistics	6
	Computational Syntax and Morphology	Finite state techniques, probabilistic approaches, formal grammars, tagging, chunking, parsing	9
	Computational Semantics, Pragmatics and Discourse	Syntax-semantics interface, semantic construction, dialogue, ontologies, formal semantics	9
Computer Science (24 ECTS)	Data Structures, Data Organization and Processing	Algebraic data-types, relational databases, semi-structured data and XML, information retrieval, digital libraries	6
	Logic, Computability and Complexity	Logic and inference, automata theory, computability theory, complexity theory, discrete mathematics	9
	Formal Languages and Algorithms	Formal grammars and languages hierarchy, parsing and compiler design, search techniques and constraint resolution, automated learning	9

3.2 The specialized Courses

Figure 3.2 shows the structure of the specialized courses. The specialized courses build on the introductory courses and offer a wide range of options, reflecting the different specialities and competences of the different partners. Between them, the partners are able to offer insight into all the key areas of modern LCT, and to do so in a variety of linguistic settings.

	Module	**Topics**	**Credits**
Language Technologies	Specialized	Machine translation, Information retrival, Question answering, Speech recognition, Speech synthesis, Models of human language processing and understanding, Psycholinguistics, Multimodality, Language Resources, Computatioanl Semantics, Formal Semantics, Inference in NLP	at least 4
Computer Science	Specialized	Artificial intelligence, Knowledge representation, Automated reasoning, Semantic web, Intelligent and multi-modal interfaces, Cognitive modelling, Computational psychology, Neural networks, Machine learning	at least 4

The European Masters in LCT is more than the sum of its parts. For a start, the range of courses the Masters makes available simply isn't available at any single university. Moreover, the European masters offer the opportunity to study LCT in different countries, with researchers who are interested in different languages. That is, as well as covering a wide-ranging introduction to the ideas that are shared by all researchers in LCT, it also gives the student to compare and contrast different approaches to LCT research. Multilinguality is built into the structure of the degree.

4 Where should I go?

The key point to ensure is that you have covered the required topics and gained sufficient ECTS credits. Within these constraints you have a lot of flexibility. For example, you might want to spend your first year at Bolzano, gaining background in computer science, and then follow up with a year in Saarbrücken, to concentrate on pure LCT.

The program ensures that all students receive a common education that covers the core topics in LCT, but their specific scientific profile is shaped by the pair of Universities they choose for the first and second year of their studies.

The partners of the consortium can be broadly classified in Language Technology partners (Saarland University, Charles University in Prague, University of Groningen) and Computer Science partners (Free University of Bolzano-Bozen, University of University of Lorraine, University of Malta, University of Melbourne and Shanghai Jiao Tong University). This distinction yields two initial scientific profiles (major in language technology/computational linguistics or major in computer science) depending on the choice of the University for the first year. These scientific profiles are further refined through the choice of the university for the second year, in terms of the research specialisations of each of the research centers.

Please note that your preferences will be subject to approval.

5 Admission and Funding Procedures

5.1 Timetable and Enrollment

Common admission criteria Applicants must have

• a Bachelor degree or equivalent in the area of (Computational) Linguistics, Language Technology, Cognitive Sciences, Computer Science, Mathematics, Artificial Intelligence, or other relevant disciplines, subject to the consent of and approval by the Consortium's admission committee (cf., Appendix "Memoranda of Understanding" (MoUs)).

• Proficiency in English (for non-native speakers) must be certified by one of the following: TOEFL score of at least 250 (computer-based test) or 575 (paper-based test) or 79 (internet-based test); Cambridge Proficiency Exam – pass; Cambridge Certificate in Advanced English – B; IELTS – > 6.5.

Application procedure Applicants apply by submitting a completed application form online available at http://lct-master.org/registration. The following documents should be uploaded into your application:

1. a statement of interest written in English. The applicants should write at most two pages focusing on: a) their reasons for applying; b) their areas of interest in relation to the various disciplines that contribute to the methods of language and communication technologies (Language Technology, Computational and Theoretical Linguistics, and Computer Science); c) their preferences for the host and partner Universities
2. a current Curriculum Vitae (CV), written in English
3. copies of: a) their Bachelor's degree b) certificates attesting to the fulfillment of the language requirements (except in the case of Toefl, which should come directly from the Toefl organization.)
4. a transcript of academic records
5. 2 Letters of Recommendation; it is preferable that the letters be sent sealed by the referees, or by email directly from the email account of the referee
6. a report on competence in other languages than English

In general, it is not necessary to follow with hard copies of the required documents. (exception: see point 3b, above).

The application deadline for admission in the LCT masters program is set for January 7th for both overseas applicants and non-overseas applicants. See the web site for information about the status overseas or non overseas.

Selection Process Students are selected by a joint committee including members from all partner institutions. Students are selected for study in the program on the basis of their performance and excellence in their previous studies.

The joint committee evaluate the application and establish a ranking (by February 15). After discussion, a shortlist of accepted candidates is devised. Following the procedure for allocating site described above, students are either granted their choices or suggested an alternative site. Students should confirm with the coordinator that they accept the proposal, and can then proceed to enrolment as students with the corresponding partner.

The *eligibility criteria* against which all applicants for an EM grant will be checked are: 1. Meeting of the application deadline; 2. Completeness of the application; 3. English language proficiency; 4. Bachelor degree.

The *selection criteria* against which all applicants will be assessed are: 1. Results in undergraduate education: mark and quality of first degree (which mark and from what university or programme); 2. Results in areas relevant for LCT; 3. Result of TOEFL, IELTS or an equivalent test: language abilities and practical application in studying or working; 4. Time spent to receive first degree and way of conducting studies (short and focused, if long additional activities, richness of studies depending on field, additional activities besides studies); 5. Having worked or studied in an interdisciplinary environment (type of first degree studies or additional experience outside the first degree studies); 6. Motivation for LCT-course (expectations, own contribution).

The selection criteria are applied as follows: 1. All criteria are equally weighted except the English proficiency criterion, which is applied in case of equal ranking otherwise; 2. We use the schema ++, +, 0, - encoding “very good”, “good”, “satisfactory” and “weak”, respectively; 3. Pre-assessment is done by a selection committee at UdS; final assessment is done by the joint commission; 4. Each application is assessed several times; 5. Diverging notations are discussed in the selection committee at UdS and the joint committee; 6. The absence of conflict of interest is guaranteed by making the assessors aware of this concern and by multiple assessment.

The assessment results are finalised by electronic discussion of the joint committee and conveyed to the students by March 15 of their application year.

Enrollment procedure 1. Fees are paid to the consortium; 2. Notification of final enrolment at each site is given at the beginning of May of each academic year; 3. Visas for overseas students are to be dealt with in the middle of May of each academic year.

5.2 Fees

Fees are set on a yearly basis. The annual fee will not exceed 8000 euro for non-European students and will not exceed 4000 euro for European students. Applicants can compete for waivers or scholarships, some of which are earmarked for particular countries or regions.

For further information please contact the programme coordinator.

6. University of Trento

Updated information will be published here soon.
Contact person: Nicu Sebe nicusebe@gmail.com

7 University of Lorraine

7.1 Language Technology Modules

7.1.1 Methodologies of Language Technology Module (LT-M1)

The **Methodologies of Language Technology** module at *University of Lorraine* is covered with the courses:

• Foundations in Linguistics **[6 ECTS of a total of 6 ECTS].**
• Logic and Statistic Tools for Language Modeling **[3 ECTS of a total of 6 ECTS].**
• Tools and Algorithms for NLP **[3 ECTS of a total of 6 ECTS].**
• Corpus Linguistics **[3 ECTS of a total of 3 ECTS].**
• Phonetics **[3 ECTS of a total of 3 ECTS].**

Statistical Methods. Course *Logic and Statistic Tools for Language Modeling*: Hidden Markov models, ngrams, perplexity, multi-grams. **[6 ECTS]**

Symbolic Methods. Course *Tools and Algorithms for NLP*: Finite state techniques. Feature Structures and Unification. Typed Lambda Calculus. Compositionality. **[6 ECTS]**

Cognition. Course *Logic and Statistic Tools for Language Modeling*.: Reasoning Models. Inference. **[6 ECTS]**

Corpus Linguistics. Course *Corpus Linguistics*: Techniques and resources for corpus construction. Annotation tools and schemes. Applications. **[3 ECTS]**

Text and Speech. Course *Phonetics*: Phonetics, phonology, prosody. **[3 ECTS]**

Foundations of Linguistics. Course *Foundations in Linguistics*: Basic concepts in Linguistics. **[6 ECTS]**

7.1.2 Computational Syntax and Morphology Module (LT-M2)

The **Computational Syntax and Morphology** module at *University of Lorraine* is covered with the courses:

• Logic and Statistic Tools for Language Modeling **[3 ECTS of a total of 6 ECTS].**
• Tools and Algorithms for NLP **[3 ECTS of a total of 6 ECTS].**
• Introduction to NLP – Morphology and Syntax **[3 ECTS of a total of 3 ECTS].**
• Grammatical Formalisms **[3 ECTS of a total of 3 ECTS].**

Finite State Techniques (FST). Course *Introduction to NLP – Morphology and Syntax* : Finite state automata. Tagging. **[3 ECTS]**

Probabilistic Approaches. Course *Tools and Algorithms for NLP*: Tabulation methods. Polarities and saturation of structures. Syntactic analysis as constraint resolution. **[6 ECTS]**
Course *Logic and Statistic Tools for Language Modeling*: Hidden Markov models, n-grams, perplexity, multigrams. **[6 ECTS]**

Formal Grammars Course *Grammatical Formalisms*: Head-driven Phrase Structure Grammars. Lexical Functional Grammars. Categorial Grammars. Tree Adjoining Grammars. Metagrammars. **[3 ECTS]**

Parsing Course *Grammatical Formalisms*: Parsing Algorithms. **[3 ECTS]**

7.1.3 Computational Semantics, Pragmatics and Discourse Module (LT-M3)

The **Computational Semantics, Pragmatics and Discourse** module at University of Lorraine is covered with the courses:
• NLP Applications **[3 ECTS of a total of 6 ECTS].**
• Introduction to NLP – Semantics and Pragmatics **[3 ECTS of a total of 3 ECTS].**
• Computational Semantics **[3 ECTS of a total of 3 ECTS].**
• Discourse and Dialogue Representation **[3 ECTS of a total of 3 ECTS].**

Syntax-Semantics Interface. Course *Introduction to NLP – Semantics and Pragmatics*: Semantic Representation. **[3 ECTS]**
Course *Computational Semantics*: Syntax-semantics interface. **[3 ECTS]**

Semantic Construction. Course *Introduction to NLP – Semantics and Pragmatics*: Lexical Semantics. **[3 ECTS]**
Course *Computational Semantics*: Montague Semantics. Compositionality. **[3 ECTS]**

Dialogue. Course *NLP Applications*: Dialogue systems. **[6 ECTS]**
Course *Discourse and Dialogue Representation*: Discourse and dialogue models. Anaphora and presupposition resolution. Treatment of the discourse and dialogue history. DRT. **[3 ECTS]**

7.1.4 LT Specialized Modules (LT-M4)

The **LT Specialized Modules** at *University of Lorraine* are covered with the courses:

• Linguistic Databases **[3 ECTS of a total of 6 ECTS].**
• NLP Applications **[3 ECTS of a total of 6 ECTS].**
• Generation **[3 ECTS of a total of 3 ECTS].**
• Mining of Textual Domains **[3 ECTS of a total of 3 ECTS].**
• Normalization and Linguistic Resources **[3 ECTS of a total of 3 ECTS].**
• Lexicology **[3 ECTS of a total of 3 ECTS].**

Information and Knowledge Representation. Course *NLP Applications*: Architectures and techniques for inference in NLP. **[6 ECTS]**

Information Retrieval (IR). Course *Mining of Textual Domains*: Symbolic Classification. Numeric Classification. Rule extraction. **[3 ECTS]**
Course *Normalization and Linguistic Resources*: Introduction to existing lexical, syntactic and semantic resources. The problem of information exchange and information structuring. Marking languages. Normalization. **[3 ECTS]**

Question Answering (QA). Course *NLP Applications*: Question answering. **[6 ECTS]**

Text and Speech Recognition and Generation. Course *Generation*: Basic architecture. Content planning. Micro-planning. Surface Realization. **[3 ECTS]**

Models of Human Language Processing and Understanding. Course *NLP Applications*: Speech Recognition. Language Understanding. Dialogue Systems. **[6 ECTS]**

Additions. Course *Lexicology*: Macro- and Micro-structure for dictionaries. Human and electronic dictionaries. Uses of dictionaries in NLP applications. **[3 ECTS]**
Course *Linguistic Databases*: Databases for linguistic applications. Corpus. **[6 ECTS]**

7.2 Computer Science Modules

7.2.1 Data Structures, Data Organization and Processing Module (CS-M1)

The **Data Structures, Data Organization and Processing** module at *University of Lorraine* is covered with the courses:

• Advanced Algorithms and OOP **[3 ECTS of a total of 6 ECTS].**
• Coding and Normalization of Information Repositories **[3 ECTS of a total of 6 ECTS].**
• Linguistic Databases **[3 ECTS of a total of 6 ECTS]**.

Algebraic Data Types. Course *Advanced Algorithms and OOP*: Basic and Advanced Data Types. Data specification. Object Oriented Programming. **[6 ECTS]**

Relational Databases. Course *Linguistic Databases*: Relational Databases. Normalization. Tools for access and exploration of corpus databases. **[6 ECTS]**

Semi-Structured Data and XML. Course *Coding and Normalization of Information Repositories*: Markup languages. Normalization. The problem of information exchange and information structuring. **[6 ECTS]**

7.2.2 Logic, Computability and Complexity Module (CS-M2)

The **Logic, Computability and Complexity** module at *University of Lorraine* is covered with the courses:

• Foundations in Mathematics **[6 ECTS of a total of 6 ECTS].**
• Knowledge Representation and Inference **[3 ECTS of a total of 6 ECTS].**

Logic and Inference. Course *Knowledge Representation and Inference*: Classical Logic. Proof Theory. Model Theory. Knowledge Representation. **[6 ECTS]**

Computability Theory. Course *Foundations in Mathematics*: Recursive functions. Decidability and undecidability. **[6 ECTS]**

Complexity Theory. Course *Knowledge Representation and Inference*: Complexity Classes in terms of space and time. **[6 ECTS]**

7.2.3 Formal Languages and Algorithms Module (CS-M3)

The **Formal Languages and Algorithms** module at University of Lorraine is covered with the courses:

• Advanced Algorithms and OOP **[3 ECTS of a total of 6 ECTS].**
• Scripting Languages for NLP **[6 ECTS of a total of 6 ECTS].**

Formal Grammars and Languages Hierarchy. Course *Scripting Languages for NLP*: Regular Expressions. Context Free Languages. **[6 ECTS]**

Parsing and Compiler Design. Course *Scripting Languages for NLP*: Parsing algorithms. **[6 ECTS]**

Search Techniques and Constraint Resolution. Course *Advanced Algorithms and OOP*: Search and sorting algorithms. **[6 ECTS]**

7.2.4 CS Specialized Modules (CS-M4)

The **CS Specialized Modules** at *University of Lorraine* are covered with the courses:

• Coding and Normalization of Information Repositories **[3 ECTS of a total of 6 ECTS].**
• Knowledge Representation and Inference **[3 ECTS of a total of 6 ECTS].**

Automated Reasoning. Course *Knowledge Representation and Inference*: Automated Theorem Proving. Reasoning methods (tableaux, resolution, etc.). Testing. **[6 ECTS]**

Additions. Course *Coding and Normalization of Information Repositories*: Advanced topics in Normalization. Efficient access to large repositories. XML theory and practice. **[6 ECTS]**

7.3 Additional Information

Course	ECTS	Obligatory?	LT M1	LT M2	LT M3	LT M4
Foundations in Linguistics	6	Yes	6			
Linguistic Databases[1]	6	Yes				3
Logics and Statistics Tools for Language Modeling	6	Yes	3	3		
NLP Applications	6	Yes			3	3
Tools and Algorithms for NLP	6	Yes	3	3		
Introduction to NLP - Morphology and Syntax	3	Yes		3		
Introduction to NLP - Semantics and Pragmatics	3	Yes			3	
Grammatical Formalisms	3	No		3		
Computational Semantics	3	No			3	
Discourse and Dialogue Representation	3	No			3	
Corpus Linguistics	3	No	3			
Generation	3	No				3
Phonetics	3	No	3			
Mining of Textual Domains	3	No				3
Normalization and Linguistic Resources	3	No				3
Lexicology	3	No				3
Total			18	12	12	15
Required			6	9	9	6

Course	ECTS	Obligatory?	CS M1	CS M2	CS M3	CS M4
Foundations in Mathematics	6	Yes	6			
Advanced Algorithms and OOP	6	Yes		3	3	
Coding and Normalization of Information Repositories	6	Yes			3	3
Knowledge Representation and Inference	6	Yes	3			3
Scripting Languages for NLP	6	No		6		
Linguistic Databases[2]	6	No			3	
Total			9	9	9	6
Required			9	9	6	4

[1]Other 3 ECTS in CS M3
[2]Other 3 ECTS in LT M3

For more information see http://tal.loria.fr.

Contact person: Sylvain Pogodalla
E-mail: Sylvain.Pogodalla@inria.fr

8 Charles University in Prague

8.1 Language Technology modules

8.1.1 Methodologies of Language Technology Module (LT-M1)

The **Methodologies of Language Technology** module at the Charles University in Prague is covered with courses on:

Statistical Methods. Course *Statistical Methods in Natural Language Processing I*: Introduction to formal linguistics and the fundamentals of statistical natural language processing, including basics of Information Theory, Language Modelling and Markov Models. **[3/6 ECTS, obligatory]**

Corpus Linguistics. Course *Language Data Resources*: The goal of the seminar is to provide students with the survey of the field of Language Resources. Selected types of linguistic annotations will be described, with emphasis on annotating textual data (morphological categories, constituency and dependency syntactic trees, anaphora, discourse structure, word-sense disambiguation, parallel-text alignment etc.) and lexical data (wordnets, translation dictionaries, valency lexicons etc.). Leading projects for English, Czech, and some other languages will be used for illustration. **[5 ECTS]**

Text and Speech. Course *Fundamentals of Speech Recognition*: This seminar deals with speech recognition tasks and feature extraction of voice and utterance characteristics. Of particular interest will be topics related to Hidden Markov Models as applied to speech (FFT, n-dimensional clustering, Gaussian mixtures, parameter value extraction from data, phonetic representation, prosodic analysis etc.). Preparation and training of own speech recognition models. **[3 ECTS]**

Course *Text-to-Speech synthesis*: Speech and its automatic synthesis; writing systems, text analysis and processing; prosody modelling; selected TTS-related language-specific phenomena; approaches and algorithms used in TTS. No prerequisite knowledge assumed. **[3 ECTS]**

Foundations of Linguistics. Course *Introduction to General Linguistics*: The course will help students to get familiar with fundamental features of human language, its structure and development, its place in the society, etc. Most topics will be addressed both from the perspective of traditional linguistics and from the formal perspective of mathematics and computer science. **[5 ECTS, obligatory]**

Course *Introduction to Formal Linguistics*: A survey and critical discussion of the main trends of developments in theoretical and formal linguistics from N. Chomsky to the most recent approaches. **[3 ECTS]**

8.1.2 Computational Syntax and Morphology Module (LT-M2)

The **Computational Syntax and Morphology** module at the Charles University is covered with courses on:

Computational Linguistics. Course *Introduction to Computer Linguistics*: The main goal of this course is to provide an overview of individual subfields of computational linguistics. Main issues being solved by these subfields are also mentioned. Among the subfields the course stresses are machine translation, syntactic parsing, morphology and corpus linguistics. **[3 ECTS]**

Probabilistic Approaches. Course *Statistical Methods in Natural Language Processing I*: Introduction to formal linguistics and the fundamentals of statistical natural language processing, including basics of Information Theory, Language Modeling and Markov Models. **[3/6 ECTS obligatory**]

Formal Grammars. Course *Linguistic Theory and Grammar Formalisms*: The aim of this course is to bridge the gap between theoretically motivated description of linguistic phenomena and a corresponding implementation in formal grammars. Following an overview of formal frameworks coupled with specific theories – Categorial Grammar (CG), Tree Adjoining Grammar (TAG), Lexical Functional Grammar (LFG), Head-driven Phrase Structure Grammar (HPSG) - and formal aspects of other theoretical frameworks (Chomskyan and dependency-based tradition), the students will be introduced to the core principles of HPSG both as a theory and as a formalism, based on examples of relevant phenomena in English, Czech and other languages. In parallel with the classroom presentations and discussions the students will develop corresponding grammars of increasing complexity, using the system Trale as the grammar writing environment. **[5 ECTS]**

Tagging. Chunking. Parsing. Course *Morphological and syntactic analysis*: Basic methods and algorithms used for morphemic segmentation, morphological and syntactic (constituency-based, dependency-based, tectogrammatical) analysis of natural languages. We will try out some of the approaches on an unknown language, as student mini-projects during the semester. Credits will be awarded for contribution to these mini-projects. **[3 ECTS]**

Course *NLP Applications*: The main goal of the course is to introduce basic types of natural language processing (NLP) applications and to give the students a chance to work with some of those applications in seminars. The course will concern machine translation, machine aided human translation tools, localization tools, information retrieval and extraction, question answering, speech recognition, spelling and grammar checking, generation etc. **[3/5 ECTS]**

8.1.3 Computational Semantics, Pragmatics and Discourse Module (LT-M3)

The **Computational Semantics, Pragmatics and Discourse** module at the Charles University is covered with courses on:

Syntax-Semantics Interface. Formal Semantics. Course *Prague Dependency Treebank*: The subject should make the students familiar with Prague Dependency Treebank (PDT 2.0) project, starting from its theoretical base, including particular layers of annotation and ending with the way how important linguistic phenomena are represented. Emphasis is also placed on annotation schemata and data format, on familiarization with useful tools and practical work with the treebank. **[2/5 ECTS]**

Course *Lexical Analysis of Natural Language*: Introduction to computational aspects of lexical semantics. Basic concepts and issues. Fundamental approaches to lexical disambiguation. **[3 ECTS]**

Discourse. Dialogue. Course *Information Structure of Sentence and Discourse Structure*: The information structure of a sentence (or according to the Czech traditional terminology, the topic-focus articulation; a division of a sentence into the part the sentence is about and the focus part) is an important starting point for an analysis of higher units than a sentence, namely discourse (text) and its structure. The course will deal with semantic consequences of the topic-focus articulation, with its treatment in the formal description of a language and with language-dependent means of the expression of the topic-focus articulation. Special attention will be paid to the way in which this aspect of the sentential structure is treated in the electronic Prague Dependency Treebank and to the question how the corpus can be used for a verification of linguistic hypotheses. In the second part of the course, the questions of the discourse structure will be discussed, especially how the observations of the sentential structure can be used in the study of different aspects of discourse. In this part, we will use the material from Prague Dependency Treebank as well, namely the annotation of coreferential and anaphoric relations. **[3 ECTS]**

Course *NLP Applications*: The main goal of the course is to introduce basic types of natural language processing (NLP) applications and to give the students a chance to work with some of those applications in seminars. The course will concern machine translation, machine aided human translation tools, localization tools, information retrieval and extraction, question answering, speech recognition, spelling and grammar checking, generation etc. **[2/5 ECTS]**

8.1.4 LT Specialized Modules (LT-M4)

The **LT Specialized Modules** at the Charles University are covered with courses on:

Machine Translation (MT). Course *Statistical Methods in Natural Language Processing II*: Continuation of Statistical Methods in Natural Language Processing I. Introduces the notion of linguistic experiment and its evaluation. The role of corpora in statistical NLP. Standard NLP tasks (tagging, parsing) are explained and methods presented. Short introduction to Statistical Machine Translation. **[6 ECTS]**

Course *Tools of Automated Translation*: The course concerns the history of machine translation and its contemporary trends. The historical part will introduce the most famous translation systems (TAUM-METEO, Systran, Eurotra, ETAP), with a special attention devoted to the systems involving Czech (Ruslan, Česílko). The students will learn about individual methods used in MT, especially with the traditional rule-based approach, with the stochastic MT, Example-based MT and Knowledge-based MT. A part of the course will be devoted to the computer-aided translation systems and tools. The course also includes practical sessions with development tools and CAT tools. **[3 ECTS]**

Course *Statistical Machine Translation*: Participants of the seminar will get closely acquainted with methods of machine translation (MT) that rely on automatic processing of (large) training data as well as with open-source implementations of these methods. We will cover a range of approaches starting with linguistically uninformed "phrase-based" MT up to surface and deep syntactic MT. The final grade will reflect mainly students' own contributions: either experimental results, implemented tools or modifications to existing systems, or survey reports. **[3 ECTS]**

Information and Knowledge Representation. Course *Prague Dependency Treebank*: The subject should make the students familiar with Prague Dependency Treebank (PDT 2.0) project, starting from its theoretical base, including particular layers of annotation and ending with the way how important linguistic phenomena are represented. Emphasis is also placed on annotation schemata and data format, on familiarization with useful tools and practical work with the treebank. **[3/5 ECTS]**

Speech Recognition and Generation. Course *Algorithms in Speech Recognition*: The course presents recent methodologies and software toolkits for speech recognition. Students will learn how to develop systems of automatic speech recognition and transcription, computer dialogue systems and speaker identification. The course shows principles, preparation and decoding algorithms of statistical acoustic and language models (HMM, n-gram and structured language models, final state transducers, graphical models, Viterbi dynamic programming, heuristic hypothesis search strategies, stack decoder). **[6 ECTS]**

8.2 Computer Science Modules

8.2.1 Data Structures, Data Organisation and Processing Module (CS-M1)

The **Data Structures, Data Organisation and Processin**g module at the Charles University in Prague is covered with courses on:

Semi-Structured Data and XML. Course *NLP Technology*: The aim of the course is to get students familiar with basic software tools used in natural language processing (bash command line and scripting, text-processing commands; introduction to Perl; object-oriented inteface for processing linguistic structures; representing linguistic structures in XML; tools for processing morphologically and syntactically tagged data, visualization, search; distributed processing of linguistic tasks). **[4 ECTS, obligatory]**

Information retrieval. Course *Information Retrieval Systems*: String matching algorithms. Searching and data retrieval from text databases. Architecture of text retrieval systems. Text compression. Corrections of texts in a natural language. **[3 ECTS]**

8.2.2 Logic, Computability and Complexity Module (CS-M2)

The **Logic, Computability and Complexity** module at the Charles University in Prague is covered with courses on:

Computability Theory. Complexity Theory. Course *Introduction to Complexity and Computability Theory*: This is a basic course on the computability theory and computational complexity. Roughly the first

half of the course is devoted to the introduction to computability theory: Turing machines. Computable functions. Recursive and recursively enumerable sets. Undecidable problems. Recursion theorem. The second half of the course is devoted to the study of space and time complexity classes: Equivalence of PSPACE and NPSPACE, Class NP. Polynomial reducibility among problems. Proofs of NP-completeness. Approximation algorithms and approximation schemes. **[5 ECTS, obligatory]**

8.2.3 Formal Languages and Algorithms Module (CS-M3)

The **Formal Languages and Algorithms** module at the Charles University in Prague is covered with courses on:

Search Techniques and Constraint Resolution. Course *Constraint programming*: The course gives a survey of constraint satisfaction techniques. The focus is on algorithms for constraint satisfaction, such as search algorithms (depth-first search and local search) and propagation algorithms (arc and path consistency). Solving over-constrained problems is also discussed as well as some modeling techniques are covered. **[3 ECTS]**

Course *Planning and Scheduling*: The course gives an introduction to planning and scheduling. It is focused on the algorithms for solving planning and scheduling problems with emphasis on using constraint-based techniques. **[3 ECTS]**

Automated learning. Course *Introduction to Machine Learning (in Computational Linguistics)*: This one-semester introductory course provides theoretical background of and key algorithms from the field of machine learning (ML) explained independently on a broad spectrum of multidisciplinary applications the ML takes place in. The seminars are application-dependent and they accompany the lecture sessions. The aim of the seminars is an acquisition of practical experience from application of ML approaches on problems from natural language processing. **[3/6 ECTS]**

Parsing and compiler design. Course *Compiler Principles*: Introductory compiler course concentrates primarily on theoretical and practical principles of fore-end compiler construction. Exercises emphasize elementary using of tools for compiler construction. A student will be capable to construct his/her own compiler into an intermediate code or an another language after finishing this course. **[5 ECTS]**

8.2.4 CS Specialized Modules (CS-M4)

The **CS Specialized Modules** at the Charles University in Prague are covered with courses on:

Artificial Intelligence (AI). Course *Artificial Intelligence I*: An introductory course on artificial intelligence with the focus on basic concepts and methods. Intelligent agents, environment, and structure of agents. Problem solving by search (DFS, BFS, ID, A*, IDA*, local and on-line search, heuristics). Constraint satisfaction. Games (minimax, alfa-beta pruning). **[3 ECTS]**

Data mining. Course *Knowledge mining*: A rapid development in the area of data mining is motivated by the necessity to "translate" huge amounts of processed and stored data into meaningful information. This lecture is focused on understanding principal concepts and techniques applicable to data mining. Basic principles of their application to novel solutions of practical tasks will be discussed in detail. These comprise mainly business and Web applications, but others as well. **[9 ECTS]**

Semantic Web. Course *Web Semantization*: An introduction to Semantic web - methods and models supporting applications that need to process and share the content of information on the web. We cover both practical aspects (e.g. crawling and annotation of web resources, user models, ontology mapping, web services and querying) and theoretical aspects (RDF structures and satisfiability, description logic (computational complexity, effectiveness versus expressiveness, OWL like DL, DL with concrete domains, connections to rule based systems)). In labs we first introduce basic techniques, each student elaborates a certain problem. [**4 ECTS]**

Neural networks. Course *Neural Networks*: The theory of neural networks is motivated by the results achieved in the area of the central neural system research. These inventions often represent the origin for the derived mathematical models which have (despite of significant simplifications of real neuro-physiological processes) some features of the natural intelligence. These models can be used in the design of non-traditional computational means applied in the solutions of many practical problems. **[9 ECTS]**

Machine Learning. Course *Introduction to Machine Learning (in Computational Linguistics)*: This one-semester introductory course provides theoretical background of and key algorithms from the field of machine learning (ML) explained independently on a broad spectrum of multidisciplinary applications the ML takes place in. The seminars are application-dependent and they accompany the lecture sessions. The aim of the seminars is an acquisition of practical experience from application of ML approaches on problems from natural language processing. **[3/6 ECTS]**

Others. Course *Modern Methods in Computational Linguistics*: The advanced course focusing on presentation and discussion of important scientific papers from the area of contemporary computational linguistics, machine learning and related areas. During the semester, each participant is supposed to give talks about selected papers. **[3 ECTS]**

Course *String Algorithms*: A survey of algorithms and data structures for efficient computation of patterns in strings with applications. Topics covered include basic properties of strings, data structures for processing strings, string decomposition, exact and approximate string matching algorithms, string distance calculations, repetitions and covers searching and applications (bioinformatics, data compression, speech recognition, computer vision). **[3 ECTS]**

Course *Pattern Recognition*: This lecture is focused on principal concepts of pattern recognition. The content of the lecture is the description and analysis of various methods applicable to pattern recognition. **[3 ECTS]**

8.3 Additional Requirements

Software project. The goal of this course is to practice a team work in a group software project. The work on the project is finished by a public presentation. The projects topics may come from various fields, therefore the credits are divided among the corresponding modules with regard to a concrete project topic. **[15 ECTS, obligatory]**

Additions. Course *Czech for Foreigners* **[3 ECTS]**

Statistical Methods in Natural Language Processing I	6 ECTS
Introduction to General Linguistics	5 ECTS
NLP Technology	4 ECTS
Software Project	15 ECTS
Introduction to Complexity and Computability	5 ECTS

Obligatory courses at the Charles University in Prague

For more info see: http://ufal.mff.cuni.cz/lct/modules.html

Contact persons: RNDr. Vladislav Kuboň, Ph.D., RNDr. Markéta Lopatková, Ph.D.

E-mail: Vladislav.Kubon@mff.cuni.cz, Marketa.Lopatkova@mff.cuni.cz

9 University of Groningen

9.1 Language Technology Modules

9.1.1 Methodologies of Language Technology Module (LT-M1)

Cognition. Course *Computer-Mediated Communication 1*: Human-Computer Interaction: In this course students study some of the problems dealt with in the area known as human-computer interaction (HCI). HCI endeavors to provide an understanding of both the human user and the computer system he interacts with in order to accomplish his tasks. The course concentrates on how users communicate with interactive systems in the performance of various tasks. It also looks at the possibilities that the computer system can offer to optimize communication. The following topics (and subtopics) get special attention: 1) HCI as an interdisciplinary field (various perspectives and theories); 2) User interfaces (interaction styles, design issues and usability); 3) Groupware (user environments and forms of communication); 4) Visual communication (diagrams and diagrammatic reasoning, and information visualizations). **[10 ECTS]**

Course *Computer-Mediated Communication 2*: Information Systems: Topic of this course are information systems and their applications within computer-mediated communication. We will review a number of these systems both from a more theoretical perspective and from a practical angle, where we will look at and work with concrete implementations. The range of systems to be discussed encompasses (but is not necessarily restricted to) intranets, communication platforms (e.g., Blackboard), databases and their user interfaces, MT systems, and the Semantic Web. usability of such systems. **[10 ECTS]**

Course *Semantics and Language Acquisition*: In this module, students will examine an issue raised in recent literature and learn to conduct research into the acquisition of semantic patterns and the lexical semantics of Dutch words and sentences. [**10 ECTS]**

Course *Language Modelling*: Each year in this course a topic of current interest in linguistics is chosen to be studied in depth. In particular, topics or problem areas where linguistic information and non-linguistic information both play a role in the production and interpretation have been choisen, and earlier topics include focus and conversational implicatures. The course begins with a teacher-led introduction to the topic, after which students will present original and recent research on the topic for critical discussion. Emphasis will be placed on comparing the different theoretical approaches, and evaluating the results of empirical and experimental work bearing on the topic. Additionally, the degree to which the research results are relevant and usable in natural language applications will be discussed. In the final phase of the course, students will set up and implement their own pilot experiment, computer model or empirical study, independently or in groups. Scientific method and critical thinking will be emphasized. In past years these studies have sometimes been of such high quality that they have been turned into workshop presentations and papers. **[5 ECTS]**

Foundations of Linguistics. Course *Semantics*: This research module teaches students to conduct their own research in the field of semantics. Attention is paid to critically reviewing the research literature, operationalization of the research question in the light of the results of the literature search, and planning and conducting the research. During the course, students must regularly report on the progress of their research in oral presentations and written reports. Students are required to give a presentation. The module must be concluded with a research paper that could form the basis for the MA thesis. **[10 ECTS]**

Course *Theoretical Linguistics* This course gives an introduction on three main topics of theoretical linguistics. The courses on Syntax will introduce the minimalist developments in the generative theory. This introduction covers the major aspects of generative syntactic theory, dealing with topics such as phrase structure, argument structure, case and agreement, binding and control, and movement of heads and phrases. The courses on Phonology offer a global overview of different phonological issues and theories, with an emphasis on data analysis and phonolocical methodology. The courses on Semantics will cover the relation between language, denotations and mental representations, a theory-neutral introduction to basic lexical semantics, sentence semantics, contextual semantics, overview of some standard semantics schools. **[5 ECTS]**

Corpus. Course *Machine Learning I* : This advanced course aims to enable students to use and understand machine learning techniques, with a particular focus on applications in linguistic analysis. We examine the foundations of learning theory, such as bias, entropy, supervised vs. unsupervised techniques, training vs. testing, baselines, evaluation especially cross-validation and the kappa coefficient, transparent vs. non-transparent techniques, maximum likelihood techniques such as estimation maximization, and incrementality. We present and experiment with rule induction, decision trees, memory-based (aka instance-based or "lazy" learning), (naive) Bayesian techniques, and then one or two advanced techniques such as inductive logic programming and/or support-vector machines. **[5 ECTS]**

9.1.2 Computational Syntax and Morphology Module (LT-M2)

Finite state techniques and probabilistic approaches. Course *Natural Language Processing*: This master course constitutes a solid introduction to the field of natural language processing. The course consists of two parts. In the first part (10 weeks) the course treats a number of foundational topics in NLP (regular expressions, finite automata, computational morphology, computational phonology, N-grams, HMMs, Part-of-speech Tagging, Context-free Grammar, Parsing, Unification grammar, Chomsky hierarchy, computational semantics). For this part of the course, there is a final exam. In the second part (4-6 weeks) the course focuses on a topic in particular for which a number of recent research papers are collected and studied. For this part of the course, the students are expected to prepare a presentation on one such paper. The topics of previous years included 'disambiguation and parse selection', 'distributional similarity', and 'automated summarization'. **[10 ECTS]**

9.1.3 Computational Semantics, Pragmatics and Discourse Module (LT-M3)

Dialogue. Course *Discourse Analysis*: The topic of this course is discourse (language use in context). We will set out from basic notions of textual coherence (e.g., cohesion or temporal structure) and textual genres (including a discussion of the orality/literacy distinction). On the basis of these notions, we will discuss approaches to discourse analysis, i.a., Rhetorical Structure Theory and Appraisal Analysis. We will investigate the way in which the functions of a text are based on the way it is organised. All theoretical notions and concepts will be applied to concrete specimens of discourse in homework assignments and a term paper. **[10 ECTS]**

9.1.4 LT Specialized Modules (LT-M4)

Models of Human Language Processing and Understanding. Course *Cognitive Models of Central Cognition*: In this course, the following topics are discussed: general introduction to cognitive modeling as a research technique; cognitive modeling of development; skill acquisition; executive control; time perception; cognitive modeling and neuro-imaging. **[5 ECTS]**

Course *Cognitive models of perception*: This course is structured as follows: general introduction in perception; psychophysics: the senses and the signals they process; hierarchies of processing in perception: how to reach recognition results; modeling perception: sound and vision; and linking the natural system with artificial systems. **[5 ECTS]**

Inforation and Knowledge Representation. Course *Advanced Web Technology*: This module offers an introduction to the Semantic Web. The first part of the course covers the concepts, languages, and tools which are important for developing the semantic web, i.e. XML, XPath, XSLT, XQuery, interoperability, ontologies and reasoning, and RDF(S) and OWL. In the second part of the course, students work on a project in the area of the hunanities, which makes use of semantic web technology. A basic knowledge of HTML and XML, and modest programming skills, are assumed. **[10 ECTS]**

Course *Machine Learning II* : This course focuses on work done to simulate one or another aspect of language acquisition. The focus will be on phonological learning, esp. phonotactics and allomorphy in morphophonolgy. Students will learn the issues involved in simulation, the idealizations and controls needed to test hypotheses, the sorts of implementations that have been realized, and the sorts of hypotheses that have been successfully tested. The seminar will survey work on computational models of various aspects of language learning, including work by Cartwright and Brent (Cognition 1996, also in Brent's book) on word segmentation; John Goldsmith (CL 2001) on morphology; Nerbonne, Tjong Kim

Sang, Stoianov and Konstantopoulos on phonotactics (various places); and Albright and Hayes (Cognition 2003) on learning phonology. We will relate these lines of work to linguistic and psychological issues such as the dispute concerning the poverty of the stimulus argument (Pullum and Scholz), connectionists' views (Elman et al. "Rethinking Innateness"), and work on children's sensitivity to distributional tendencies (work by Newport, Mintz, and others). **[5 ECTS]**

Course *Methodology and Statistics*: This seminar revolves around presentations by participants. Presentations primarily concern statistical or methodological issues in the research of the participants, both students involved in the Research Master's in Linguistics and gradudate students conducting PhD. projects. See www.let.rug.nl/ nerbonne/teach/rema-stats-meth-seminar/ for some of the topics presented and discussed in 2004 and 2005. The content varies depending on the participants research needs, but the 2005 topics were: association statistics for recognizing multi-word units, magnitude estimation, repeated measures ANOVA, Fisher's exact test, corpora and counting, search in automatically analysed corpora, binomial models, log odds ratios, logistic regression, multi-level (hierarchical) regression, clustering, entropy as a measure of syntactic Influence, and kappa as a measure of quality in indentifying technical terminology automatically. The schedule for 2006 is incomplete, but will likely include permutation tests, bootstrapping, multinomial models, ANCOVA, factor analysis, latent semantic analysis, and various analytical techniques for categorical data (again). **[ECTS points: 10]**

Additions. Course *Handwriting Recognition*: The goal of this course is to provide insight into the problems which are encountered in the automatic recognition of script (characters, words), at the level of sensor data (pixels, vectors), at the level of pattern classification and at the level of natural language and the context in which the recognition system ('Reading System') is used. The following processing stages are treated: preprocessing, normalisation, segmentation, classification and post processing. Practical assignments are done, such that the student obtains experience in the design and computation of geometrical features. Usually, a common benchmark data set for training and testing classifiers is used. **[5 ECTS]**

Course *Sound Recognition*: This advanced course is the world's first course in (nonspeech) sound recognition and is aimed at the design of recognition systems that function reliably with unconstrained input using Continuity Preserving Signal Processing. The course has a theoretical and a practical part. The theoretical part starts with the design constraints of recognition systems that must function autonomously with unknown, variable, uncontrollable, and complex input. The second part of the course uses this approach to implement and test a concrete system, e.g., a sound source detector or a phoneme recogniser, by the students (Matlab). **[5 ECTS]**

9.2 Additional Information

For more information see http://www.rug.nl/let/onderzoek/onderzoekinstituten/clcg.

Contact person: Dr. Gosse Bouma
.

E-mail: g.bouma@rug.nl

10 Saarland University (UdS)

10.1 Language Technology Modules

10.1.1 Methodologies of Language Technology Module (LT-M1)

The **Methodologies of Language Technology** module at UdS is covered with courses on:

Symbolic Methods. Course *Syntactic Theory*: Characteristic properties of different grammar models, Phrase Structure Grammars, Lexical Functional Grammar (LFG), Head-Driven Phrase Structure Grammar (HPSG), Categorial Grammar. **[6 ECTS]**

Course *Semantic Theory*: This course teaches the prevalent formalisms and methods in natural-language semantics and their applications in natural language understanding systems. The students acquire the background knowledge necessary for an understanding of the current literature, and are acquainted with phenomena and methods in the semantics of words, sentences and texts, together with their formal modelling, as well as with the modelling of the syntax semantics interface and the interface to logic-based inference systems. Topics of this course include: (a)Sentence semantics: Montague-style type-theoretical representations; semantics construction; scope ambiguities; underspecification. (b) Discourse semantics: Anaphora; discourse representation theory (DRT). (c) Lexical semantics: Events; thematic roles; frames; lexical resources. **[6 ECTS]**

Cognition. Course *Computational Psycholinguistics*: Computational psycholinguistics is concerned with the development of computational models of how people process natural language. Models are guided by the desire to explain the cognitive architectures and mechanisms that underlie language comprehension, as revealed by evidence from on-line psycholinguistic experiments. This course will begin with an introduction to the aims and central issues in psycholinguistic research, summarising the key empirical observations about human language understanding before considering a range of models which have been proposed. In examining a wide range of computational models – symbolic, probabilistic, connectionist, and also 'hybrid' computational mechanisms – we consider both how well models explain human behaviour, as well as what they contribute to central theoretical debates concerning the nature of language acquisition and linguistic performance. **[6 ECTS]**

Text and Speech. Course *Language Technology I* : Information Management: Classification, IR, Summarisation, IE, Named-entity recognition, Language checking, Question Answering. **[6 ECTS]**

Course *Language Technology II* : The goal is to teach concepts and common technologies in the application areas of multilingual information management, machine translation and natural language interaction. Students learn exemplary approaches to building and evaluating such applications and acquire practical experience through designing, implementing and evaluating small systems. Topics covered in the course include concepts in machine translation, multilingual resources, multilingual language technology, computational dialogue modeling and dialogue management, issues in spoken and multimodal dialogue system development and natural language-based computer-assisted learning, user modeling, usability testing and evaluation. **[6 ECTS]**

Course *Speech Science*: This course gives an overview of and provides accompanying practical exercises in the three main areas of Speech Science: speech production, the acoustic structure of speech and speech-sound perception. The topics covered include the anatomy and physiology of speech-sound production; articulation and the symbolic representation of speech; theories of speech-production control; the source-filter model of speech production and the acoustic properties of sound classes; the ear and hearing physiology, theories of speech-sound and speech perception. **[6 ECTS]**

Course *Speech Technology*: Systems driven by speech technology consist of various components. On of the furthest developed is speech recognition which at its heart is a pattern recognition problem. In this course, we will cover the basic principles of pattern recognition and machine learning and see how they are applied to speech recognition. Specific topics are: classifiers, supervised Learning, unsupervised

learning, Hidden Markov Models, Gaussian Mixture Models and Acoustic Modeling, classification and Regression Trees, adaptation, search. **[6 ECTS]**

Foundations of Linguistics : Course *Foundations of Language Science and Technology*: Map of Language Science and Technology, why is language/speech difficult and interesting, ambiguity, commication. inference, linguistic phenomena, levels, concepts, phonetics, prosody, morpholoy, syntax ... pragmatics, automata, morphology, CFGs, parsing, corpora and data-intensive linguistics, human parsing: memory limitation and attachment, differences between human and machine processing, logic, ontologies, wordnet, HMMs: grapheme-phoneme-convert, machine learning, spoken dialog systems. **[6 ECTS]**

Course *Phonological Theory*: This course provides an overview of and applicational practice in the main phonological theories of the 20th century. Apart from offering knowledge of these theories, one of the main goals is to place present-day approaches into a wider theoretical perspective and avoid a blinkered acceptance of any one school of thought. Topics covered will include Structuralist schools and theories of the Phoneme; feature specification and generative rules in linear generative phonology; the cyclical stress rule; the shift to nonlinear phonology and the autosegmental-metrical approach with feature geometry and hierarchical segmental and prosodic representations; phonology and variation - from linear generative rules to Optimality Theory. **[6 ECTS]**

10.1.2 Computational Syntax and Morphology Module (LT-M2)

The **Computational Syntax and Morphology** module at UdS is covered with courses on:

Probabilistic Approaches. Course *Data-Oriented Parsing and Generation*: Data-Oriented Parsing (DOP) models embody the assumption that humans produce and interpret natural language utterances by invoking representations of their concrete past language experience, rather than the rules of a consistent and non-redundance competence grammar. DOP models therefore maintain large corpora of sentences with syntactic structures. They analyze new input-sentences by combining partial structures from the corpus, and employ the occurrence frequencies of these structures to estimate which of the resulting analysis are the most probable one. During this seminar we will have a closer look to the computational and linguistic aspects of DOP. Recently, first Data-Oriented methods for natural language generation have been proposed, which we will discuss at the end of the seminar. **[9 ECTS]**

Formal Grammars. See Course *Syntactic Theory*.

Parsing. Course *Computational Linguistics*: The goal of the course is to introduce the students to various standard algorithms in computational linguistics. The focus is on the structure of the algorithms, i.e. their data structures and mechanisms. The course discusses standard algorithms used for various types of linguistic processing in computational linguistics. The algorithms discussed in the course range from shallow methods such as pattern matching algorithms for strings and trees, and finite state methods; to machine learning and statistical techniques such as Hidden Markov Models and decision trees; to various algorithms used in deep linguistic processing. Examples of the latter are memorization techniques, unification, graph algorithms, and inferencing with ontologies. The algorithms are illustrated with practical applications from computational linguistics. The students will gain hands-on experience with the algorithms either through using existing implementations or by having to implement provided exercises. **[9 ECTS]**

10.1.3 Computational Semantics, Pragmatics and Discourse Module (LT-M3)

The **Computational Semantics, Pragmatics and Discourse** module at UdS is covered with courses on:

Syntax-Semantics Interface. Course *Morphosyntax-Semantics Interface in Lexicalist Theories*: In recent years, there has been an increasing interest in the interface between (morpho-)syntax and word meaning. One of the main reasons for that is that generalizations over word classes have been proven to help linguistic theories - especially the ones developed in the generative tradition - overcome the natural limitations of syntax. In lexicalist theories like Lexical-Functional Grammar (LFG) and Head-Driven Phrase Structure Grammar (HPSG), where the structure of the lexical knowledge plays a central role in the

theory, the interest in the interaction between (morpho-)syntax and word meaning has led to the development of linking models like the Lexical Mapping Theory (LMT), the "Optimal Linking" Theory of Butt, Dalrymple and Frank (1997) and the Hierarchical Lexicon models. But although these linking models have as a common starting point the recognition of the importance of word classes for the interface between semantics and syntax, they vary both ontologically and in the range of linguistic phenomena they attempt to explain. The aim of this course is to present and explore approaches on formal, empirical and computational issues related to the Morphosyntax-Semantics Interface in lexicalist theories (mainly LFG and HPSG). It also intends to address inter-framework discussions, since it focuses on both LFG and HPSG. **[9 ECTS]**

Semantic Construction. Course *Semantic Theory*.

Dialogue. Course *Modeling Grounding Subdialogues in an ISU-Based System*: The goal is to gain theoretical knowledge and practical experience in developing dialog systems in the Information-State Update (ISU) framework, with focus on various grounding strategies. Grounding subdialogs, e.g., various degrees of explicit/implicit feedback, verification, clarification or correction, are needed in any dialog system. We will first introduce ISUbased dialog modelling. Then the students will present papers addressing various grounding issues and (optionally) implement ISU-based dialog models. The seminar is related to two research projects at CoLi, DIALOG (tutorial dialogue in mathematics) and TALK (generic methods for flexible, adaptive, multimodal dialogue, with in-car and in-house application scenarios). **[9 ECTS]**

Formal Semantics. See Course *Semantic Theory*.

10.1.4 LT Specialized Modules (LT-M4)

The **LT Specialized Modules** at UdS are covered with courses on:

Machine Translation (MT). Course *Human and Machine Translation*: Examination of the process of translation as done by professional and amateur translators, and existing and proposed machine-translation systems, illuminating what each might learn from the others. **[3 ECTS]**

Course *Empirical Approaches to Machine Translation*: This course will focus on the empirical evaluation of recent developments in statistical machine translation (SMT). After the presentation of some essential mathematical background, participants will learn to use open-source software and parallel corpora to build statistical, phrase-based translation models for language pairs of their choice. Available languages include the 11 "old" official EU languages, Chinese, Arabic, Japanese, Inuktitut, and many more languages used in the OPUS project. The quality of the resulting translation systems will be evaluated both by automatic comparison (BLEU score) to reference translations and by inspection and linguistic classification of the errors they make. If applicable, comparative evaluations with commercial or web-based MT systems will also be conducted. [**3 ECTS]**

Information and Knowledge Representation. Course *Language Technology for the Semantic Web*: Knowledge Markup and Ontology Learning: Establishing the SemanticWeb on a large scale implies widespread automatic annotation of web documents with ontology-based semantic metadata (knowledge markup) and dynamic adaptation of corresponding ontologies to evolving applications and domains (ontology learning). Language technology tools will be needed to support this process by providing an automatic analysis of the semantic structure of textual documents. In this way, free text documents will become available as semi-structured documents, from which meaningful units can be extracted automatically (information extraction) and organized through clustering or classification (text mining). The course focuses on the definition and implementation of one or more applications of language technology for the Semantic Web, e.g. ontology-based semantic tagging and information extraction; text mining for ontology development. **[8 ECTS]**

Question Answering (QA). Course *Question Answering*: Question Answering (QA) systems search for answers to a natural language-question a user has asked in a given document collection. The aim of this

course is to improve the already existing QA system QuALiM ("Question Answering with Linguistic Methods"), with which we participated in TREC 2004 and plan to participate in TREC 2005. **[8 ECTS]**

Speech Recognition and Generation. Course *Spoken Language Systems*: This course discusses various components of spoken language systems on top of speech recognition. Those components are needed to make complete systems. A particular focus will be language modeling. Specific topics are: Basics in Information Theory, Language Models, Backing-Off, Smoothing, Maximum-Entropy, Speech Understanding, Topics in Spoken Dialogue Systems, Speaker Recognition and Verification. **[6 ECTS]**

Course *Digital Signal Processing*: Signal processing is required in various disciplines, speech technology and science among them. On the one hand side hardware and implementation issues will be covered in this course. On the other hand algorithms that extract the essential information of a signal will be considered. Specific topics are: Digital Signal Processors, Real Time Programming, Microcontrollers, Feature Extraction and Feature Transforms, Noise Suppression and Filtering, Wavelets. **[5 ECTS]**

Psycholinguistics. Course *Experimental Methods in Psycholinguistics*: Speech comprehension and production: The goal of this course is to enable students to critically evaluate experimental studies in the literature and, more centrally, to run well designed experiments of their own. This seminar will focus on the fundamental concepts and skills necessary to psycholinguistic experimentation, including design, assembly, data collection, analysis and interpretation of results. Following an introduction to various experimental tasks and issues of design and analysis, students are required to conduct an experiment of their own, assembling the materials, collecting and analyzing the data, and finally presenting the results to the class. **[6 ECTS]**

Additions. Course *Introduction to Grammar Engineering using HPSG*: The implementation of linguisticallybased grammars for natural languages draws on a combination of engineering skills, sound grammatical theory, and software development tools. This course provides a hands-on introduction to the techniques and tools needed for building the precise, extensible grammars required both in research and in applications. Through a combination of lectures and in-class exercises, students will investigate the implementation of constraints in morphology, syntax, and semantics, working within the unification-based lexicalist framework of Head-driven Phrase Structure Grammar. Topics to be addressed in the course include: the use of types and features, lexical rules, constructions, semantic composition, and reversibility of grammars for parsing and generation. The weekly implementation exercises are conducted on the open-source LKB grammar development platform, and will be the basis for evaluation in the course. **[3 ECTS]**

10.2 Computer Science Modules

Students may attend courses offered by the department of Computer Science if necessary and after approval by the coordinator. Module assignments will be made on an individual basis.

10.3 Additional Information

For more info see http://www.coli.uni-saarland.de/msc/

Contact persons: Dr. Ivana Kruijff-Korbayová**,** Prof. Hans Uszkoreit

E-mail: lct-info@coli.uni-sb.de

11 University of Malta (UoM)

11.1 Language Technology Modules

Under language technology UoM is primarily offering 15 advanced credits + 15 credit project + 30 credit dissertation. In addition there are 9 introductory level credits.

11.1.1 Methodologies of Language Technology Module (LT-M1)

The **Methodologies of Language Technology** module at UoM is covered by courses on:

Text and Speech. Course *Fundamentals for Digital Signal Processing*: This unit teaches the basics and fundamentals underlying DSP as required for speech technologies. Sampling and quantising; Fourier transform; convolution; signal processing operations. Assignments will be carried out on audio, image and weather signals. **[5 ECTS]**

Foundations of Linguistics. Course *Introduction to Computational Linguistics*: Introductory course including linguistic and computational fundamentals, linguistic categories, tagging, computational morphology, computational syntax, grammar formalisms. **[2 ECTS]**

11.1.2 Computational Syntax and Morphology Module (LT-M2)

The **Computational Syntax and Morphology** module at UoM is covered by courses on:

Finite State Techniques. Course *Computational Morphology*: Morphology concepts, morphological analysis and synthesis, finite state techniques, xerox tools, concatenatative and non-concatenative morphology, computational lexicon. **[2 ECTS]**

11.1.3 Computational Semantics, Pragmatics and Discourse Module (LT-M3)

The **Computational Semantics, Pragmatics and Discourse** module at UoM is covered by courses on:

Syntax-Semantics Interface. Course *Interfacing NL Syntax and Semantics*: A practical exploration of the interface between syntactic phenomena, (e.g. verbs, simple and complex noun phrases, relative clauses), and their semantic representation using DCGs. Development of a simple NLDB query system. **[2 ECTS]**

11.1.4 LT Specialized Modules (LT-M4)

The **LT Specialized Modules** at UoM are covered with courses on:

Machine Translation (MT). Course *Machine Translation*: History, FAMT / HAMT / MAHT, Direct / Transfer / Interlingual Models, Example based MT, Translation Memory, Statistical MT, evaluation of MT systems. **[2 ECTS]**

Speech Recognition and Generation. Course *Speech Technology with Digital Signal Processing*: Speech technology concepts, speech analysis, speech synthesis, TTS speech corpora, speech recognition, noise, variability, compututational tools, HMMs and neural networks. **[5 ECTS]**

Models of Human Language Processing and Understanding. Course *Natural Language Algorithms*: This unit presents and examines algorithms used at different levels of NL analysis, notably: words (finite state tools and techniques, stemming), sentences (left corner, earley etc), texts (tokenisation, tagging). **[2 ECTS]**

Additions. Course *Information Extraction*: The IE problem, overall architecture of IE systems, text representations and tokenisation, named entity recognition, coreference determination, representation of results, text mining. **[2 ECTS]**

Course *Natural Language Generation*: NLG compared to NLU, components of NLG systems, NLG and planning, generation of referring expressions, path planning and description, adaptive generation. **[2 ECTS]**

11.2 Computer Science Modules

11.2.1 Data Structures, Data Organisation and Processing Module (CS-M1)

The **Data Structures, Data Organisation and Processing** module at the UoM is covered with courses on:

Information Retrieval (IR). Course Adaptive Hypertext Systems. Adaptive Hypertext Systems must be able to discover, represent, and manipulate user interests and requirements as users navigate and search through a hyperspace, and then adapt the organisation of and the presentation of information accordingly. This study-unit introduces essential components of Adaptive Hypertext Systems: information storage and retrieval, hypertext reference models, cognitive modelling, and adaptation techniques. **[6 ECTS]**

11.2.2 Logic, Computability and Complexity Module (CS-M2)

The **Logic, Computability and Complexity** module at the UoM is covered with courses on:

Computability Theory. Course *Computability and Complexity*: Chomsky's language hierarchy, Pumping lemmata to prove limits of each of Chomsky's language classes, Turing Machines and the limits of computing, Intractability, and NP-completeness. **[2 ECTS]**

Discrete Mathematics. Course *Mathematics of Discrete Structures*: The course is primarily aimed to introduce the basic mathematical tools that are required for the formal and rigorous treatment of the various aspects of computing. Topics include Set theory, Relations and Functions, Natural Numbers and cardinality, Group theory, Graph theory. This unit is also intended to introduce the concept of logic as a tool for studying the validity of arguments. Topics include an introduction to: Predicate and propositional logic, Logical equivalence and satisfiability The syntax of First Order Logic, Axioms and inference rules, Proof systems and techniques, Set theory. **[4 ECTS]**

Algorithms. Course *Algorithmics*: Theory and use of algorithms. Programming language concepts including Turing Machines and Turing powerful languages, programming constructs, and the theory of algorithms; imperative, object-oriented and declarative programming. Concepts of algorithms and data structures, highlighting the relation that exits between the two. Correctness and efficiency will be emphasized as theSorting, searching, graph and tree processing, and hashing techniques. Abstract data types (ADTs) will be formally defined and illustrated with case studies for list, stack, queue, priority queues, and MAX heaps. Cousework includes a small software project. **[6 ECTS]**

11.2.3 Formal Languages and Algorithms Module (CS-M3)

The **Formal Languages and Algorithms** module at the UoM is covered with courses on:

Formal Grammars and Languages Hierarchy. Course *Formal Languages and Automata*: This module deals with the formal treatment of languages and automata (or machines) to recognise languages. The aims are not only at instilling the basic notions of languages, grammars and automata using formal mathematical notation but also provides a practical perspective, by applying the mathematical results to design parsers. The syllabus includes Formal languages and grammars; Regular languages; Context-free languages; Closure properties of regular and context-free languages; Normal forms for grammars; Recognition algorithms for grammars. **[3 ECTS]**

Parsing and Compiler Design. Course *Compiling Techniques*: This unit discusses the basic concepts of compilers for computer programming languages and compiling techniques. It includes a presentation of the theory of formal languages and presents the concepts and texhniques behind: Lexical Analysis; Syntax Analysis and Code Generation. **[4 ECTS]**

Search Techniques and Constraint Resolution. Course *Techniques in AI* : This module aims to familiarize students with the nature of AI problems and related practical solution techniques. Topics include: Problem Solving; Searching - Graphs; Agenda/Means-End Analysis/Constraint Satisfaction; Game Playing; Knowledge Representation; Expert Systems; Planning - Blocks World. The second half of the course is oriented towards Search and Optimization problems; Function Approximation problems (i.e. Function Learning and Synthesis); ANNs (theory, architecture, design, and implementation); Genetic and Evolutionary Algorithms (background, design, and implementation); Montecarlo Techniques applied to Search and Optimization problems; Artificial Life; WISARD neural network for scene analysis; Feature extraction and the Hough Transform. **[3 ECTS]**

11.2.4 CS Specialized Modules (CS-M4)

The **CS Specialized Modules** at the UoM are covered with courses on:

Semantic Web. Course *Agent Technology*: The course deals with the theory and practice of intelligent agents including basics (history, subject matter), software architecture, properties and models of agents, agent inter connectors and agent systems, aspect models, mobility, co-ordination and security, architecture types for agentbased application systems, commercial agent application, standardization efforts, web services, ontologies, markup languages, semantic web and future directions. **[6 ECTS]**

Machine Learning. Course *Machine Learning, Expert Systems and Fuzzy Logic*: Machine Learning includes Principles of learning machines, Gold's Theorem; Concepts and Categories in Cognitive Science; Computational learning theory (COLT); PAC-learning; Grammatical inference; Concept learning; Find-S, Candidate Elimination, and the ID-3 learning algorithms. Pattern Recognition includes Clustering techniques; Linear discriminant analysis; Pattern Feature Extraction; Pattern Understanding; Advanced Neural Networks (Hopfield, Kohonen networks); Support Vector Machines. **[6 ECTS]**

Additions. Course *Formal Methods*: When designing safety-critical systems, it is not sufficient to test the software / hardware written, but it may be necessary to mathematically verify that the system works correctly. This course explores different topics in this research area, including mathematical tools (such as logics, and property languages), formal modeling of computer languages and systems, and techniques used for automatic model checking of such systems. Syllabus: Temporal Logic, process calculi; Synchronous programming, circuit semantics; Model Checking of software and hardware. **[4 ECTS]**

11.3 Additional Information

For additional information see http://staff.um.edu.mt/mros1/HLST/.

Contact person: Michael Rosner

E-mail: mike.rosner@um.edu.mt

12 Shanghai Jiao Tong University

12.1 Computer Science Modules

Note: 1 Credit = 17 Lecture Hours; 1 Lecture Hour = 45 Min.

12.1.1 Data Structures, Data Organization and Processing Module (CS-M1)

The Data Structures, Data Organization and Processing module at SJTU is covered with courses on:

Algebraic Data Types. Course Applied Algebra. Introduction of Finite Field; Euclidean Domain and the Unique Factorization Theorem for Euclidean Domain; Building Finite Fields from Euclidean Domain; Properties of Finite Fields; Factoring Polynomials over Finite Fields; The Concept of Linear Feedback Shift Register (LFSR) and Applications to LFSR; The Application to Cryptography; The Application to Error-correction Codes. [3 Credits]

Database Theory and Technology. Course Advanced Dababase Techniques. DBMS Implementation Techniques; New Data Models; Database Systems Architectures; New Applications. [3 Credits]

Web Data Types and Service. Course Web Services and .Net Framework Technologies. Reality of Web Services; Microsoft .NET Framework; Deliverables of WS-I; Basic Web Services Standards; Implementation and Deploying of Web Services using Standard .NET APIs; .NET Platform for Web Services. [3 Credits]

Information Extraction (IE). Course Internet-Based Information Extraction Technology. Introduction; Named Entity Recognition; Chinese Information Extraction; Template-based Information Extraction; Web Mining; Opinion Mining. [3 Credits]

12.1.2 Logic, Computability and Complexity Module (CS-M2)

The Logic, Computability and Complexity module at SJTU is covered with courses on:

Logic. Course Applied Logic: PL, the Design,Correctness & Complexity of a Boolean SAT Solver; FO, its Proof Theory & Model theory, its Expressivity & Undecidability; TCL, Universal & Existential 2nd. Logic: an Objective Modelling Language Alloy & its Applications; Verification by Model Checking; LTL, BTLs, an Open Source Verification Tool NuSMV etc.; Program Verification; Model Logics and Agents; BDDs & Symbolic Model Checking. [3 Credits]

Course **Concurrency Theory**: Concurrent Systems & Basic Definition of CCS; Strong Bisimulation and Strong Equivalence Relation; Weak Bisimulation, Weak Equivalence Relation; Application Examples; Observation of Congruence Relations; CCS as a Model for Programming Languages; Operator, Norms and Logic; Productive Ability of Several Variants of CCS; Introduction to other Models in Process Calculus and other Computational Models. [3 Credits]

Learning and Inference. Course Statistical Learning and Inference: Introduction; Overview of Supervised Learning; Linear Methods of Regression; Linear Methods for Classification; Basis Expansions and Regularization; Kernel Methods; Model Assessment and Selection; Model Inference and Averaging; Additive Models, Trees, and Related Methods; Boosting and Additive Trees. [3 Credits]

Course **Stochastic Processes and Queuing Theory**: Poisson Processes; Markov Processes; Queueing Theory. [2 Credits]

12.1.3 **Formal Languages and Algorithms Module (CS-M3)**

The Formal Languages and Algorithms module at SJTU is covered with courses on:

Formal Language. Course Programming Language: Introduction; Untyped Lambda Calculus; Simply-typed Lambda Calculus; Propositional Logic; First and Second-order Logic; Curry-Howard Isomorphism; System F; Type Inference; Products, Unions, Existentials; Axiomatic Semantics; Type theory. [3 Credits]

Algorithm. Course Concurrency Theory: Concurrent Systems & the Basic Definition of CCS; Strong Bisimulation and Strong Equivalence Relation; Weak bisimulation, Weak Equivalence Relation; Application Examples; Observation of Congruence Relations; CCS as a Model for Programming Languages; Operator, Norms and Logic; Productive Ability of Several Variants of CCS; Introduction to other Models in Process Calculus and other Computational Models. [3 Credits]

Course **Cryptographic Algorithms and Protocols**: Shannon Theory; Block Cipher; AES Hash Function; RSA Public-key Cryptography; Integer Factoring; Discrete Logarithm Cryptography; Discrete Logarithm Digital Signature. [3 Credits]

Course **Information and Coding Theory**: Measurement of Information; Weakly Typical Sequence; Strongly Typical Sequence; Block Codes; Linear Codes; Cyclic Codes; Decoding Principles. [3 Credits]

Course **Soft Computing: Introduction**: Soft-computing Development Context; Artificial Neural Networks; Fuzzy Logic; Evolutionary Algorithms; Fuzzy Logic and Neural Network Integration and Collaboration; Evolutionary Algorithms and Neural Network Integration and Collaboration; Evolutionary Algorithms and Fuzzy Logic Integration and Collaboration; Latest Developments of Soft Computing. [3 Credits]

12.1.4 **CS Specialized Modules (CS-M4**)

The CS Specialized Modules at SJTU are covered with courses on:

Artificial Intelligence (AI). Course Artificial Intelligence: Production Systems; Neural Networks; Genetic Programming; Computer Vision; Heuristic Search; Logic; Knowledge Representation and Reasoning; Bayes Networks; Automatic Planning and Multi-agent Communication. [3 Credits]

Machine Learning (ML). Course Machine Learning: Introduction; Concept Learning and the General-to-Specific Ordering; Decision Tree Learning; Artificial Neural Networks; Evaluating Hypotheses; Bayesian Learning; Computational Learning Theory; Instance-Based Learning; Genetic Algorithms; Learning Sets of Rules; Analytical Learning; Combining Inductive and Analytical Learning; Reinforcement Learning. [3 Credits]

Natural Language Processing (NLP). Course Natural Language Understanding: Brief Introduction of NLU (including the Development History and Application of Theories, Methods and Techniques etc.); Machine Dictionary and Lexical Analysis (including WordNet, Generative Lexicon, and Finite State Model for Lexical Analysis etc.); Syntactical Analysis (including General Grammars, Augmented Grammars, the Corresponding Automatic Analysis, and Syntactical Ambiguity Resolution etc.); Semantic Analysis (including Logical Analysis, Relationship between Semantics and Syntax, and Semantic Ambiguity Resolution etc.). [3 Credits]

Bioinfomatics. Course **Bioinfomatics**: Mimicking Nature for Problem Solving; Complex Networks, Structure, Comparison and Evolution; Associative Memories and Neuronal Coding and Decoding; Network Modeling; Self-organizing Maps; Alignment Problems; Memory Structures, Permutations and Combinations; Believe Propagation and Interferences; Permutation-based Assessment Methods; 3D Structural Modeling and Indirect Interactions; Artificial Life; The Power and Computational Complexity of Computational Intelligence Models. [3 Credits]

Graphics. Course **Computer Graphics**: Basic Raster Graphics Algorithms for Drawing 3D Primitives, Geometric Transformations in 2D and 3D Space, Viewing in 3D, Representing Curves and Surfaces, Visual Reality and Computer Animation. [3 Credits]

Course **Digital Image Processing**: Introduction; Basis of Digital Image Processing; Image Transform; Image Enhancement; Image Recovery and Reconstruction; Image Compression; Image Segmentation and Image Description. [3 Credits]

Communication. **Computer Communications and Networks**: Basic Principles of Computer Networking; Public Protocols; Typical Network Applications; Ethernet, Wireless Local Area Networks; Internetworking Devices; Ethernet Switches. [2 Credits]

12.2 Additional Information

For additional information see http://lt-lab.sjtu.edu.cn/en/

Contact person: Tianfang Yao

E-mail: yao-tf@cs.sjtu.edu.cn

13 University of Melbourne

13.1 General Information

The LCT Masters at The University of Melbourne is structured as follows:

Overall course = 120 ECTs, made up of 75 ECTs of course work and 45 points of Masters thesis

The coursework must include the following components:
- three "foundation" subjects (total of 22.5 ECTs)
- "Research Methods" (7.5 ECTs)
- one "professional tools" subject
- where deemed necessary by the University (as part of the terms of the acceptance, based on English proficiency), "Engineering Communication"

These subjects align with the structure of the LCT Masters as follows:

Subject	Mandatory?	Semesters	ECTs
Foundation LT subjects (7.5 ECTs):			
Knowledge Technologies	Y	1&2	7.5
Foundation CS subjects (22.5 ECTs):			
Declarative Programming	Y	1&2	7.5
Distributed Systems	Y	1&2	7.5
Research Methods	Y	1	7.5
Foundation Numeric/Communication subjects (7.5–15 ECTs):			
Thinking and Reasoning with Data	Y	1	7.5
Engineering Communication	Y*	1&2	7.5
Elective CS subjects (up to 22.5 ECTs):			
Statistical and Evolutionary Learning	N	2	7.5
Constraint Programming	N	2	7.5
Software Agents	N	2	7.5
Advanced Database Systems	N	1	7.5
Elective LT subjects (37.5–50 points):			
Web Search and Text Analysis	N	1	7.5
English Phonetics and Phonology	N	2	7.5
Grammar and Discourse	N	1	7.5
Issues in Linguistic Research	N	1	15
MSc Thesis	Y	1&2	45

*Engineering Communication must be taken where deemed necessary by the Melbourne School of Engineering on the basis of English language competence; if not deemed necessary, students will not be required to take this subject.

13.2 Descriptions of the individual subjects

are as follows.

Knowledge Technologies: Much of the world's knowledge is stored in the form of unstructured data (e.g. text) or implicitly in structured data (e.g. databases). In this subject students will learn algorithms and data structures for extracting, retrieving and storing explicit knowledge from various data sources, with a focus on the web. Topics include: data encoding and markup, web crawling, clustering, regular expressions, pattern mining, Bayesian learning, instance-based learning, document indexing, database storage and indexing, and text retrieval.

Declarative Programming: Declarative programming languages provide elegant and powerful programming paradigms that every programmer should know. Topics covered include functional programming, logic programming, constraint programming; declarative programming techniques, including higher order programming and the exploitation of advanced type systems; declarative languages as a competitive advantage, and how they fit into an environment dominated by imperative languages.

Distributed Systems: Topics covered include: introduction, principles and paradigms, design issues, communication, processes, naming, synchronization, consistency and replication, fault tolerance, and security issues in distributed systems and applications; distributed computing environments and standard toolkits, case studies in distributed systems and applications.

Research Methods: The aim of scientific research is to produce new knowledge. To be useful, new knowledge must be able to stand up to critical scrutiny, and its presentation to other researchers and/or to the public must be persuasive. This subject is an introduction to the processes of science as they apply to computer science, including designing experiments, locating relevant literature, writing papers, giving presentations and refereeing. Underlying all of these, the subject will foster the development of critical thinking, a sceptical, scientific perspective, and scientific ethics.

Thinking and Reasoning with Data: What conclusion can be drawn from a pool of data? How can a scientist draw meaningful conclusions while not overreaching? How can modelling help the scientist interpret data? This subject will address these questions by teaching students critical thinking and data analysis skills. After completing this subject students will understand the basic principles of sampling and experimental design, how the results of statistical analyses are reported, the statistical thinking behind common statistical procedures and will be able to carry out a range of standard statistical techniques.

Engineering Communication: This subject introduces the nature of engineering work, at the heart of which is communication and problem solving using sustainability principles. Specific topics include: skills required for engineering practice; the engineering recruitment process; engineering problem solving; systems thinking and design; teamwork; meetings and group dynamics; oral and written communication; use of library services and information services; sharing information and knowledge management; creation of alternative solutions; and evaluation and decision making processes using sustainability.

Statistical and Evolutionary Learning: With exponential increases in the amount of data becoming available in fields such as finance and biology, and on the web, there is an ever greater need for methods to detect interesting patterns in that data, and classify novel data points based on curated data sets. Statistical machine learning and evolutionary computation provide the means to perform this analysis automatically, and in doing so to enhance understanding of general processes or to predict future events. Topics covered will include:

association rules, clustering, instance-based learning, statistical learning, evolutionary algorithms, swarm intelligence, neural networks, numeric prediction, weakly supervised classification, discretisation, feature selection and classifier combination.

Constraint Programming: Constraint programming is used to solve constrained satisfaction and optimization problems such as scheduling and allocation, which are of vital importance to modern business. Constraint programming allows these problems to be modeled at a high level, and allows the solving to take advantage of the structure of the problem. Topics covered will be: modelling with constraints; Algorithms for manipulating constraints; finite domain constraint solving; global constraints; programming search; and hybrid constraint solving.

Software Agents: Software agents are software systems that use automated reasoning algorithms operating on symbolic knowledge representations to determine how they should interact with their environment (the real world, human beings, or other agents) in order to achieve the goals set for them. Agent modelling and programming techniques are becoming increasingly popular for solving problems for robotic, artificially intelligent or inter-networking processes. The subject covers the principles of agent-oriented modelling and the underlying automated reasoning and agent programming techniques involved. Topics covered include: agent modelling of goals, roles and organisations, planning, sensing and acting, introduction to modal logic and model theory, epistemic logic, theorem proving, dynamic logic, the situation calculus, logical regression, non-deterministic programming, concurrency and multi-agent programming. The subject focuses on the foundations that enable agents to reason autonomously about goals, perception, actions and the knowledge of other agents during collaborative task execution.

Advanced Database Systems: Many applications require access to very large amounts of data. These applications often require reliability (data must not be lost even in the presence of hardware failures), and the ability to retrieve and process the data very efficiently. The subject will cover the technologies used in advanced database systems. Topics covered will include: transactions, including concurrency, reliability (the ACID properties) and performance; and indexing of both structured and unstructured data. The subject will also cover additional topics such as: uncertain data; Xquery; the Semantic Web and the Resource Description Framework; dataspaces and data provenance; datacentres; and data archiving.

Web Search and Text Analysis: The web is a vast and expanding storehouse of semi-structured textual information. Accessing and processing this information is one of the major challenges of the information age. In this subject, students study the technologies behind search engines, spam filtering, plagiarism detection, information extraction, question answering and newly emerging fields of information engineering. Topics include: web indexing, query evaluation, probabilistic language modelling, document classification and filtering, grammar and spelling correction, topic detection, cross-language information retrieval, machine translation and summarisation.

English Phonetics and Phonology: This subject focuses on aspects of the English sound system. Students gain a detailed understanding of the sound system of different varieties of English. The main topics covered include: phonetic analysis (vowel and consonant systems, phonotactics, stress and intonation) of selected English varieties including L2 varieties of English. Throughout the subject, material is drawn from different varieties of English, and in particular, English spoken by second language learners, to gain an understanding of

phonological transfer. This subject also provides the chance for training in the accurate perception and production of English speech, transcription, and basic acoustic phonetic analysis.

Grammar and Discourse: This subject explores the main areas of grammar and discourse, including morphology, syntax, discourse analysis, and pragmatics, from an applied linguistics perspective. Throughout the subject, material is drawn from different spoken and written varieties of English, including various regional and social varieties as well as Standard English. Through seminars and practical application students gain a systematic understanding of the major features of English grammar and discourse and develop their skills in transcribing spoken English and using techniques of linguistic analysis in the study of English.

Issues in Linguistic Research: This subject takes a broad and historically based view of some of the great topics that have preoccupied thinkers in linguistics, and how they interrelate, by juxtaposing the history of ideas on each of a dozen topics with relevant contemporary debates. Topics to be covered include defining properties of language, sign and system, arbitrariness versus constraints on the system, the role of typology, the role of formal modelling of language systems, linguistic relativity, synchrony versus diachrony, linguistics as a science, the linguistic system and the community of usersm, the boundaries of the language system, functionalism, adaptation and evolution of language systems. categories, classical and otherwise, and language, mind and brain.

13.3 Additional Information

For additional information see http://www.cs.mu.oz.au/research/lt/

Contact person: Dr. Timothy Baldwin

E-mail: tb@ldwin.net