CS635+CS728: Updated Syllabus

INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

Proposal for New Academic Course

Name of the Academic Unit:
COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

Notes:

*1. The duly signed hard copy of course proposal should be sent to the Academic office. Also, the soft copy **(doc & pdf file)** should be sent to the Jt. Dy. Registrar (Academic) <dracad@iitb.ac.in>.*

*2. For courses to be offered in **Autumn semester**, proposal be sent to the academic office **latest by April** and for courses to be offered in **Spring semester**, proposal be sent **latest by October.***

3. Tick appropriate items, Add separate sheet (if required)

*4. * - Refer last page of: https://www.iitb.ac.in/newacadhome/MTechRules.pdf*

<table>
<tr><td>1</td><td>Title of the Course
(Limited to 50 characters, including the spaces between the words):</td><td>Indexing and retrieving text and graphs</td></tr>
<tr><td>2</td><td>Credit Structure
C = 2(L+T) +P for Full Semester; C = L+T+0.5*P for Half Semester.</td><td><table><tr><th>Lectures (L)</th><th>Tutorials (T)</th><th>Practicals (P)</th><th>Total Credits (C)</th></tr><tr><td>3</td><td>0</td><td>0</td><td>6</td></tr></table></td></tr>
<tr><td>3</td><td>Duration of Course</td><td>☐ Half semester
☑ Full semester</td></tr>
<tr><td>4</td><td>Academic Programme for which course is applicable</td><td>☑ Undergraduate
☑ Postgraduate
☑ Ph.D.
☐ M. Engineering</td></tr>
<tr><td>5</td><td>Desired course number range *
Usually course numbers assigned for:
UG: 101 - 499
PG: 501- 599, 701 – 799
UG & PG – 601 - 699
PhD: 801 and above</td><td>CS601 through CS699.
Courses CS635 and CS728 will be retired.</td></tr>
<tr><td>6</td><td>Type of Course</td><td>☑ Theory
☐ Seminar
☐ Lab
☐ Project
☐ Non-credit
☐ Studio</td></tr>
<tr><td>7</td><td>Course Tag</td><td>☐ Core
☑ Department Elective
☐ Institute Elective
☐ Minor
☐ Honor
☐ SLP
☐ R & D Project</td></tr>
<tr><td>8</td><td>Course would be offered in</td><td>☑ Autumn Semester (1st)
☐ Spring Semester (2nd)
☐ Both (1st & 2nd)</td></tr>
<tr><td>9</td><td>Pre-requisite for the course (if any)</td><td></td></tr>
</table>

<table>
<tr><td>10</td><td>Justification for Introduction of course</td><td>This course will replace CS635. That code has been reused for decades while the course was overhauled multiple times. Recent developments in deep retrieval make it necessary to launch an entirely new course.</td></tr>
<tr><td>11</td><td>Contents of the course</td><td>Classic inverted index for sparse retrieval; relevance ranking; learning to rank; latent semantic indexing and topic models; word embeddings; dense retrieval; locality sensitive hashing; graph proximity and matching; graph indexing and retrieval.

Also see details in the attached sheet.</td></tr>
<tr><td>12</td><td>Texts and References
(Minimum 5 – Maximum 8)
(Complete name of Author/ Title/ Edition/ Publisher/ Volume, Web references/
e-references, year of publication, etc.)</td><td>https://www.amazon.in/Managing-Gigabytes-Compressing-Multimedia-Information/dp/1558605703
https://nlp.stanford.edu/IR-book/information-retrieval-book.html
https://ciir.cs.umass.edu/irbook/
https://www.nowpublishers.com/article/Details/INR-061
https://www.amazon.in/Mining-Graph-Data-Diane-Cook/dp/0471731900
https://www.amazon.in/Graph-Data-Mining-Application-Management-ebook/dp/B099MWV845
Social Network Analysis</td></tr>
<tr><td>13</td><td>Names of Instructors
(Require the names of at least two permanent faculty members of IITB for core course and one permanent faculty member of IITB for elective course.</td><td><table><tr><th colspan="2">Name of Instructor & Academic Unit</th></tr><tr><td>1.</td><td>Soumen Chakrabarti</td></tr><tr><td>2.</td><td>Ganesh Ramakrishnan</td></tr><tr><td>3.</td><td>Abir De</td></tr></table></td></tr>
<tr><td>14</td><td>Existing overlapping course(s)
(Offered by the same or other academic unit)</td><td>No significant overlap.</td></tr>
<tr><td>15</td><td>Another Academic unit to whom the course may be relevant (if any):</td><td>CMInDS</td></tr>
</table>

The above proposal for a new course is found to be acceptable by (DUGC/ DPGC/ PGC) in its meeting held on _____. The committee recommends this course proposal for consideration of UGPC / PGPC.

_____

Signature of the Convener,

DUGC/ DPGC/ PGC of the Academic Unit

Date:

CS635[1]→New course code CS6101

Indexing and retrieving text and graphs

~~Web Search and Mining~~

~~Information Retrieval & Mining for Hypertext & the Web~~ 🤮

(Owing to the rapid research developments, this course has been continually overhauled over the years. Autumn 2023 was the first semester after the “large language model” revolution, so we further modified and enhanced the course, discarding outdated material, and including new material suitable for the current workplace, as well as continuing research. Because of logistical limitations, we will continue using the old course codes until Spring 2024 and plan to assign two new course codes commencing Autumn 2024.)

Summary: Searching structured and unstructured information sources was the foundation of Google and is of immense value to other companies like Amazon, Bing and IBM. This course focuses on core scientific and engineering principles driving indexing, retrieving and analyzing text and graphs, which are at the heart of search systems. We will focus on scalable algorithms for indexing text and graphs, and modern machine learning techniques to represent and rank information items like passages and subgraphs.

Target: Btech4 and beyond, but, in principle, anyone with some basic background (JEE/GATE math, prob-stat, algorithms). Attendees will learn modern techniques widely used in industry as well as the frontiers of current research. Both classical and neural learning methods will be explored. The course will be hands-on with programming assignments.

Evaluation (tentative): Best 60% from 3x 30% written exams; 30% programming assignments (possibly Kaggle-like contests); 10% weekly safe quizzes. Extra credit for course projects and paper reviews.

Topics:

- Classic information retrieval
  - Tokenization before the age of transformers
  - Dictionary, compression
  - Inverted list, compression
  - TFIDF, vector space model
  - Sparse retrieval query processing, DAAT, TAAT
- Ranking
  - Text classification, SVM
  - Ranking evaluation and losses
  - Learning to rank for sparse retrieval
  - Reward for term proximity
  - Exposure bias and mitigation
- Dense text representation learning
  - Pseudo relevance feedback
  - Latent semantic indexing (Wikipedia, Stanford notes)

[1] Course code will eventually change.

    - Word2vec and GloVe (Stanford slides, notebook, notebook)
    - ConvNet for (late interaction) retrieval
    - The need for contextual embeddings: LSTM, transformer
    - Efficiency issues for long attention context; MEGABYTE
- Dense indexing
    - Need for locality sensitive hashing (LSH)
    - Major LSH families: Hamming, cosine, L2, L1, dot, Jaccard
    - DPR and dense indexing, ANNS
    - Contrastive sampling, batch design
    - ColBERT, SPLADEv2, GTR, XTR, WARP (survey)
    - Symmetric vs asymmetric LSH, HNSW
    - Machine learning for data-sensitive DPR
    - Extreme classification, DEXA etc
    - Generative retrieval: DSI
- Graph search and mining
    - Social graphs (Web, social media)
    - Knowledge graphs (WordNet, Wikidata, Wikipedia)
    - Salient properties (degree, diameter, dense cores)
    - Notions of graph proximity for diverse applications
    - Hitting time, commute time, escape time, effective conductance
    - (Personalized) PageRank, SimRank, TotalRank etc.
- (Plus latest papers on many of the above topics)

Soft prerequisites

- UG probability and statistics, e.g., An Introduction to Probability Theory and its Applications (Volume 1) by William Feller.
- UG data structures and algorithms, e.g., based on Introduction to Algorithms by CLRS.

Non-book references

- https://arxiv.org/pdf/1910.10687 term importance estimation
- https://arxiv.org/pdf/2104.07198 ◂
- https://openreview.net/pdf?id=MFPYCvWsNR ◂
- Autoregressive search engines
- https://arxiv.org/pdf/2206.14286
- An Offline Metric for the Debiasedness of Click Models
- Click Model-Based Information Retrieval Metrics
- Unbiased Learning-to-Rank Needs Unconfounded Propensity Estimation
- Unbiased Learning to Rank with Unbiased Propensity Estimation
- Policy Learning for Fairness in Ranking

INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

Proposal for New Academic Course

Name of the Academic Unit:

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

Notes:

*1. The duly signed hard copy of course proposal should be sent to the Academic office. Also, the soft copy **(doc & pdf file)** should be sent to the Jt. Dy. Registrar (Academic) <dracad@iitb.ac.in>.*

*2. For course to be offered in **Autumn semester**, proposal be sent to academic office **latest by April** and for course to be offered in **Spring semester**, proposal be sent **latest by October.***

3. Tick appropriate items, Add separate sheet (if required)

*4. * - Refer last page of: https://www.iitb.ac.in/newacadhome/MTechRules.pdf*

<table>
<tr><td>1</td><td>Title of the Course
(Limited to 50 characters, including the spaces between the words):</td><td>Deep Learning for Sequences, Graphs, and Language Models</td></tr>
<tr><td>2</td><td>Credit Structure
C = 2(L+T) +P for Full Semester; C = L+T+0.5*P for Half Semester.</td><td><table><tr><th>Lectures (L)</th><th>Tutorials (T)</th><th>Practicals (P)</th><th>Total Credits (C)</th></tr><tr><td>3</td><td>0</td><td>0</td><td>6</td></tr></table></td></tr>
<tr><td>3</td><td>Duration of Course</td><td>☐ Half semester
☑ Full semester</td></tr>
<tr><td>4</td><td>Academic Programme for which course is applicable</td><td>☑ Undergraduate
☑ Postgraduate
☑ Ph.D.
☐ M. Engineering</td></tr>
<tr><td>5</td><td>Desired course number range *
Usually course numbers assigned for:
UG: 101 - 499
PG: 501- 599, 701 – 799
UG & PG – 601 - 699
PhD: 801 and above</td><td>CS601 through CS699.
Courses CS635 and CS728 will be retired.</td></tr>
<tr><td>6</td><td>Type of Course</td><td>☑ Theory
☐ Seminar
☐ Lab
☐ Project
☐ Non-credit
☐ Studio</td></tr>
<tr><td>7</td><td>Course Tag</td><td>☐ Core
☑ Department Elective
☐ Institute Elective
☐ Minor
☐ Honor
☐ SLP
☐ R & D Project</td></tr>
<tr><td>8</td><td>Course would be offered in</td><td>☐ Autumn Semester (1st)
☑ Spring Semester (2nd)
☐ Both (1st & 2nd)</td></tr>
<tr><td>9</td><td>Pre-requisite for the course (if any)</td><td></td></tr>
</table>

<table>
<tr><td>10</td><td>Justification for Introduction of course</td><td>This course will replace CS728. That code has been reused for decades while the course was overhauled multiple times. Recent developments in deep retrieval make it necessary to launch an entirely new course.</td></tr>
<tr><td>11</td><td>Contents of the course</td><td>Probabilistic sequence and graph modeling from pre-deep-NLP era;
Pre- and early neural representations of text and graphs;
Knowledge graphs; RNNs and transformers; graph transformers;
Retrieval augmented inference and generation;
Question answering and semantic interpretation;
Language model adaptation, prefix tuning, instruction tuning.</td></tr>
<tr><td>12</td><td>Texts and References
(Minimum 5 – Maximum 8)
(Complete name of Author/ Title/ Edition/ Publisher/ Volume, Web references/
e-references, year of publication, etc.)</td><td>• https://nlp.stanford.edu/fsnlp/
• https://www.amazon.in/Language-Processing-Synthesis-Lectures-Technologies/dp/1627052984
• https://web.stanford.edu/~jurafsky/slp3/
• https://onlinecourses.nptel.ac.in/noc25_cs45/preview
• Introduction to Large Language Models</td></tr>
<tr><td>13</td><td>Names of Instructors
(Require the names of at least two permanent faculty members of IITB for core course and one permanent faculty member of IITB for elective course.</td><td><table><tr><td colspan="2">Name of Instructor & Academic Unit</td></tr><tr><td>1.</td><td>Soumen Chakrabarti</td></tr><tr><td>2.</td><td>Sunita Sarawagi</td></tr><tr><td>3.</td><td>Ganesh Ramakrishnan</td></tr></table></td></tr>
<tr><td>14</td><td>Existing overlapping course(s)
(Offered by the same or other academic unit)</td><td>CS772 (Deep Learning for Natural Language Processing) has considerable topic overlap. However, the proposed course is more theoretical and algorithmic and less about NLP applications. There is more stress on analyzing the expressivity of various networks and learning how to design new networks for various tasks. The probabilistic graph modeling, knowledge graph, retrieval augmentation and question answering components are exclusive to the proposed course. These are color coded above.</td></tr>
<tr><td>15</td><td>Another Academic unit to whom the course may be relevant (if any):</td><td>CMInDS</td></tr>
</table>

The above proposal for a new course is found to be acceptable by (DUGC/ DPGC/ PGC) in its meeting held on _____. The committee recommends this course proposal for consideration of UGPC / PGPC.

_____

Signature of the Convener,

DUGC/ DPGC/ PGC of the Academic Unit

Date:

CS728[2]→new course code

Deep Learning for Sequences, Graphs, and Language Models
Machine Learning for Natural Language and Knowledge Graphs
Advances in Natural Language and Knowledge Representation
~~Deep Learning for Text and Knowledge Graphs~~
~~Deep Knowledge Representation, Search, and Complex Inference~~
~~ASC = Organization of Web Information~~

As of Spring 2024, there is no hard prerequisite for taking CS728.
Students of all departments are permitted to enroll.

Topics:

- Some classes of learning problems involving sequences and graphs
  - Whole sequence classification
  - Labeling tokens in a sequence, various transition constraints
  - Labeling nodes in a graph
  - Predicting if a node pair constitutes an edge
- Probabilistic sequence and graph modeling from pre-deep-NLP era
  - HMM, MEMM, MRF, linear CRF and HMSVM
  - Factor graphs, message passing / belief propagation
- Pre- and early neural representations of text and graphs
  - Quick review of (non contextual) word embeddings: word2vec, GloVE
  - Graph neural network (GNN) families: GCN, RGCN, GraphSAGE, readouts
  - Applications to graph matching tasks: edit distance, isomorphism
- Knowledge graphs (KGs, e.g. Wikidata)
  - Representation, completion, alignment
- Short review of LSTMs, then transformers
  - Easy-to-use categorization and nomenclature among encoder-decoder, decoder-only, causal attention, etc.
  - Relative position encoding [1] + Rotary embedding [2]
  - Long contexts (linear attention [3, 4] + sparse attention [5, 6]), decoding methods (greedy, beam, nucleus), MEGABYTE
- Graph transformer, TokenGT, RT
- NLP tasks that were solved using custom techniques now all under a transformer umbrella
  - POS, NER, relation classification
  - Entity linking, possibly coref
  - Closed-book question answering
  - Multi-task transformer objectives (GLUE, superGLUE etc.)
- Why closed book LLMs will not solve all problems
  - The case for retrieving from structured and unstructured sources
  - Key-value memory networks, use in QA for simple paths in KGs
  - Early HotPotQA solvers, pointer networks

[2] Course code will eventually change.

    - Fusion-in-Decoder, retrieval augmented generation (REALM, Retro)
- Open-book question answering
    - KBQA/KGQA
        - Benchmarks WebQSP, CWQ, GrailQA
        - “IR methods” EmbedKGQA, GraftNet, PullNet, Subgraph-NSM, NodePiece, CLOCQ
        - Semantic interpretation TIARA
        - KG + LLM-planning: Reasoning on graphs
    - Corpus-QA (passage based)
        - Benchmarks SQuAD, HotputQA, NQ
        - Simple (single clause/hop)
        - Multi-hop
    - Table + text QA
- Semantic interpretation
    - Text2sql: Seq2seq → bottom-up
    - Neural programmer-interpreter (pre-LLM)
- From LM to LLM — what changed?
    - Retrieval augmented generation [13]
        - Constrained generation CoLD
        - Diffusion models for text DiffusionLM [14]
        - Provenance and hallucination-avoidance [15, 16]
    - Multi-turn, interactive, dialog
    - LLMs with modular tools [17, 18, 19]
    - Language model adapters
        - Older like K-adapter, adapterhub
        - LORA [20] and more recent [21]
    - Prompt design (starting around T5) and continuous prompt tuning
        - Fun with poisoning and jailbreaking [7, 8]
        - Peculiar biases [9]
    - Memorization [10] and response editing [11, 12]
    - Model distillation via logit tracking
- (Plus latest papers on many of the above topics)

Soft prerequisites

- The Autumn course I have proposed along with this course.
- Foundations of Machine Learning.
- Probability and Statistics.

Non-book references

[1] Shaw, et al., Self-Attention with Relative Position Representations, https://arxiv.org/abs/1803.02155
[2] Su, et al. RoFormer: Enhanced Transformer with Rotary Position Embedding, https://arxiv.org/abs/2104.09864v5
[3] Peng, et al., Random Feature Attention, https://arxiv.org/abs/2103.02143
[4] Choromanski, et al., Rethinking attention with Performers, https://arxiv.org/abs/2009.14794
[5] Beltagy, et al., Longformer: The Long-Document Transformer, https://arxiv.org/abs/2004.05150
[6] Zaheer, et al., Big Bird: Transformers for Longer Sequences, https://arxiv.org/abs/2007.14062
[7] Qi, et al., Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To!, https://openreview.net/forum?id=hTEGyKf0dZ
[8] Wolf, et al., Fundamental Limitations of Alignment in Large Language Models, https://arxiv.org/abs/2304.11082
[9] Berglund et al., The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". https://arxiv.org/abs/2309.12288
[10] Biderman, et al., Emergent and Predictable Memorization in Large Language Models, https://arxiv.org/abs/2304.11158
[11] Meng, et al. Locating and Editing Factual Associations in GPT. https://arxiv.org/abs/2202.05262
[12] Yao, et al. Editing Large Language Models: Problems, Methods, and Opportunities. https://arxiv.org/abs/2305.13172
[13] Gao, et al. Retrieval-Augmented Generation for Large Language Models: A Survey. https://arxiv.org/abs/2312.10997
[14] Li, et al. Diffusion-LM Improves Controllable Text Generation. https://arxiv.org/abs/2205.14217
[15] Zhang, et al. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models. https://arxiv.org/abs/2309.01219
[16] Xu, et al. Hallucination is Inevitable: An Innate Limitation of Large Language Models. https://arxiv.org/abs/2401.11817
[17] Schick, et al. Toolformer: Language Models Can Teach Themselves to Use Tools. https://arxiv.org/abs/2302.04761
[18] Cai, et al. Large Language Models as Tool Makers. https://arxiv.org/abs/2305.17126
[19] Paranjape, et al. ART: Automatic multi-step reasoning and tool-use for large language models. https://arxiv.org/abs/2303.09014
[20] Hu, et al. LoRA: Low-Rank Adaptation of Large Language Models. https://arxiv.org/abs/2106.09685
[21] Hayou, et al.. LoRA+: Efficient Low Rank Adaptation of Large Models. https://arxiv.org/abs/2402.12354
[22] Earth is flat?
[23] Neural spacetime; neural snowflakes
[24] LMs prefer what they know