Natural Language Processing for Data Science (DSA4213)

Doudou Zhou
https://doudouzhou.github.io/

Semester 1, AY2025/2026

Class Room: UT-AUD1	Class Hours: TUE, 5-7 pm;
Class Room: UT-AUD2	Class Hours: FRI, 2-4 pm;
Office: S16-6-110	E-mail: ddzhou@nus.edu.sg

The material below is subject to change. Please stay updated by checking regularly. Last updated on: **Aug 15, 2025.**

Course Description

This course offers an in-depth exploration of Natural Language Processing (NLP) techniques, with an emphasis on their practical applications in data science. Students will build a robust foundation in text preprocessing and analytics, progressing to advanced topics such as text classification, machine translation, and sequence-to-sequence modeling. The course also covers Large Language Models (LLMs), exploring their architecture, training methodologies, and practical applications in the real world. By the end of the course, students will be proficient in developing NLP applications, utilizing LLMs to automate tasks, generate content, and address complex language-related challenges. This course equips students with the expertise required to excel in data science and leverage NLP across diverse fields.

Schedule

The following topics will be covered during the course:

- Lecture 1: **Word Embedding**. Explore foundational techniques for representing words as vectors in a continuous space.
    1. Introduction to NLP
    2. Word embedding: Word2vec, Co-occurrence based mehtods, Glove
    3. Evaluating word vectors
    4. Word sense ambiguity

  Assignment 1: train word embedding with different algorithms.

- Lecture 2: **Language Modeling**. Understand probabilistic models for predicting and generating text sequences. Learn about Recurrent Neural Networks (RNNs) for sequential data.

    1. Introduction
    2. n-gram LMs
    3. Neural LMs and Recurrent Neural Networks (RNN)
    4. Evaluating LMs
    5. Problems with RNNs

    Quiz 1: Word embedding algorithms (loss functions, definitions) and evaluation methods. Closed-book.

- Lecture 3: **LSTM, Machine Translation, and Attention**. Learn about advanced RNN models including Long Short-Term Memory (LSTM) networks and Attention. Dive into models and methods for translating text between languages.

    1. LSTMs
    2. Machine translation
    3. Attention

    Assignment 2: build your own (small) LM with RNN, LSTM, and/or Transformer.

- Lecture 4: **Transformer**. Study the architecture that revolutionized NLP by enabling parallel processing of sequences.

    1. Impact of Transformers on NLP (and ML more broadly)
    2. Understanding the Transformer model
    3. Drawbacks and variants of Transformers

- Lecture 5: **Pretraining**. Understand pretraining methods like Bidirectional Encoder Representations from Transformers (BERT) and Generative Pretrained Transformers (GPT).

    1. Introduction
    2. Model pretraining three ways: encoders, encoder-decoders, decoders
    3. Clinical BERT
    4. What do we think pretraining is teaching?

    Assignment 3: Fine-tune a pretrained Transformer model (e.g., BERT or GPT) for a domain-specific text classification or QA task.

- Lecture 6: **Post-training**. Explore how language models are refined after pretraining to improve alignment, efficiency, and robustness.

    1. Instruction fine-tuning
    2. Reinforcement Learning from Human Feedback (RLHF)
    3. InstructGPT and ChatGPT

4. Limitation of RL and reward modeling
5. Direct Preference Optimization (DPO)
6. Human preference data; human vs. AI Feedback
7. Evaluating post-trained models

Quiz 2: Post-training concepts (RLHF, DPO, preference modeling, evaluation). Closed-book.

- Lecture 7: **Efficient Adaptation**. Explore techniques for adapting large language models efficiently without full retraining.

1. Prompting
2. Introduction to PEFT
3. Pruning / subnetwork
4. LoRA
5. Prompt tuning
6. Adapters
7. Other adaptation methods

- Lecture 8: **Knowledge Distillation**.

1. Distilling the knowledge in a neural network
2. Distilling task-specific knowledge from BERT
3. Sequence level knowledge distillation
4. Knowledge distillation of large language models
5. Distilling reasoning capabilities into smaller language models

Assignment 4: Implement knowledge distillation on a language model and compare the performance and efficiency of teacher–student setups.

- Lecture 9: **Question Answering and Knowledge**. Explore the fundamentals of QA systems, their significance, and how they leverage different knowledge sources. Discover how external knowledge retrieval can enhance text generation.

1. What's QA? Why do we care?
2. "Parametric" knowledge and why this is interesting
3. Reading comprehension and Retrieval-Augmented Generation (RAG) systems
4. RAG in EHR Modeling

- Lecture 10: **Reasoning and Agents**. Explore how language models perform reasoning and how they can be integrated into agent-based systems.

1. Reasoning in language models
2. Language model agents
3. Generative agents
4. Applications: tool use, autonomous task completion, and real-world deployments

- Lecture 11: **Multimodal and Contrastive Learning, Open Problems in 2025**.

Suggested Reading

- Lecture 1: Word Embedding.
    1. Efficient Estimation of Word Representations in Vector Space (original word2vec paper)
    2. Distributed Representations of Words and Phrases and their Compositionality (negative sampling paper)
    3. Glove: Global vectors for word representation
    4. Neural Word Embedding as Implicit Matrix Factorization
    5. Clinical Concept Embeddings Learned from Massive Sources of Multimodal Medical Data (Word embedding in healthcare)
- Lecture 2: Language Modeling.
    1. N-gram Language Models (textbook chapter)
    2. The Unreasonable Effectiveness of Recurrent Neural Networks (blog post overview)
    3. Sequence Modeling: Recurrent and Recursive Neural Nets (Sections 10.1 and 10.2)
- Lecture 3: LSTM, Machine Translation, and Attention.
    1. Learning long-term dependencies with gradient descent is difficult (one of the original vanishing gradient papers)
    2. On the difficulty of training Recurrent Neural Networks (proof of vanishing gradient problem)
    3. Attention Is All You Need
- Lecture 4: Transformer.
    1. The Illustrated Transformer
    2. Transformer (Google AI blog post)
    3. Layer Normalization
    4. Image Transformer
    5. Music Transformer: Generating music with long-term structure
    6. Jurafsky and Martin Chapter 9 (The Transformer)
- Lecture 5: Pretraining
    1. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
    2. Contextual Word Representations: A Contextual Introduction
    3. The Illustrated BERT, ELMo, and co.
    4. Jurafsky and Martin Chapter 11 (Masked Language Models)
- Lecture 6: Post-training.
    1. Aligning language models to follow instructions

2. Scaling Instruction-Finetuned Language Models
3. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback
4. How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources
5. Direct Preference Optimization: Your Language Model is Secretly a Reward Model

- Lecture 7: Efficient Adaptation

1. Language Models are Few-Shot Learners
2. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models
3. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks
4. LoRA: Low-Rank Adaptation of Large Language Models
5. Parameter-Efficient Transfer Learning for NLP

- Lecture 8: Knowledge Distillation

1. Distilling the Knowledge in a Neural Network
2. Knowledge Distillation: A Survey
3. Distilling Task-Specific Knowledge from BERT into Simple Neural Networks
4. Sequence-Level Knowledge Distillation
5. Lifelong Language Knowledge Distillation
6. Distilling Reasoning Capabilities into Smaller Language Models
7. A Survey on Knowledge Distillation of Large Language Models

- Lecture 9: Question Answering and Knowledge

1. SQuAD: 100,000+ Questions for Machine Comprehension of Text
2. Dense Passage Retrieval for Open-Domain Question Answering
3. Bidirectional Attention Flow for Machine Comprehension
4. Reading Wikipedia to Answer Open-Domain Questions
5. REALM: Retrieval-Augmented Language Model Pre-Training
6. Lost in the Middle: How Language Models Use Long Contexts
7. https://dl.acm.org/doi/10.1145/3627673.3679582

- Lecture 10: Reasoning and Agents

1. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models
2. Language Models as Agent Models
3. Self-Consistency Improves Chain of Thought Reasoning in Language Models
4. Least-to-most Prompting Enables Complex Reasoning in Large Language Models
5. Orca: Progressive Learning from Complex Explanation Traces of GPT-4

6. Measuring Faithfulness in Chain-of-Thought Reasoning
7. HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face
8. The Rise and Potential of Large Language Model Based Agents: A Survey
9. Generative Agents: Interactive Simulacra of Human Behavior
10. LLM Powered Autonomous Agents
11. ChemCrow: Augmenting large-language models with chemistry tools
12. Tina: Tiny Reasoning Models via LoRA

- Lecture 11: Multimodal and Contrastive Learning, Open Problems in 2025

1. Visual Instruction Tuning
2. UniT: multimodal multitask learning with a unified Transformer
3. Flamingo: a Visual Language Model for Few-Shot Learning
4. Learning Transferable Visual Models From Natural Language Supervision (CLIP)
5. FLAVA: A Foundational Language And Vision Alignment Model

Assessments

Assessments will include a combination of quizzes, assignments, and a final project to evaluate your understanding and application of course concepts.

Grading Policy

- Class participation **5%**:
  - Attendance and engagement: question asking and answering during the lecture
- Quizzes/Tests **20%**: two, each 10%.
- Assignments **30%**: three, each 10%.
- Project/Group Project **45%**: 10% proposal report; 25% final report; 10% presentation

Project/Group Project

The Final Project provides an opportunity to apply your newly acquired skills to an in-depth, real-world application. Students are encouraged to apply NLP techniques to publicly available data. Further details and guidance will be provided during class.