

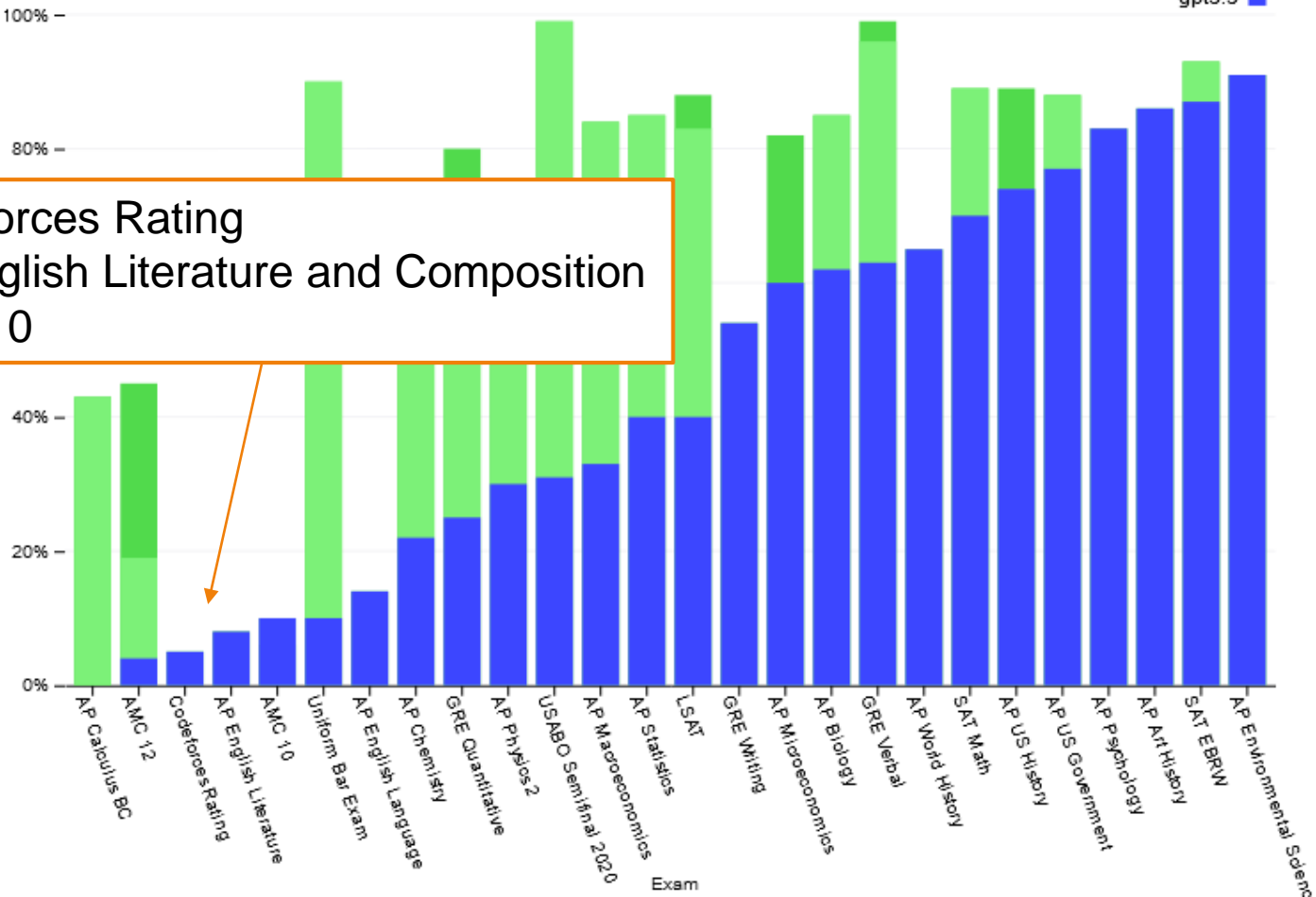
OpenAI launches GPT-4!

Exam results (ordered by GPT-3.5 performance)

Estimated percentile lower bound (among test takers)

gpt-4
gpt-4 (no vision)
gpt3.5

Codeforces Rating
AP English Literature and Composition
AMC 10



Midterm Exam

- ▶ Time

- ▶ in class (10:15-11:55am) on Mar. 21 (Tue)

- ▶ Location

- ▶ 教学中心 201
 - ▶ Seat arrangement will be announced later

- ▶ Format

- ▶ Similar to homework
 - ▶ Closed-book. You can bring **an A4-size cheat sheet + a calculator** and nothing else.
 - ▶ 可帶涂卡笔

- ▶ Grade

- ▶ 35% of the total grade





Midterm Review



Disclaimer

- ▶ Topics covered in this review may not appear in the exam.
- ▶ Topics not covered in this review may appear in the exam.



What we have covered

2. Text normalization
3. Text representation
4. Text classification
5. Text clustering
6. Language modeling
7. Sequence to sequence
8. Pretrained LM
9. Sequence labeling



Text Normalization

- ▶ Word tokenization
 - ▶ Regular expression, BPE
- ▶ Word normalization
 - ▶ Lemmatization, stemming
- ▶ Sentence segmentation
- ▶ Methods: Rules + ML



Text Representation

- ▶ Sparse vector representations
 - ▶ Co-occurrence matrices
 - ▶ Adjustments: tf-idf, PPMI
- ▶ Dense vector representations
 - ▶ Singular value decomposition
 - ▶ Latent Semantic Analysis
 - ▶ Word2vec
 - ▶ Skip-gram
- ▶ Evaluation



Text Classification

- ▶ Rule-based methods
 - ▶ Regular expression
- ▶ Machine learning methods
 - ▶ Generative classifiers
 - ▶ Naive Bayes
 - ▶ Discriminative classifiers
 - ▶ Logistic regression
- ▶ Evaluation
 - ▶ Precision, recall, F-measure
 - ▶ Macro-/micro-averaging



Text Clustering

- ▶ Mixture of Gaussian
- ▶ Unsupervised Naive Bayes
- ▶ Topic models
 - ▶ pLSA, LDA
- ▶ Learning
 - ▶ Expectation-maximization
- ▶ Evaluation



Language Modeling

- ▶ Compute the probability of a sentence
 - ▶ Chain rule: predicting the next word
 - ▶ Evaluation: perplexity
- ▶ n-gram LM
 - ▶ Probability of each word is conditioned on the preceding $n-1$ words.
- ▶ Recurrent neural networks (LSTM, GRU, +Attention)
 - ▶ Probability of each word is conditioned on a hidden vector summarizing all the preceding words
- ▶ Transformers
 - ▶ Probability of each word is computed by attending to preceding words



Sequence to Sequence

- ▶ Many applications
 - ▶ MT, paraphrase, summarization, ...
- ▶ Methods: encoder-decoder
 - ▶ Recurrent neural network (+attention)
 - ▶ Transformer: cross-attention
- ▶ Learning
 - ▶ Maximizing conditional likelihood on a parallel corpus
- ▶ Decoding
 - ▶ Greedy, beam-search
- ▶ Extensions
 - ▶ Pointer Net / Copy Mechanism
 - ▶ Seq2Set, X2Seq, Null2Seq



Pretrained LM

- ▶ Pretraining: the new paradigm
- ▶ ELMo
 - ▶ BiLSTM + LM
- ▶ BERT
 - ▶ Transformer + MLM
- ▶ GPT
 - ▶ Transformer + LM
- ▶ Utilizing PLMs
 - ▶ Finetuning
 - ▶ Prompting



Sequence Labeling

- ▶ Hidden Markov model (HMM)
 - ▶ Inference: Viterbi, Forward, Backward
 - ▶ Learning: Maximum Likelihood Estimate, Expectation-Maximization / SGD
- ▶ Conditional random field (CRF)
 - ▶ Label bias problem of MEMM
 - ▶ Inference: Viterbi, Forward, Backward
 - ▶ Learning: conditional likelihood, margin-based loss, CRF-AE
- ▶ Neural models
 - ▶ Neural softmax, neural CRF





Good Luck!

