ANLP Exam Review Guide 2023

1 Exam content and format

- The final exam will cover content from the whole course.
  - That includes materials from lectures, assigned readings, tutorial and lab exercises, homework assignments, self-assessments, and quizzes.
  - The exam normally includes some questions that are more linguistic, some that are more mathematical or computational, some that ask you to consider social or ethical implications, and some where you will need to apply what you've learned in new situations. (For more discussion of question types, see the Revision Tips below.)
  - This course covers a lot of material. If you are struggling with that, we suggest focusing on the core material that is covered in detail in the lectures, exercises, homework, and tutorials. Some lectures also include more advanced topics where we refer to the textbook for details, or where we briefly mention more recent research related to the topic. While there may be questions about this on the exam, they will typically be a very small proportion of the marks, designed to identify the very strongest students.
- Questions will be worth 50 marks in total and all questions will be marked (you don't get to choose which ones to answer).
  - The first 20 marks will be from several smaller questions, most of which are more straightforward and typically don't require more than a sentence or two to answer.
  - The final 30 marks will consist of three 10-mark, multi-part questions that may have some more open-ended parts.
  - Each question will indicate the number of marks it is worth, and partial credit may be awarded.
- Calculators and dictionaries are not permitted.
- You will write your answers on paper in a provided exam booklet.

2 Revision tips

- **Don't wait until the last minute.** Study early and often. Research shows that spaced repetition is the most effective way to remember material.
- **Study actively.** Simply reading information is not a good way to remember or understand it. Work through examples, making sure you get the same answer that's in the text or notes, and think about what questions we might ask related to that material (see below about types of questions). Also ask yourself how different parts of the course connect to

each other. What are the different themes? What are different ways to approach some of the tasks?

- **Consider the types of questions** we might ask, and how they would apply to different parts of the course. Types of questions include:
  - **Bookwork** questions, which just require you to remember facts or definitions. We rarely have many marks associated with this type of question, because simply remembering a fact doesn't demonstrate that you understand how to apply it appropriately, and in a real-life situation you would typically be able to look up facts as needed.
  - **Applying** a method, model or linguistic concept to a particular situation. For example, parsing a sentence, either by hand or using an algorithm; computing the probability of a construction under a particular probabilistic model.
  - **Synthesising** knowledge to address a new situation. For example, given a new task related to one we've discussed, discuss how you could apply or extend methods we have seen to address it, and the problems you might expect to encounter in doing so.
- **Practice the way you will be tested.** You will need to write your answers in pen on paper, in a limited amount of time. So practice doing that, using questions from past papers, tutorial exercises, etc.
- **Do a practice exam(s).**
  - It is a good idea to do (at least one) full practice exam during revision week, after you have done most of your revision.
  - See below for more information on past papers.
  - Sit down and do the practice exam in a single 2-hour sitting, writing your answers on paper, as you will during the real exam. This may help you discover issues with timing or gaps in your knowledge.

3 Past papers

Past papers are available at this link. They cover previous years starting in 2015, except for the past three years. However, there have been several updates to the course since 2014 (the biggest in 2017 and 2018), and the exam rubric has changed more than once.

- In comparison to 2014-2016, the main topics we have **eliminated** are parsing algorithms other than CKY, details of advanced smoothing methods, and nearly all topics related to discourse. We also focus much less on Pointwise Mutual Information (which used to feature in one of the homework assignments).
- The main topics we have **added** are dependency parsing and more material on distributional semantics, basic neural networks, evaluation, and social and ethical issues.
- This year, due to industrial action in week 2, we also missed topics related to finite automata, edit distance, and dynamic programming. Because it would disadvantage you to assess topics that we did not cover, these topics are **not examinable**, though we still encourage you to learn them.
- Therefore, if you are working through past papers, you should *not* expect to be able to answer the following questions:

- 2019-20: 1(a)iii, 1(e), 3(b), 3(c)
- 2018-19: 4(b)
- 2017-18: 1(a), 1(c), 2(a), 2(e), 2(f), 3(a), 3(b)
- 2016-17: 1(e), 2(b), 2(c), 3(d)
- 2015-16: 1(c), 2(b), 3(c)

You should also expect more emphasis on contextual (social and ethical) issues than in the earlier exams.

4 Exam-taking tips

- **Plan your time and use it wisely.** You've got 120 minutes to answer 50 marks worth of questions, so think ahead about how you want to use that time.
  - You'll probably want to spend a few minutes at the beginning to skim the questions and plan your approach, and a few minutes at the end to review your answers or come back to questions you did not fully answer.
  - After accounting for that, you'll probably have, on average, around 2 minutes per mark. However, some questions will be much quicker to answer than others. Since you don't need to answer questions in any particular order, you may want to start with the ones that you find quick and easy. You could also start with a partial or basic version of your answer, and leave some space to fill in more details later if you have time. (But see the next several tips.)
- **Keep track of your progress.** For example, you can tick off each question (or question part) when you've done it, or make a note if you've written something for that question but you want to come back to it if you have time.
- **Look for easy parts.** . Where a question has multiple parts, *often* the earlier parts are easier than the later parts, but not always!
  - Some multi-part questions have a "narrative" or natural ordering that doesn't align well with moving from easy to hard. So do read all parts of the question: even if you can't answer one part, you still may be able to answer the next part.
  - The number of marks on a question is only weakly correlated with its difficulty, because if all the hard questions were worth a lot, most students would do very poorly! Our goal is to achieve a spread of marks that shows how well each student has met the learning outcomes. This also means that most high-value questions have ways to achieve partial credit, and most questions where partial credit is not possible (e.g. multiple-choice questions) are not worth as much.
- **Answer the question that is asked.** Read each question carefully, and make sure you understand what it's asking, and what type of answer is needed. For example, if the question asks you to justify your answer, you will get little or no credit for simply stating the answer, even if it is correct; you will need to provide a reason.
- **Don't write more than needed.** There are at least two potential failure modes to avoid:
  - If a single word or phrase can clearly demonstrate you know the answer, don't waste time writing full sentences or repeating parts of the question.

  - If (say) a single reason or a brief explanation is enough to answer the question, don't add more just for the sake of it. And in particular, if you don't know the answer, please don't tell us everything you know about the topic of the question just in case you stumble on the right answer. Answers that include a lot of irrelevant information indicate to us that you do not have a strong grasp of the relevant concepts, so you are not doing yourself any favors and may be using up valuable time.

  In some cases, you might be unsure how much detail is needed. Most questions have hints to help you, but if you're still not sure, consider providing an initial answer with the most important information, and coming back to it later to add more detail if you have time (and if those extra details are actually relevant).

- **Do demonstrate your understanding clearly.** As noted above, a single word or phrase can sometimes be enough—but not always! Take note of any hints provided, and make sure you've expressed yourself clearly. We don't deduct marks for minor problems with spelling or grammar, but your English does need to be good enough to convince us that you know the right answer and can explain it clearly. (Note: your job is to convince us. It is not our job to guess whether you might know the right answer given what you wrote. You do not get the benefit of the doubt.)

5 ANLP 2023 examinable topics

Below is a list of concepts you should be familiar with and questions you should be able to answer if you are thoroughly familiar with the material in the course. It is safe to assume that if you have a good grasp of everything listed here, you will do well on the exam. However, we will not guarantee that only the topics mentioned here, and nothing else, will appear on the exam. In a few cases (mainly formulas) we do specify what *will not* be required for the exam, but otherwise we make no guarantees.

5.1 Generative probabilistic models

We have discussed the following probabilistic generative models:

- N-gram models
- Naive Bayes classifier
- Hidden Markov Model
- Probabilistic Context-Free Grammar

For each of these, you should be able to

- describe the generative process and write down the associated formula for the joint probability of latent and observed variables.
- compute the probability of (say) a tag-word sequence, parse tree, or whatever the model describes (assuming you know the model parameters).
- for the models with latent variables, compute the most probable [tag sequence/parse tree/class] for a particular input, hand-simulating any algorithms that might be needed (again assuming you know the model parameters).

- explain how the model is trained.
- give examples of tasks the model could be applied to, and how it would be applied.
- say what the model can and cannot capture about natural language, ideally giving examples of its failure modes.

5.2 Discriminative probabilistic models

We have discussed the following discriminative probabilistic models:

- logistic regression
- multi-layer perceptron neural networks

For these models, you should be able to:

- understand the formula for computing the conditional probability of the output class given the obervations/features, and be able to apply that formula if you are given an example problem with features and weights. We are not likely to ask you to write down the full formula yourself, but you must know it well enough to be able to answer questions like "which class is the most probable" (without us giving you the formula).
- give examples of tasks the model could be applied to, and how it would apply (e.g., what features might be useful).
- explain at a high level what training the model aims to achieve, and how it differs from training a generative model.
- explain the role of regularization, and identify situations in which it is most important.
- discuss the pros and cons of discriminative vs generative models.

6 Other formulas

In addition to the equations for the models listed above, you should know the formulas for the following concepts, what they may be used for, and be able to apply them appropriately. Where relevant you should be able to discuss strengths and weaknesses of the associated method, and alternatives.

- Bayes' Rule (also: definition of Condition Probability, law of Total Probability aka Sum Rule, and all other relevant formulas in the Basic Probability Theory reading)
- Maximum Likelihood Estimation
- Add-One / Add-Alpha Smoothing
- Interpolation (for smoothing)
- Dot product, cosine similarity
- Precision, recall, and F-measure

7 Algorithms and computational methods

For each of the following algorithms, you should be able to explain what it computes (its input and output), what it is used for, and be able to hand simulate each one.

- Byte-Pair encoding algorithm
- Viterbi algorithm
- Forward algorithm
- CKY algorithm
- Chu-Liu-Edmonds algorithm
- arc-standard transition-based parsing algorithm

For each of the following methods, we haven't discussed algorithms at the level of data structures or implemention, but you should still be able to explain what each method computes (its input and output), what it is used for, and be able to hand simulate each one.

- Parsing with semantic attachments

For each of the following methods, you should be able to explain what it computes (its input and output), what it is used for, and be able to describe how it works in some detail. You will *not* be expected to hand simulate it.

- Expectation-Maximization (forward-backward algorithm) for HMMs

8 Additional mathematical and computational concepts

Overarching concepts:

- Zipf's Law and sparse data: What is Zipf's law and what are its implications? What does "sparse data" refer to? Be able to discuss these with respect to specific tasks.
- Probability estimation and smoothing: What are different methods for estimating probabilities from corpus data, and what are the pros and cons of each, and the characteristic errors? Under what circumstances might you find simpler methods acceptable, or unacceptable? You should be familiar at a high level at least with:
  - Maximum Likelihood Estimation
  - Add-One / Add-Alpha Smoothing
  - Interpolation
  - Cross-entropy loss
  - Stochastic Gradient Descent
  - Negative sampling

  Except as noted under "Formulas" above, you do not need to memorize the formulas, but should understand the conceptual differences and motivation behind each method, and should be able to *use* the formulas if they are given to you.
- Prior and likelihood: What do these refer to (in general, and in specific models)? What is their role in a probabilistic model?

- Training, development, and test sets: How are these used and for what reason? Be able to explain their application to particular problems.

In addition, for the following concepts you should be able to explain each one, give one or two examples where appropriate, and be able to identify examples if given to you. You should be able to say what NLP tasks these are relevant to and why.

- Pointwise mutual information
- Sparse and dense vector representations of words/word embedding
- Vector-based similarity measures

9 Linguistic and representational concepts

You should be able to explain each of these concepts, give one or two examples where appropriate, and be able to identify examples if given to you. You should be able to say what NLP tasks these are relevant to and why.

- Ambiguity (of many varieties, with respect to all tasks we've discussed)
- Stems, Affixes, Root, Lemma
- Inflectional and derivational Morphology
- Part-of-Speech
- Open-class Words, Closed-class Words
- Context-Free Grammar
- Chomsky Normal Form
- Terminal and non-terminal (phrasal) categories
- Dependency syntax
- Projective and non-projective dependencies
- Lexical head words (in syntax)
- Word Senses and relations between them (synonym, hypernym, hyponym, similarity)
- Distributional hypothesis
- Meaning representations (MR)
- First Order Logic
- Verifiability
- Compositionality
- Quantifiers and quantifier scoping
- Lambda expressions
- Reification of events
- Coreference and anaphora

10 Tasks

You should be able to explain each of these tasks, give one or two examples where appropriate, and discuss cases of ambiguity or what makes the task difficult. In most cases you should be able to say what algorithm(s) or general method(s) can be used to solve the task, and what evaluation method(s) are typically used.

- Tokenization
- Language modelling
- Text categorization
- Part-of-speech tagging
- Span labeling and BIO tagging
- Syntactic parsing
- Word sense disambiguation
- Sentiment analysis
- Semantic analysis (semantic parsing)

11 Resources

You should be able to describe what linguistic information is captured in each of these resources, and how it might be used in an NLP system.

- Penn Treebank
- Universal dependencies
- WordNet

You should also be able to identify legal and ethical issues in the creation and collection of linguistic resources.

12 Evaluation concepts and methods

For each of the following, you should be able to explain what each of the specific methods measures, what tasks it would be appropriate for, and why.

- Perplexity
- Accuracy
- Precision, recall, and F-measure

In addition:

- Instrinsic vs. extrinsic evaluation: be able to explain the difference and give examples of each for particular tasks.
- Corpora: Issues involved in collection, annotation and distribution

13 Ethical issues

In addition to the topics listed under Resources and Evaluation, you should be able to identify and briefly discuss other potential ethical issues arising from developing or deploying NLP tools, and say how they might be relevant when presented with an example task. You should understand the following concepts:

- Demographic bias
- Representational vs. allocational harm
- Direct vs. indirect stakeholders
- Dual use