

Panel 1: What do we teach in NLP courses?

Panelists: [Isabelle Augenstein](#), [Emily M. Bender](#), [Yoav Goldberg](#), [Dan Jurafsky](#)

*We want to make sure that we address some of your questions to our amazing panelists. Please write your questions below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

Ways to continue this convo:

- Google Discussion group: <https://groups.google.com/g/teaching-nlp> (See past emails/messages for discussions as well)
- Participate in our interactive activities later today and tomorrow at our workshop!
 - Panel 1 followup interactive activity at 2 pm ET / 11 am PT
- DM each other and create channels for specific initiatives, activities, interests here: teachingnlpnaacl2021.slack.com
- Follow us on Twitter: [@TeachingNLP](#), tag teaching-related thoughts using [#TeachingNLP](#)
- [Yoav] How do you handle students who come with “knowledge”, as in, they heard of / used embeddings, BERT, GPT, etc etc --- but really don't know the deeper concepts.
- [Yoav Goldberg] If we were to teach “modern NLP” (ie, BERT-based, pre-training, fine-tuning.. what we see in conferences today) only, and had 12 weeks beyond the ML stuff: what would be the core concepts to teach? +3
 - [Ken Church] I'm not happy with the answer we just heard. There are too many classic topics that have been forgotten. Consider smoothing. People are talking about few-shot methods (and lots of hacks). They really should know about Good-Turing methods before inventing hacks.
 - Similarly, modern methods don't have good ways to talk about priors, and transforms. Discussions of interpretation don't know about ANOVA.
 - I'm seeing lots of people that think that all you need to know are modern methods, and you don't need to know any stats, linguistics, speech science, etc. It really helps if you know something about the language, and classic treatments. Modern methods have lots to offer, but you should know when you are reinventing old ideas (badly).
 - Some people asked for a list of classic papers: [class1.htm](#)
- [Ted Pedersen] In your own teaching of NLP, is there an example of a topic that you really liked and enjoyed teaching but that you felt like you needed to drop from a class because it was no longer current or state of the art? What topic was it, and what did you replace it with? +4
- [Emily] How do you foreground the interdisciplinarity of our field in your teaching? +3

- [Brendan O'Connor: *I have a related question below*]
- [Sowmya] How do you comment on how much of SOTA research is actually relevant in real-world, and how frequently are heuristics, regular expressions etc still useful? +6
- [Sam Bowman] What have you seen work well as way of fitting NLP into the surrounding ecosystem of classes? Can you productively combine NLP and CL into one class? Should an NLP class presuppose deep learning? What topics can you save for an 'advanced' second NLP course? +4
- [Djamé Seddah] How to make current NLP accessible to humanities students (linguistics, literature, sociology, etc..) that may lack sufficient mathematical backgrounds? +5
- [Dragomir] Has anyone tried to create a curriculum with a few videos and assignments that students can go over to prepare for an Intro to NLP class for which they are missing some of the background material?
- [Greg Durrett] Assuming you teach some topics like PCFGs or n-gram language models which are less popular at NAACL these days: do you view teaching these topics as imparting conceptual understanding (e.g., parsing is an exemplar for resolving ambiguity in language), "keeping the flame alive" (maybe GPT-6 will have a latent grammar), providing something still useful in industry, or something else? +7
 - [Dan Jurafsky]: I generally drop the topics that aren't used at all any more, except for use as scaffolding (like we all use n-grams for), but also some topics, like CKY (or beam search, etc) are still useful in neural guise, so never went away, so I try to figure out how to teach them generally!
- [Rebekah Baglini] What do you regard as the most important prerequisites for a course in NLP or CL (if different), and how do you design to accommodate students coming from different programs/fields in the same class? +4
 - [John Hale] I ask for both Intro to Linguistics and Intro to Computer Science. CS students sometimes say that they took e.g. Spanish in high school and is that enough? Linguistics students wonder if trying some simple expressions at the python prompt suffices.
- [Djamé Seddah] How to make it available on their own hardware without tying them up to currently proprietary platforms such as google collab that may or may not remain free in the forcible future. +3
 - [Steven Bedrick] This comes up *all* the time for us- I'm very curious to hear what people say
 - [Dan Jurafsky: I agree, this is a huge problem! For my undergrad class we simplify the homeworks so they are all doable on CPUs, but of course that doesn't work for the state-of-the-art stuff in the grad classes]
- [John Hale] In an *introduction* to NLP, for which topics $T_1 \dots T_n$ is it worthwhile to go inside prepackaged software components and really learn how they are built? To what degree is it acceptable to simply say "deep learning relies on backpropagating derivatives of the loss function" or "CFG parsing involves manipulating a stack" and

then move on? My sense is that unpacking those statements calls for multiple lectures, homeworks, activities and that the detailed answer certainly won't be retained a week later. Nor will it be required in the student's future given the availability of software toolkits. OTOH it imparts confidence and authority over the subject that ultimately is what I would want them to come out of college having. (note from Djamé: I was really not aiming at something too technical. My question is actually more general : how can we go further than "look people, let's say this is magical and we don't know how it works") +3

- [Dan Jurafsky: i wrestle with this too. My thought (in both the textbook and in courses), is to pick some things (SGD in logistic regression, say) or backprop, and teach them in enough depth, with a homework, that they do really feel like "I successfully implemented a neural net training", even if the part they actually implemented is quite small. So find a way to get the kernel intuition into a task. But for other things (like word2vec training, say) i lecture on it but maybe we don't make them implement it, so it's more at the level of "i saw how it worked, and i know how to use it but maybe not implement it"
- [Anna Koufakou] I have taught an intro to ML class recently with a small part in intro to text classification and NLP, with a small project, undergrad level. I see that students really get the TFIDF and NB parts and then have a real issue with keras, word embeddings etc never mind BERT and similar. What are ideas about engaging students in that level and showing a gamut of different methods without going very deep into ML and models? Our program only has one ML or NLP elective course they will ever encounter. +1
- [Dragomir Radev] Has anyone tried to create a curriculum with a few videos and assignments that students can go over to prepare for an Intro to NLP class for which they are missing some of the background material? +2
- [Brendan O'Connor] Re interdisciplinarity -- Does anyone have challenges from anti-linguistics, pro-CS sentiment among students? When I try to integrate DL + non-DL / linguistically-motivated NLP, I get severe pushback from (some) CS MS students who want to exclusively focus on DL - for example, they complain they "can't land an NLP job" without that focus. I believe Emily's view she set out at the start - we should elevate non-CS perspectives - but students may disagree! This may be symptomatic of our particular CS MS population (ling students have a diff perspective, but there are far fewer of them here currently; and CS undergrads are more open-minded) and goes to deeper questions about program design & student populations to attract or serve, the academic-business objectives behind them, etc. +3
 - [Dan Jurafsky]: I got a lot of pressure in the past to teach DL, so once all that material was in the class, I didn't get complaints any more. Just a demand for more ethics content, so now we're adding that. Students have never complained about the linguistic material that is clearly fundamental to the questions we are studying (like turn-taking in dialogue, or lexical

semantics/sense/connotation/etc when studying embeddings). I think they might complain more about syntax if I were to add it, since that's a large and complex body of knowledge that is hard to master in a short time.

- [Jason Eisner] @Isabelle -- just curious -- when you say you teach syntax, how detailed is the theory you teach? (I ask because my approach is to make the students write a PCFG of a fragment of English, but to emphasize that the variable names (i.e., nonterminal names) and code factorization (i.e., tree structure) are a matter of aesthetics and convention. So in a way, I'm being deliberately atheoretic, although through examples they certainly learn about NP, VP, PP, S, etc.)
- [Brendan O'Connor]: What challenges do you face with growth in enrollments? Or is this an incorrect assumption? (it's true here...) Can we get a data-collect a distribution of class sizes in introductory NLP classes from everyone here? Maybe this needs to be a new google form or something. size=30 vs size=150 (doesn't stanford have size>=300 now?) imply different approaches.
 - [Dan Jurafsky] We definitely have size problems. I have 250-300 in my undergrad "intro to NLP/IR/Recsys/social networks/etc" course, Chris Manning has 500 (at least) in his grad NLP, and Chris Potts has 200ish in his grad NLU course. It's a huge huge challenge, it makes it hard to have the room in the curriculum for seminars, it means students get less attention, etc. We do a lot of thinking about how to do break-out groups (pre-covid) or rooms (now) where discussions can still happen.
- [Jason Eisner] Could we / should we discuss, share, and co-develop course materials year-round? I've been at all of these TNLP workshops (one is held every few years) but AFAIK, we've never created mailing lists or resource libraries or anything to continue the discussion in between. It's sort of wasteful if we each have to make our own drawings of the same neural architectures, come up with our own examples, stress-test assignments and develop autograders, etc.
 - [Brendan O'Connor: the organizers made google group mailing list - on the webpage i think]
 - [Jason Eisner] Thanks, yes, I joined yesterday, but we might want to plan a bit; maybe there are also websites for sharing / discussing / upvoting materials (e.g., aimed at K-12 teachers, or it could be something like Pinterest)
 - [Dragomir Radev] I sent out a message to the Google Group to suggest the creation of a working group to develop shared teaching resources.
 - FYI: Drago's [LectureBank corpus repo](#).
- [Hope Schroeder]: What kind of curriculum compromises do you think are fairest to make when teaching an intro to NLP class targeted towards a particular background of non-CS, non-CL students, like digital humanities or law? Would love your opinions about whether it's worth it to push to get to DL stuff by end of course because it has exciting

applications to those fields, or whether it's irresponsible to do that at the cost of getting the basics right, especially with students with less math/stats/CS background. +1

- [Dan Jurafsky]: i think at the very least it's worth getting embeddings in there; comp social scientists and DH folks have long used topic modeling, and embeddings are sort of the next likely borrowing of methods into those fields, and I think embeddings can be taught even at a very gentle level.
- [Jason Eisner] How do you keep assignments "real" while preventing them from being "too big," especially for students who might be less experienced programmers (or less experienced with a particular required language or library)? +2
 - [Steven Bedrick] I run into this one all the time, and always have to tell students that if they are spending more time fighting with the programming part than the conceptual part, to come ask for help- otherwise the students with less programming experience spend their whole time dealing with data ingest issues, etc. Super-useful practical experience, but not the idea of the assignment...
 - [Jason Eisner] This is also the solution I've used, telling them they can get as much help as they want from anyone they want on the non-NLP stuff like file I/O. But last year I switched to giving them a partial solution so that they didn't have to write the plumbing (and to demonstrate what good code looked like). I am not sure it helped; they still have to understand the code we give them! (or at least parts of it)
 - [Steven Bedrick] Ha! That exact thing happened to us, in an earlier iteration of the course- we had what I thought were very nicely-organized scaffolds, but it turned out that most of the class had never worked with (e.g.) code organized in Python modules and classes, and so many of the students who actually needed the support from the scaffold ended up being more confused by the whole thing... I've tried to make that into a "teachable" aspect of the class, and have gotten good feedback from students over it, but I do worry that it distracts from the core algorithmic/mathematical/scientific parts of the assignment.
 - [SB] One other thing that I noticed when I didn't make the students write the plumbing was that their final projects suffered, since they didn't have as much practice. I'm consistently surprised by how far one can get in an ML education without having to do a substantial amount of programming...
 - [Dan Jurafsky] yes, i agree, we try to give them very scaffolded assignments where the part they have to do is the conceptual kernel or nugget, but they don't have to fight as much with the programming infrastructure.
- [Arnav Arora] Do you cover the history of the field in your curriculum in any way? How the field has evolved, how tasks were defined etc. Should that be covered to provide some context or for evolution of new tasks? As a student, it feels easier to find authoritative resources on newer developments compared to good old resources. +2

- [Dan Jurafsky] Not in our regular classes but I have tried once to teach a “History of NLP/Compling” seminar and I thought it went well, but I had trouble convincing students to take it!! I wonder if history of the field maybe gets more interesting as a topic as you get older and you have lived through it.
- [Ada Wan] How best to wean students from evaluating sequences based on meaning (for end-to-end sequence models)?

Resources

- Emily’s popular press assignment: http://faculty.washington.edu/ebender/2020_472/assignment3.html
- Envisioning cards: <https://www.envisioningcards.com/>
- Another idea for an ethics-related exercise that I [Emily] haven’t tried yet, but hope to next year: <https://twitter.com/emilymbender/status/1390028180304527360>
- Ethics in NLP courses: https://aclweb.org/aclwiki/Ethics_in_NLP

Oral presentations 1

*Please write your questions to the authors below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

- To Jacob: Are the books listed on the slide “Why a book” ones that you like or would recommend? [Lucy]
- To Jacob: For students a barrier to textbooks is often \$. You mentioned a free online pdf, is the link this one: <https://github.com/jacobeisenstein/gt-nlp-class/tree/master/notes>

Keynote 1: What does "real-world NLP" look like and how can students get ready for it?

Speaker: [Ines Montani](#)

*Please write your questions to Ines below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

- How do we teach students how to look for utility? Is this something you see students not realize right away?
- Would you agree that linguistics is only necessary with word-tokenized data (i.e. not for tokenization-free models)?
 - There are people working on end-to-end speech systems that think spectrograms are cool, but they don't know what they mean (and they plot them upside down).
 - >>> Surely, one needs to know and understand (how to work) their data.
 - Facts like phrase-final lengthening are a surprise to such people. There are deep connections between phrase structure and prosody, but that gets swept under the rug if you think that all you need to know is machine learning.
 - >>> I don't think ML is all one needs to know. But I think people need to be more specific when it comes to the kind of domain/data knowledge that is relevant in (information-theoretic) computation.
- [Kenneth Church] When the data are generated by Zipf's Law, then intuitions about random samples (and normal distributions) don't work so well. This is hard to explain, even to people with lots of experience.
 - [Steven Bedrick] In response to the conversation during the Q&A about focusing on the "head" of a Zipfian distribution, I wanted to point out that this focus on the head over the tail is part of how systems built using NLP (and ML more generally) can exhibit biased behavior. To continue on Ken's surname example, a system that handles the N-most-frequent surnames might achieve a certain level of performance, it by definition will underperform on less-frequent surnames, which likely would be a problem- importantly, a problem whose impacts are not evenly distributed across the user population. I love the "utility" framing that Ines used, and this is a clear case where our hypothetical system would not be delivering sufficient utility to our end users. In the "real world", do y'all find that customers are responsive to this sort of thinking? If not, any thoughts on how we might get there?
- [Jeffrey Flanigan] In industry, how do you see the value of knowing how to use tools vs knowing how to implement algorithms from scratch? +4
- [Yoav] Current NLP classes are mostly focused on, generally speaking, "how does spacy work", with hardly any "how to use spacy to do useful stuff". what would a course of the second kind look like? +1

- [?] What if you want to teach low-resource languages (like the indigenous languages of Peru or Mexico)? Many of the industry tools only work for languages with a lot of data, should we focus on making the classes with the algorithms from scratch and highlighting the linguistic aspects of each language?

Oral presentations 2

*Please write your questions to the authors below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

-

Panel 2: What does industry need?

Panelists: [Lenny Bronner](#), [Delip Rao](#), [Frank Rudzicz](#), [Rachael Tatman](#)

*We want to make sure that we address some of your questions to our amazing panelists. Please write your questions below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

- [Annie] What do you think current NLP curriculum is missing for preparing students for work in industry (i.e. as Machine learning engineer)? +1
 - Practical considerations - performance, annotating/crowdsourcing, data availability (i.e. Low-Resource languages), utility, product thinking
- [Yoav] Are there parts of the current NLP curriculum that you find to be “useless”? ie, that you will remove in order to make space for other, more important things? +4
- [Jason] It is quite clear how to define ML metrics, but how do you clearly define utility?
- [Roberto] Do 40-50 years old people who are introducing themselves in NLP, have a chance in industry? +1
- [CL] If you teach or are going to start teaching in NLP in a country with a language other than English like Spanish, the course must necessarily be taken to teach the techniques that work in English because the industry works in English? . How could you keep some NLP rules in the source language class and not lose your study too?
- [Adam] For students who are going into careers where they will be consumers of machine learning tools (e.g the journalists who Lenny builds tools for), a) is there value for them to be technically (NLP/ML) competent or aware, and b) if so, to what extent?
 - [Adam] there is a growing number of interest in coding/data science/ML from students who will not be going into “technical”/“tool building” jobs
- [Jason] It is highly challenging to build start ups, 9 out of 10 fail! What would you recommend to PhD NLP students that want to enter into the entrepreneur world for NLP products (rather than services)?
- [Fred] Wondering about interdisciplinarity at the individual vs team level. Does it make “more” sense to train individuals to span disciplines (CS & Ling inter alia), or to be able to communicate (and work) effectively with people from other disciplines?
- [Fred] It feels like there's an implicit distinction underlying this workshop (that may have been discussed yesterday, which I missed) between “NLP for Linguists” and “NLP for CS ppl”; can/should industry expect different things from these ppl and how does that translate into curricular “specializations”?

Keynote 2: Is "Introduction to NLP" just "Machine Learning: NLP Edition"?

Speaker: [Jason Eisner](#)

*Please write your questions to Jason below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

- [Anna Koufakou] +1 to the topics listed for this keynote, especially for introducing students to topics such as DL/transformers and tools such as keras and so many different libraries that are constantly changing.
- [Dragomir Radev] We have been working on a framework to share teaching materials for NLP and related topics. What do you think needs to be done to make such an initiative successful? +2
 - [Jason] My main concerns about the repo -- (1) we choose carefully before locking in an infrastructure, (2) we have a good story for why it won't get stale over time.
- [Varada] How do you usually explain bias-variance tradeoff in the context of language in an intuitive way?
- [Ted Pedersen] It seems like many of the papers/posters in this workshop, as well as the keynote by Ines Montani, and your keynote, all are talking about taking a more applied approach to teaching NLP. There are a few reasons for this mentioned, one of them being that many of our students are not planning careers in research and would simply like to use NLP to help with their problems they find in their future work or other courses, or even personal interests. However, it seems like most NLP classes don't take this approach (?). What's happening here, or what is preventing a more applied focus in teaching NLP from taking hold? +2
- [Adam] How do you fit everything discussed in the talk into a one semester class? [trick seems to be 2x lectures of a "normal" course..] +1
- [Yoav] how do "transformers" fit into your story of embedded trees --> RNNs as chain-structured trees --> [transformers??]
- [Yoav] Jason, do you believe the analogy between lambda terms and node-embeddings?
- What do you tell the students who cannot keep up with the pace and content of your course? +1
- [Ada] One can think of ML from a cognitively inspired or anthropo-centric perspective, but one does not have to. Would one be doing (some) students a disservice by assuming a worldview (your systematic thinking and analogies) that they may not subscribe to or agree with?

- [Ada] Perhaps it'd help to define "NLP" next time? I'm surprised by the diversity of definitions/interpretations. Some of the presentations assume a digital humanities perspectives, the one I am used to is more CS/CL... for example....
 - [Jason's quora post on this.](#)
- Request for Jason to share his correspondence between Machine Learning Concepts and Leading Ideas within Linguistics as well as a his nice table of Possible Ways Of Organizing an NLP class!

Resources

- <https://github.com/AngelNikoloff/Neural-Network-in-spreadsheet>
- [Jason's teaching nlp paper on competitive grammar writing](#)

Your general questions

*Please write your general questions below. You don't have to but if you feel comfortable, you can sign your question with your name. If you really like an **existing question** you could indicate it by either writing your name to it or adding +1.*

-
-