A New Balancing Act:
Reflections on the Relationship between Computational Linguistics and AI

Philip Resnik
University of Maryland
resnik@umd.edu

Keynote, 64th Annual Meeting of the Association for Computational Linguistics
San Diego, California, United States
July 5, 2026

Summary

In this talk I argued that the field of computational linguistics – a term that includes NLP as its engineering-research subdiscipline – is experiencing a "success catastrophe". The commercial success of LLM-based AI has thrown three key aspects of our research community out of balance. Here are three balancing acts we face:

> First: Like any research community, we can recognize and take advantage of the knowledge obtained in earlier generations of work; we can also lean into new approaches.
>
> Second: We can focus on language *as language*, which is to say, the properties of language that make it distinctive and human; we can also treat language as an input/output modality for AI systems.
>
> Third: We can emphasize our role as a research community, where our primary purpose is to contribute to the stock of human knowledge; we can also emphasize our role in making sure that our young members have a path forward to get jobs – particularly jobs in industry since the path into academia is never a sure bet and for many of them industry is the goal.

In each of these pairings, the central importance of the former has given way to the overwhelming dominance of the latter.

Moreover, because of the concentration of power and the convergence on methods that require enormous resources, our natural corrective processes – the historical pendulum-swings back and forth – are fundamentally broken. We have lost scientific pluralism.

These losses of balance are bad for the research enterprise, where the explosion of "let me demonstrate my skills to use LLMs and move a score" submissions has brought quality control via peer-review to its knees.

These losses of balance are bad for society, where deployment of systems without a solid understanding of their underlying theories has already caused significant harms to people.

And, I argue, these losses of balance represent an existential threat to our community. AI and LLMs are now inextricably a part of what we do. But if that's *all* we do, if we abandon our core value of contributing to human scientific and engineering knowledge about language *as language*, then we lose who we are. We become just another entry on the list of machine learning conferences.

I think there are ways to restore balance. Our workshops are full of work showing that a focus on language as language, not just text-based machine learning, is relevant for contributing essential knowledge and advancing crucial capabilities even in this new age of LLMs. Our main conferences are full of people who want jobs, yes, but who also want to make more of a difference than just improving some powerful company's valuation.

But my main job in this talk was not to put solutions on the table. It was to take what I've heard being discussed in water-cooler conversations, in social media threads, in conference hallways, and to put it into words, explicitly and out loud, in front of a few thousand people at our community's central conference. My "balancing act" framing was intended as a way to introduce a constructive common ground so that our community – and, I hope, the broader AI community – can move forward productively with conversations we really need to have.

The following is the full text of my ACL 2026 keynote, minus some ad libs added while I presented it. Minor edits and major section headings have been added. Selected slides are included where they can help with clarity.

Introduction: The "Balancing Act" Framing

Many thanks to the organizers for inviting me. I'm honored to be one of the last human keynote speakers at an ACL conference. ☺

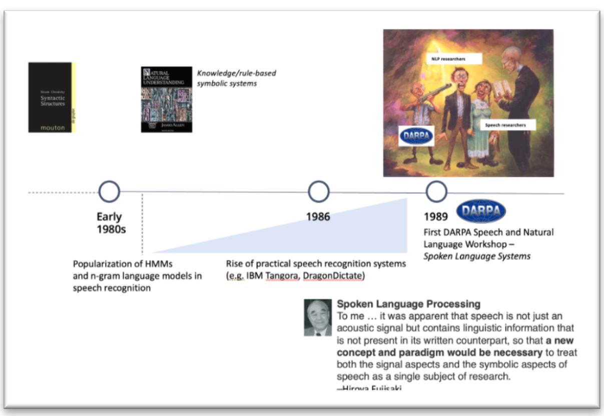
I've been around the field for quite a while.. The people who've known me a long time know that I have a tendency to open my mouth – to ask a lot of questions. A number of years ago I was giving a talk at an ACL conference, and, as I was walking up the center aisle to go up to the front, someone who knows me well looked up as I passed her, and she said "Well, at least we know you won't be asking the first question!"

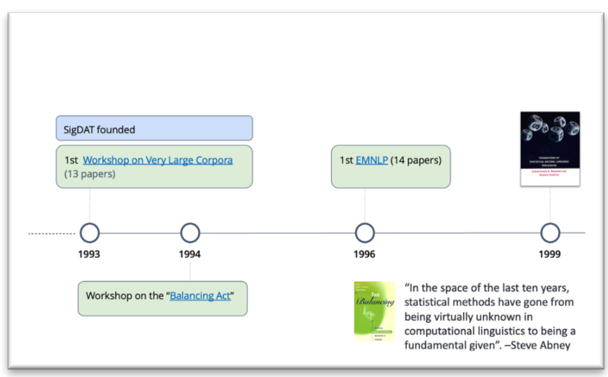
Which is fair, but I mention this because, a lot of the time when I ask questions, it's not just to get an answer, it's because I think the asking the question can add constructively to the conversation. I find myself a lot of the time asking the question or making the point that a lot of other people were already thinking. I feel like I've done something useful when people are nodding along as I ask the question, or when they come up afterwards and say, "I was thinking that", or "I'm really glad you asked that".

And that's my goal here today. I, and a lot of people, have been having a lot of side conversations about what's going on in our field in private conversations, sometimes on social media… What I'm hoping to do is bring out some of that conversation. My main goal is not for people to agree with my perspective, though of course that's always nice. It's to frame some of the discussion in a way that might be useful, whether you wind up agreeing with some or all of what I say.

Philosopher Thomas Nagel observes that nobody has a "view from nowhere" – we all have a "somewhere" that our perspective comes from. My own perspective, to be up front, is going to be somewhat US-centric, because that's the somewhere I come from.

Also informing my perspective is the fact that I do both applications oriented and scientific research – so when I say "computational linguistics", it's a cover term that includes both, it's not "NLP versus computational linguistics". This is consistent with how the term is explained on aclweb.org, our community's web site.





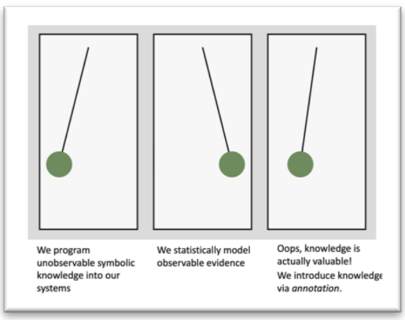
Another important part of my perspective comes from having come of age, career-wise, during our last big inflection point, the "statistical revolution". When I came into this field, it was firmly grounded in symbolic knowledge based systems. Then outside our field, in the speech community, data driven statistical modeling came on the scene, marking the rise of more and more practical speech recognition systems. Hiroya Fujisaki, in Japan, recognized and promoted the value of bringing speech research and language research together, and here in the U.S. the Defense Advanced Research Projects Agency, or DARPA, really picked that up – I think in part motivated by the fact that the speech systems were now showing a pace of progress that rule-based NLP systems weren't. And DARPA basically said, you two communities are going to start talking to each other and working together, if you want our money. And there was a lot of weeping and gnashing of teeth; this was not a happy wedding celebration.

But – it gained tremendous momentum. From a special interest group and workshop to a separate "Empirical Methods" conference and from there to textbook-level status, the "statistical revolution" redefined the field, and at the time it felt like it happened blindingly fast.

In the midst of this process, Judith Klavans and I organized a workshop we called "The Balancing Act".

When we did that, it was because clearly the field was at an inflection point, and a lot of the discussion was, "this is the right way", "no this is the right way" – it was about oppositions. And our premise was that, although well-defined boundaries can be useful, it's often way more constructive to frame things inclusively. To ask, what is valuable or useful from this perspective and that perspective – what's the right balance across multiple ways of thinking or doing things.

For us at the time, balance meant balance between the traditional symbolic work and the newer statistical approaches.



And in fact, that's largely how things panned out. For example, the pendulum was on one side, with symbolic knowledge programmed in by hand, and then there was a hard swing toward statistical modeling, and then it became clear that fully unsupervised approaches lacked knowledge we needed systems to have and we had a really productive era where you did put knowledge into systems, but you got it there through supervision – the pendulum swung back but the knowledge in people's heads got into the system via annotation, rather than trying to program it straight in.

Still another part of my own balancing background is relevant here. I'm a longtime academic, but I've also been in the industry world. I've worked in industry groups, I've had a fair amount of experience creating and advising startups. I know what it is to take a company from its founding to a 9-figure exit; to actively advise a company that went public for 10 figures. I've analyzed product-market fit; and written code that ran in production for years; I've consulted and recruited and led technical efforts and pitched VCs …And, the reason you need to know all this is that I have at least some clue about industry, and industry is a big part of the story of what's going on in our field right now.

The Success Catastrophe

And what's going on? These are exciting but also very challenging times. A lot of us, especially some of us who've been around long time, are having water cooler conversations – and sometimes more public conversations – where we're worried about the state of the field, right now and where it's going. Ehud Reiter, for example, has written really articulately about core problems that are the elephant in the room for how our field now goes about its business – more there than I can tackle individually in this talk. (But you should go read what he's written.)

By the end of this talk I'm going to argue that we're at a way more challenging inflection than the statistical revolution, that our relationship with industry is a big part of this, and I'm going to go so far as to say that I think the challenge is existential.

Of course, guys with gray in their hair have been ranting about change for a really long time – Socrates for example, criticized the introduction of writing. (The speech bubble on my slide is in ancient Greek and it says, roughly, "Hey you kids, get off my lawn!") But my point here is not *resistance* to change, it's *recognition* of change and one way to frame the discussions we're having about it.

That said, I do have a point of view you might or might not agree with. In a nutshell, I think our field, starting before the new AI industry but dramatically accelerated, is experiencing a "success catastrophe".

What does that mean? When a website is so popular it crashes, that's a success catastrophe. When someone wins the lottery and it ruins their relationships or worse; success catastrophe.

No question about industry success. What I did used to be niche; I couldn't explain to my parents what I did for a living. And then, more or less overnight, I could point to Google Translate and Siri and Alexa and say, "Those are my people"; when I say we're trying to get computers to do smarter things with language, people know what I'm talking about. Doing that is now making a lot of money. And that's having enormous effects on our field and our community.

My framing of what's going on – whether or not you agree with that premise, begins with the question of what our field is, what the community right here is. What is the ACL?

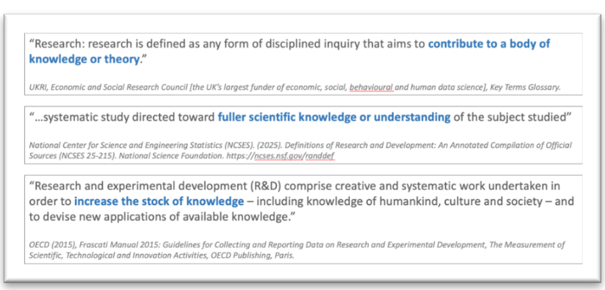
I don't think it's too contentious to say that ACL is *the* *__home community__* *for* *__R&D__* *in* *__computational linguistics__* (including and in fact mostly NLP).

That gives us three elements to structure my framing. Using those I want to talk about not a balancing act, but balancing *acts* – multiple ways in which I think the field's success has thrown us out of balance. Again, whether you agree with the "success catastrophe" part or not, this three-part framing in terms of balance is what I hope is going to be useful.

R&D and Contributions to Knowledge

Let's start with research and development.

Universally – here I've got examples from the UK, and the US, and Europe – research is definitionally about contributions to human knowledge.



"Research: research is defined as any form of disciplined inquiry that aims to **contribute to a body of knowledge or theory**."

UKRI, Economic and Social Research Council [the UK's largest funder of economic, social, behavioural and human data science], Key Terms Glossary.

"...systematic study directed toward **fuller scientific knowledge or understanding** of the subject studied"

National Center for Science and Engineering Statistics (NCSES). (2025). Definitions of Research and Development: An Annotated Compilation of Official Sources (NCSES 25-215). National Science Foundation. https://ncses.nsf.gov/randdef

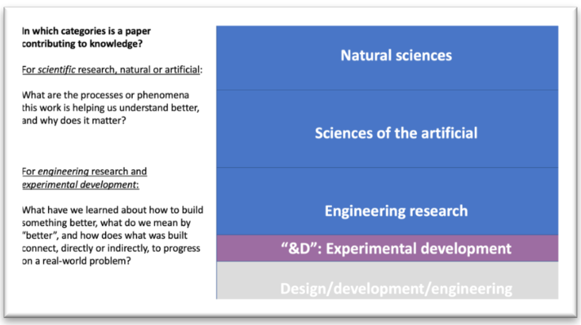
"Research and experimental development (R&D) comprise creative and systematic work undertaken in order to **increase the stock of knowledge** – including knowledge of humankind, culture and society – and to devise new applications of available knowledge."

OECD (2015), Frascati Manual 2015: Guidelines for Collecting and Reporting Data on Research and Experimental Development, The Measurement of Scientific, Technological and Innovation Activities, OECD Publishing, Paris.

That can take different forms. Herbert Simon's book *The Sciences of the Artificial* provides one really useful framework. To the well-known categories of natural sciences and engineering research, he added the concept of "sciences of the artificial", science involving human-made as opposed to natural objects of study. At the time this meant studying things like farms, or road systems, but today it applies exactly to LLMs as an object of study. If you're probing why LLMs succeed or fail at some task, if you're doing mechanistic interpretability… If your goal is scientific understanding of LLMs, you're doing science of the artificial.

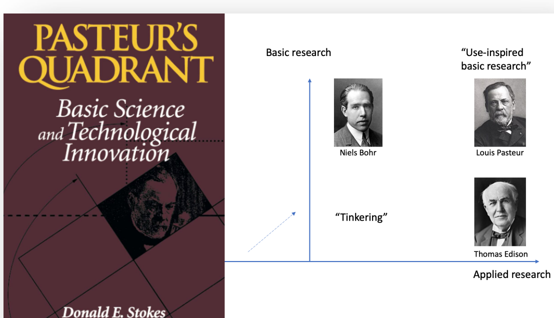
That is a different kind of contribution than engineering research. As Simon puts it, scientific research is about "what is", and engineering research is about "what ought to be" in order to serve a purpose. But crucially, *all* forms of research are about adding to human knowledge. Engineering research is not just the enterprise of engineering things to serve a purpose. It's not just about building better artifacts, it's about what we *learn* from doing that that we didn't know before. That's what makes Engineering Research different from designing and development and engineering to build a thing.

Now, having said that, we can't forget the 'and D' in 'R&D.' Historically, that's "experimental development" -- application-oriented work, and it absolutely is R&D. But trying and evaluating several approaches is neither necessary nor sufficient to make something R&D. The key question is *what* you're trying and *why*: it has to be aimed at *doing* something new in order to learn something new. The boundaries aren't perfectly crisp, but that's different from just trying variations on a theme to see which works best for a business need.



What all this means is that when we review papers, we should be asking, what is the nature of the contribution? For scientific contributions, we should be asking what are the processes or phenomena this work is helping us understand better, and why does it matter? For engineering research and experimental development: What have we learned about how to build something better, what do we mean by "better", and how does what was built connect, directly or indirectly, to a solving real-world problem?

Even if a paper doesn't fit neatly in these categories, what have we added to our knowledge, and why do we care? That's what it means to have a research result.



As another useful framework, Donald Stokes argued that the distinction between "basic" and "applied" research is too uni-dimensional, and his clever solution was to fold up the left half of that two-way arrow to make things two-dimensional. So you can distinguish not just between Niels Bohr doing fundamental physics, and Thomas Edison's deeply practical experimentation, but also someone like Pasteur who was motivated by both.

This leaves you with a fourth quadrant that's often labeled as "tinkering". Tinkering is fundamentally empirical; it doesn't have the goal of expanding scientific understanding. At the same time, tinkering isn't fundamentally use inspired – you're trying things out, seeing what happens and maybe even measuring it, but without trying hard to connect to the real-world use cases.

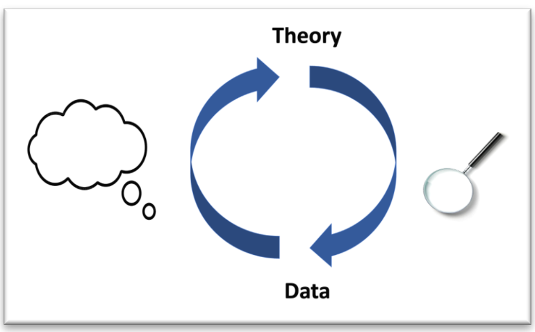
Computational Linguistics/NLP, Theory, and "Language as Language"

Which brings us to thinking about what we mean by "computational linguistics".

If you look at how our field defines itself, this community is clearly defined in terms of R&D goals – what we are about is *not* the tinkering quadrant in Stokes's picture. Even the application-motivated pursuit of knowledge is a motivation in pursuit of knowledge.

I think that's important, because it highlights that as a research community, a central value we have involves not just doing but understanding. I'm not talking about "language understanding" by systems, I'm talking about *human* understanding – interpretable human knowledge.

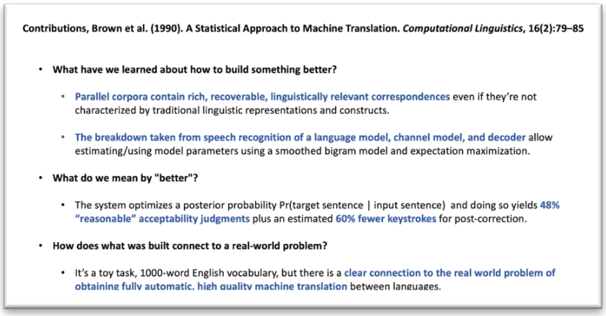
Now, there are claims that have been made that theory, in the sense of human-interpretable explanations, is just not needed anymore, because having enough data makes theory unnecessary.

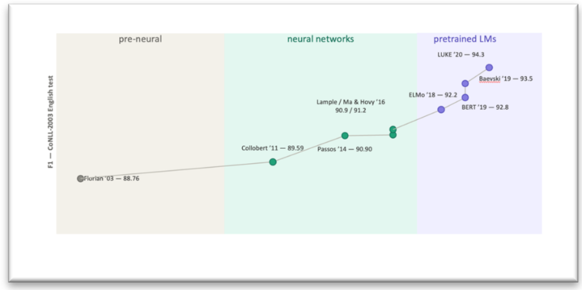


I want to argue that that's not only mistaken, but it's actually not possible. I was once teaching a seminar and I said something about wanting to stay away from philosophy; and my colleague Norbert Hornstein, who was sitting in on the seminar, looked me straight in the eye, and he said: "You can do philosophy with your eyes open, or you can do philosophy with your eyes closed". And the same is true of theory. There is no such thing as "data" in the absence of a theory, whether you're *acknowledging* the theory or not. Your theory – your current understanding of whatever generated the observations available in the world – that theory is your lens on the world; it determines which observations you're going to consider data, what you're going to measure about them, what questions you think those measurements are going to help you answer. If you are asking questions or building things based on observations of human language use, whether it's a psycholinguistics experiment or a huge corpus you collected to train on, then you are employing a theory of language.

Take something like the bigram model, the kind of thing people sometimes call "purely statistical". It's nothing but numbers, conditional probabilities, right? But every probability model has an underlying algebraic structure that's not about numbers; for bigram models, it's theory of language based on finite state automata. And that architecture is *every bit* as much a theory of language as what Chomsky and his students and all their students have been producing in Linguistics departments over the past 70 years.

Now let's be clear: I'm not advocating Chomsky's brand of theory. Some of it makes sense – especially the care with which he asks, "What exactly are we studying, when we say we're studying language", whether or not you agree with his answer. Other aspects don't. And sometimes a radically different theory of language really works better for a purpose you care about, even if it's a huge departure from our previous understanding. When the IBM group introduced statistical machine translation, it was also, a lot like Chomsky's, a generative model, but grounded in a completely different way of thinking about a generative process, and in practical terms it worked better.





And notice, just working better was not what made it an important research contribution. The paper established the power of parallel corpora and applying the noisy channel approach from speech recognition, and had measurable results that clearly connected with the real-world problem of language translation.

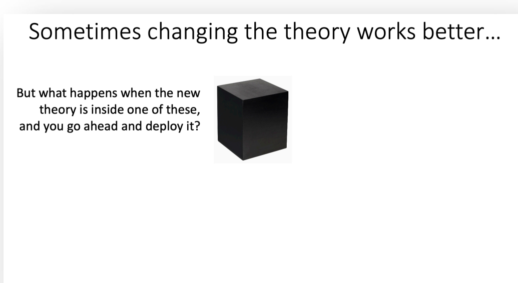
Changes to the underlying theory – and yes, I'm calling this progression a progression of theories -- have served us really well in terms of measuring performance on things we care about, like named entity recognition, where you can see the clear progression from pre-neural, to neural network, to LLM based approaches.

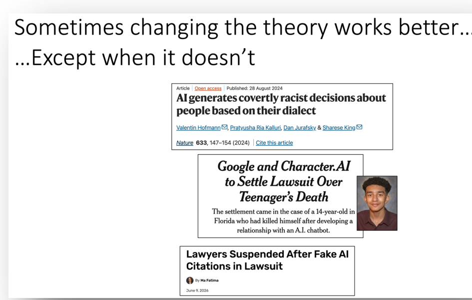
And now, changing the theory enables us to build systems that do things many of us could never have imagined doing in our lifetimes. Capabilities I definitely took advantage of when creating this talk.

Unexamined Theory and its Consequences

But... What happens when your theory lives inside a black box and you go ahead and deploy it? That takes us back to the discussion of tinkering. Our stated values as a community put us outside the tinkering quadrant, but I would argue that where we are right now in practice is actually at the cusp of pre-scientific and scientific research, tinkering and systematic investigation. We're putting things out into the world that are created without much of an an explicit or explanatory theory, and also were not developed through a systematic process driven by carefully thought out use cases. Now, why does that matter? It matters because if you're relying on a theory that's unexamined or unexaminable, what you're doing can blow up in your face.

Unacknowledged or unexamined theory has consequences.





Work by Hofmann and colleagues on racial biases in models, for example, shows that the models have theorized a relationship between a speaker's dialect and traits or characteristics of the speaker, that nobody explicitly specified, and a generic, theory-independent fix like RLHF doesn't fix it – and yet there *is* relevant theory out there about the relationship between linguistic variation and the interpretation of speaker characteristics.

The Sewell Setzer tragedy involves something we've known since ELIZA, which is that systems are capable of creating user engagement without any theory about the relationship between the nature of that engagement and the actual state of the individual they're engaging. Without any pragmatic theory – the kind of theory that talks about the internal states of the participants in a conversation – you get situations like this where the human in the conversation is basically being invited to supply their own theory of their artificial partner. And my honestly ungenerous reading is that a lot of the time, like in the case of "chatbot companions" or mental health chatbots, that's exactly the inference the system is designed to encourage.

And as a third example, if you look at the issue of hallucinations, these systems are built without any formal distinction between what's *true* and what's merely *plausible*. This connects to an argument I've made recently that harmful biases in LLMs are unavoidable because of their purely, radically distributional assumptions about meaning, one of the few things we *can* state clearly about their theoretical grounding.

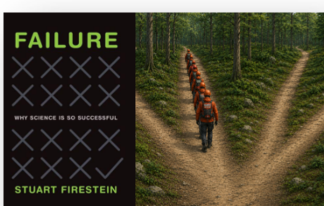
All three of these cases are showing there are real costs to a failure to examine what theory is or isn't a part of the systems we're building.

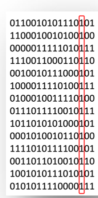
And, here's the thing. We have decades upon decades of work where really serious, careful scientific effort *has* gone into understanding the nature of language as something beyond an input/output modality for machine learning. The properties of linguistic interaction; the differences between linguistic knowledge and world knowledge; between factuality and plausibility. Did the work in previous traditions produce anything resembling the kinds of systems we've got now? Hell no. Was Chomsky's brand of linguistic theory, or good old-fashioned AI, the right foundation for practical systems? No. But even before the statistical revolution there was more to the explanatory side of computational linguistics than just Chomsky, or just old fashioned AI. A lot was learned about the nature of human language, and properties of human beings in relation to language.

Yet, out of 20 different fields, Wahle et al. (2025) found that NLP reaches the least far back for past knowledge -- a decline that started right when LSTMs and then transformers started taking over everyone's attention and, I would add, right when you started seeing some major commercial impact. A lot of that past *knowledge* was real even if the *methods* failed in practical terms. Ignoring that knowledge has consequences. I'm not talking about clutching onto the past here, I'm talking about not throwing out the baby with the bathwater.

What's a Research Community For?

And finally that brings us to the idea of a community. Our community has many goals, but I think two rise above the others. The first is to advance knowledge in computational linguistics -- what I've mainly been talking about. The other is to take care of the people in the community, particularly the young people who are coming up in the field. If we are not placing the people we're training in jobs that meet their needs and their goals, then we're not doing our job as a community.

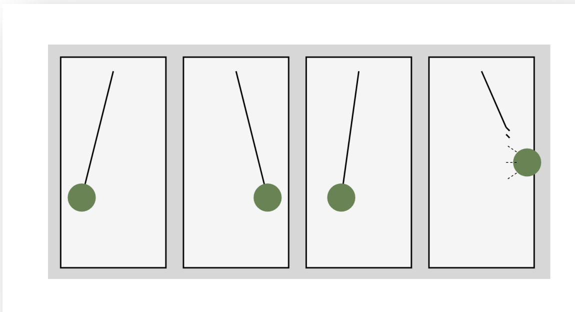




Stuart Firestein argues for the the importance of scientific pluralism in a community -- if you're looking for a lost camper in the forest, you don't send all the rescuers down the same path, even though you know some paths will fail.

As perhaps a better analogy, I did some work with genetic algorithms back in the day, and it informed how I think about exploration and exploitation. If you've got a population – artificial or real – that has completely converged on the value for a gene, then you're done with exploration for that value, unless you happen with tiny probability to hit a lucky mutation. So, if that gene needs a different value for better solutions in your search space, you've converged prematurely and you're just plain out of luck.

And that's where we are right now. How many of you [show of hands] are developing a *competitor* to Transformer-based language models? [One or two in a room full of thousands of people.] Right – because you're not Yan LeCun, and you don't have a billion dollars, and you would like to get *published* and you would like to get *funding* and you would like to get a *job*.



The pendulum is broken.

And that affects the community. For the same paper, if it weren't about the timing or the location, nobody cares if they're submitting a paper to a conference with "ACL" in the name or to EMNLP. And that makes perfect sense. When statistical NLP came on the scene it was outside the mainstream, and it needed a special interest group and then a conference of its own. But ACL itself evolved over time and today only the conference title is different.

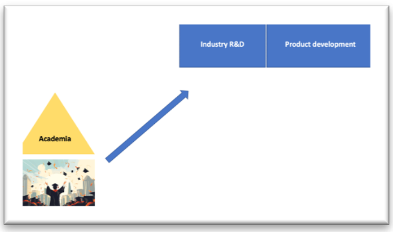
My worry is, that we're now approaching the same situation with ACL and the machine learning conferences like NeurIPS. If "language" just means an input/output modality for machine learning, then don't we have just another name-only difference, like "ACL" and "EMNLP"? I've heard the worry, stated out loud, that if we're not careful we could turn ourselves into, quote, "a second-tier machine learning conference".

But consider an alternative. We have for a long time been the primary research community with a central focus on computational approaches to language *as language*. You can see how we differentiate from NeurIPS and the rest in the workshops, where for us, yes, we're using machine learning and LLMs, but the fact that it's *language* often matters. Where we talk about argumentation and discourse and culture and multilinguality and reference and terminology and meaning and language development...

Caring about language as language *differentiates* us as a community. And without that – or some other differentiator – my concern is that we will cease to exist, as a community, the way that EMNLP and ACL ceased to be independent communities.

Industry Jobs, and the Paper-Submissions Crisis

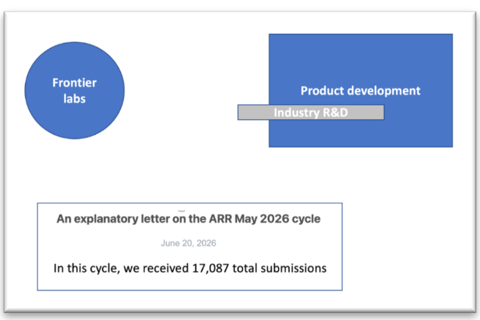
Yes, but on the other side of the balance, not just research value but taking care of people in our field, isn't the LLM juggernaut where the jobs are? Let's look at that.



Getting a job in academia has always been challenging. But historically, when you finished a PhD., whether by preference or necessity there were other options for using the skills and knowledge from your PhD, including a pretty robust set of industry labs – where R&D was in a business context, but it still was R&D. Knowledge discovery. Publications.

But that kind of corporate R&D has been thinning out for a long time, and getting more and more product driven, even in R&D mainstays like Microsoft, FAIR, DeepMind.

The dominant knowledge-seeking is now in a tiny number of frontier labs, all focused on variations of the same approach, and largely driven by an arms race where your corporate valuation depends on higher and higher benchmark scores, public perception, and people using your product. In the process, the industry open-publication approach is being dismantled: traditional research roles getting shifted toward product, or cut; research is being walled off with publication embargoes or layers of business approval; and when things do come out, the value of industry-watchers seeing it on ArXiv is much more immediate than the value of scientific quality control.



In that environment, what's a student to do?

On the frontier labs side, there *is* genuinely an appetite for people who can create new knowledge -- within the boundaries of LLM approaches – with awesome salaries… but the actual number of jobs is small compared to the number of new PhDs, and a lot of that is demand for research *engineers*, where showing you've improved something you can measure is an essential part of the job. Otherwise "industry" is about, well, what industry has always been about: satisfying business needs. A lot of which involves improving "key performance indicators" of some kind.

So the bulk of the jobs out there prioritize taking a problem someone else defined and showing that you can improve the performance metrics regularly – say, oh, 2, 3, 4 times a year.

Does that sound familiar? Is it any wonder that students feel pressured to publish methods-focused metric-improving papers 2, 3, 4 times a year?

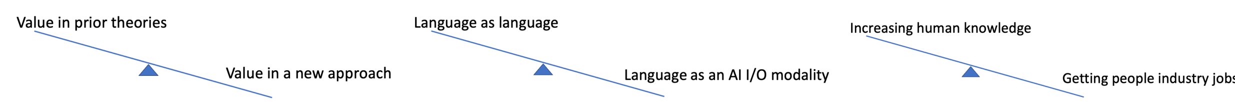
Where the bulk of the jobs live, the shape of the industry is telling students that they need to be submitting and submitting papers, and that what's valued in those papers is not the research question, it's showing someone they have the skills to use methods to move a metric, on *something*.

And the irony is, that, whether or not I'm right about the reason students feel this pressure, they do, and the volume of submissions this is producing is actually working *against* those same people. Because with more and more submissions, the reviewing is stretched thinner and thinner: you have less qualified or poorly matched reviewers, people with less time to do a good job of it, and a sense among a lot of people that the signal is being lost: with this much noise, selectivity in terms of percentages may no longer be the signal of high quality that it once was. And that isn't even talking about the problem of dishonest submissions using AI, which makes the volume of submissions even worse.

What we have is not a reviewing problem. It's submissions problem where the reviewing problem is the main symptom.

Conclusion

So that's my framing: three things where our balance is out of whack.



> We've lost the balance between extracting value from a new theory, and retaining what was valuable in prior theories. A lot like the 1990s revolution, except with a much more dramatic loss of scientific pluralism.
>
> We've lost the balance between language as an input/output modality, and language as the central object of study for our field, which is what makes our computational discipline different from the others.
>
> We've lost the balance between our obligation as a community to prepare our students for the jobs that are out there, and our purpose as a research community that's not driven *just* by a powerful industry's agenda.

Like a lot of people, I have thoughts on where to go from here, but laying out this framing was my main goal.

Hopefully articulating things in terms of balance, or balances, might be useful in figuring out the path forward. Thank you, and I hope a lot of you will be a part of the conversation.

Thanks!

Many thanks for feedback and comments from Kfir Bar, Kai Golan Hashiloni, Alexander Hoyle, Maria Liakata, Hilary Owusu, Shramay Palta,, Rebecca Resnik, Rupak Sarkar, Joel Tetrault, and Aya Zirikly. All errors and opinions are entirely mine.