Podcast Vibes

Analyzing emotions in podcasts

Note: This was a presentation given to Elena Glassman's group in the Harvard CS department on Thursday, February 27, 2025. Thanks to everyone for their attention and feedback!

This version was auto-generated from the slides by iA Presenter and while it doesn't really work as a standalone blog post I figured I'd put it online as an annotated presentation, in the spirit of the other messy first drafts on my personal site.

Most of the text below the slides written after the presentation was complete and I've tried to keep it reasonably close to what was discussed.

Feel free to skim.

Hi everyone, it's great to be here. Elena and I were chatting recently and she graciously invited me to come and share a little side project I've been working on that you might find interesting.

Please stop me at any time if you want to comment or ask anything.

About me

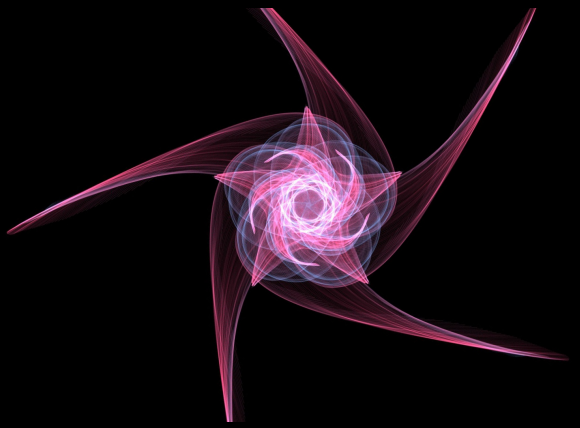
My background is in computer science and linguistics, and professionally I've spent my career working at startups and as an independent data visualization consultant for companies of all sizes.

My work integrates full-stack engineering, data analysis, and design, with the aim of helping people understand and control complex systems.

I'm also the co-founder of an infrastructure observability startup where we use fine-grained telemetry from the Linux kernel to help people make their computer programs more efficient.

In my free time I work on projects a bit more towards the humanities side of things -- like making interactive generative art tools, and visualizing dog names.


mpkin Spice Latte
TOP 250K BY VIEWS
views last month
Pumpkin Spice Latte (PSL) is a coffee drink made with a
of traditional fall spice flavors, steamed milk, espresso,
often sugar, topped with whipped cream and pumpkin ...
VIEW ON WIKIPEDIA »
thly trend
October, 32k views
2022
2023
2024
kly pulse
MON
TUE
WED
THU
FRI
SAT
1×
1×
1×
1×
1×
0.9×
Yearly seasonality
DEC
JAN
FEB
NOV
OCT
SEP
MAR
APR
AUG
JUL
JUN
MAY
2025
2024
2023
2022
1k
568
15 views



Two parts

1. A story from the past

2. The podcast project

This talk is made up of two parts.

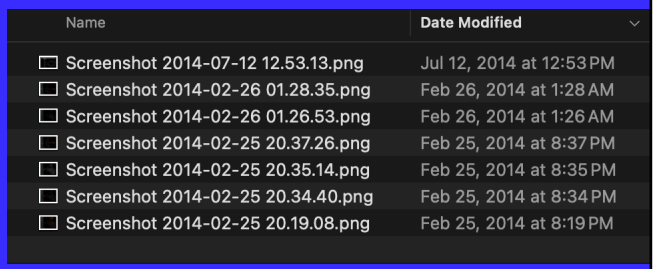
First, I'd like to tell the story of some work I did over a decade ago, from 2012 to 2015, exploring the use of vector space representations of text to build tools for understanding "large" (ish) collections of text documents (eg. 10,000 to 100,000 product reviews).

Then I'll talk about a little side project I've been tinkering with lately which is a bit of a spiritual successor to some of that work.

Once upon a time...

In a land far far away...

(Cambridge)



This work was done in collaboration with the rest of my team at a startup I worked at over a decade ago, and in particular in close collaboration with Elia Robyn Lake, Jason Alonso, Ken Arnold, Avril Kenney, Christina Laverentz, Alice Kaanta, and Andrew Lin. We were all early employees (or cofounders) at an AI company called Luminoso that spun out of the MIT Media Lab.

The company had its origins before the deep learning revolution and was the commercialization of research into vector-based text representations that combined background knowledge together with the knowledge from a body of documents.

The background knowledge came from a graph-based knowledge representation, ConceptNet, which was created and maintained by members of the founding team. We had a proprietary data pipeline that would "blend" this background knowledge together with conceptual associations derived from the user's document set.

How can you make sense of 10,000 product reviews?

Our primary commercial focus was trying to help companies understand their users better.

In particular, while there were well-established techniques for analyzing survey response data that was numeric, very little was known in the industry about how to deal with freeform text.

This text, which sometimes contains the most valuable feedback because it can answer questions you didn't think to ask, would often be discarded due to the lack of good techniques and tools.

Our answer:

The Concept Cloud

In addition to our vector-based models, another innovation was on the presentation side, and was something we called a Concept Cloud.

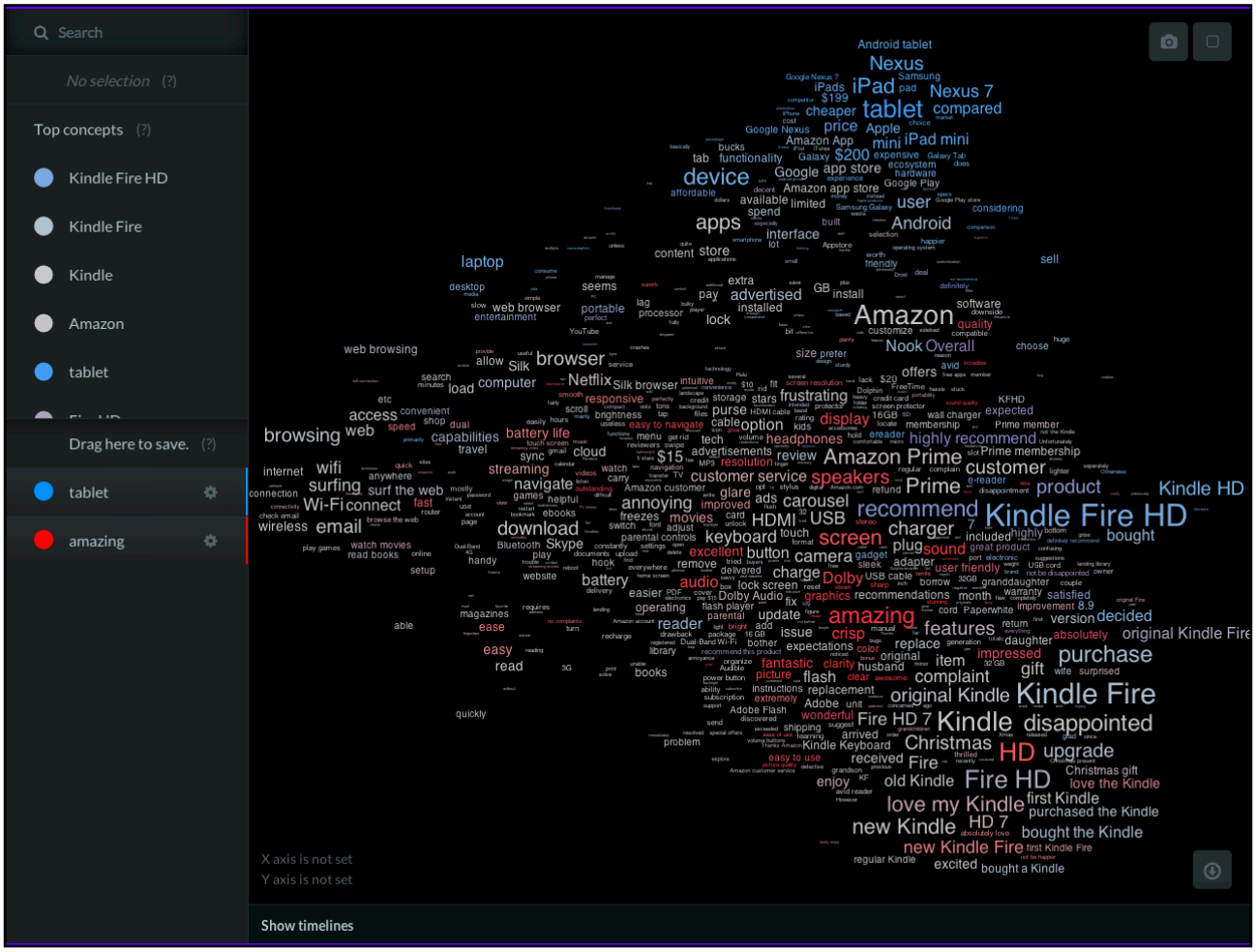
Word clouds are famously considered to be poor data visualizations since only the size of words is meaningful, leaving other data channels, such as the positions and colors of the words, as free parameters chosen somewhat arbitrarily. This makes word clouds easy to misunderstand since there is little signal and a lot of noise.

But what if you make all of those other channels meaningful?

That's what the Concept Cloud did. We called it a *Concept Cloud*, since the visualized elements were the high-dimensional concept vectors associated with words and phrases rather than the words themselves.

By default we would lay the cloud out based on a dimensionality reduction to 2D, and would use color dynamically to indicate associations of individual concepts with user-selected topics.

I have a patent together with Elia Robyn Lake for part of this idea.



Here's what the user interface looked like.

The cloud shows the most relevant concepts, arranged with a UMAP-like technique and colored by user-selected topics, in this case "Tablet" and "Amazing".

The data here is roughly 10,000 product reviews of the Amazon Kindle Fire tablet, and you can see the "Tablet" cluster up top, with words like "iPad" and a "Device" colored blue, and words related to "Amazing" colored red, such as "Fantastic", "Speakers", and "Screen".

Unlike today's vector embeddings, which capture background knowledge only, ours captured conceptual associations in terms of both the common-sense meanings of words -- eg. "Tablet" is related to "iPad" and "Device" -- as well as document-based associations, such as the relationship between "Amazing" and "Speakers" (since this device had really good audio quality).

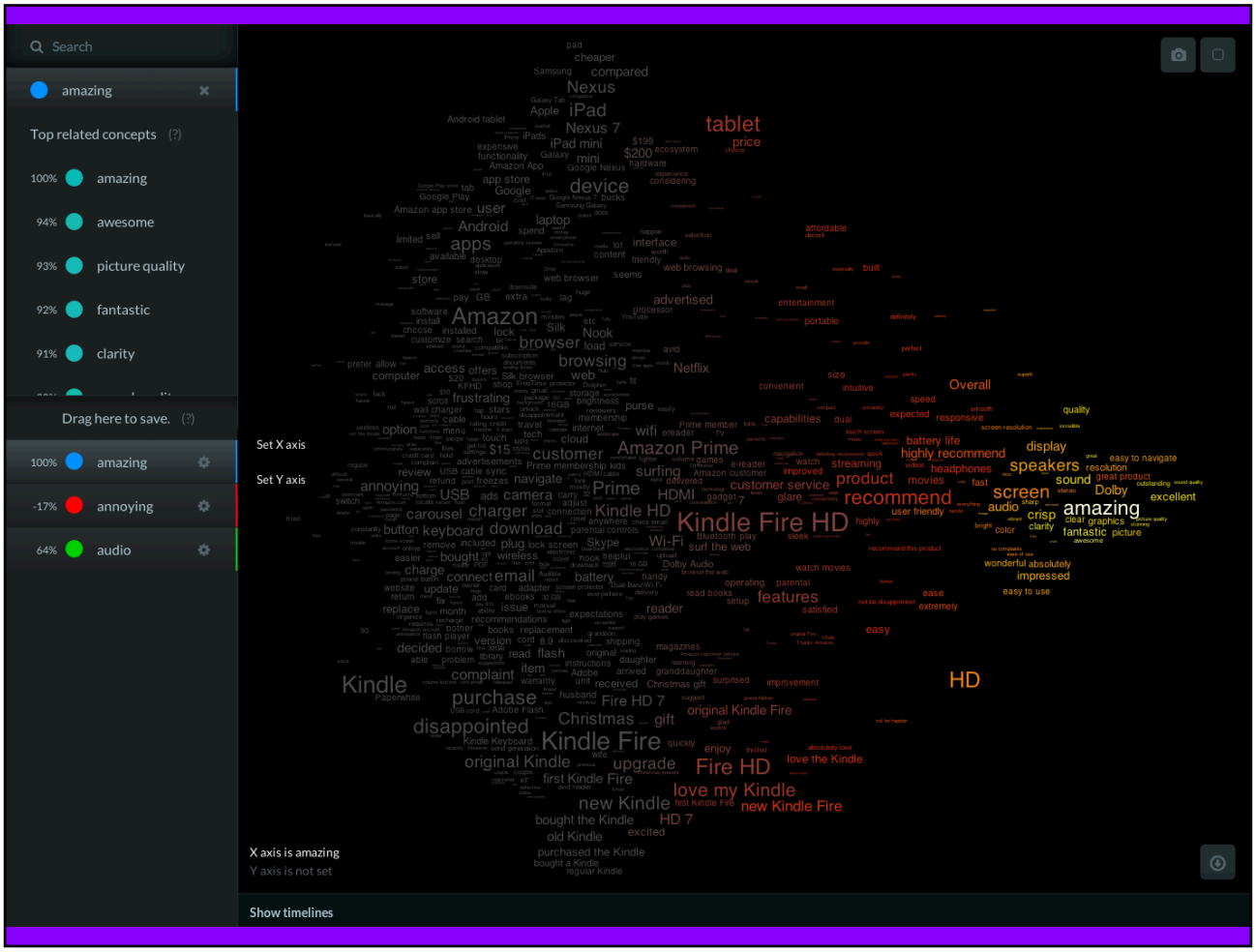
If you click on a word, the interface would highlight related words. For example, clicking on "Browsing" will fade out all of the concepts that are unrelated to it. The highlighted words are those that relate to browsing in some way, such as "Surfing" and "Wi-Fi".

If you click on "Audio" instead, then you see that the most related words are things like "Speakers" and "Sound" and "Amazing" again.

This is because in the collection of product reviews we're looking at, the speakers are, in fact, amazing.

Unlike the other clusters we've seen so far, the related words for "Audio" are pretty scattered. This is because the two-dimensional layout cannot possibly represent all relationships accurately in a two-dimensional space.

It can be really useful to organize, for example, all of the "Amazing"-related concepts together. So for this, we had a feature that allowed you to set a particular topic as an axis. For example, let's make the *X* axis be "Amazing".



Here are the same words as before but rearranged so that all of the words with a high relatedness to "Amazing" are on the right.

Since that word is also selected, you can see the same relationship in the color gradient too.

The other thing to notice here are the numerical scores on the left among the user-selected topics. We'll come back to those later.

These numbers tell you that "Amazing" is 100% related to itself, 64% related to "Audio", and -17% related to "Annoying". (Under the hood, these were the dot product relationships between the vectors for those terms.)

Looks cool, but...

Our clients thought that this cloud looked amazing and really enjoyed playing with it. They would nod sagely when we told them that "Audio" is 64% related to "Amazing", and would walk away feeling wise.

I was not so easily satisfied, and eventually realized that this was actually a deeply unproductive way to present our information, even though it contains some seeds of greatness.

I was fairly inexperienced at this kind of work in 2013, and it took me a while to understand the issues that I'm about to explain.

So what's wrong with the interface? Well one thing that's wrong are these numbers. When we say that "Audio" is 64% related to "Amazing", what on earth does that actually mean? That number is not interpretable, and in fact even the people at the company did not have a clear answer as to what that number means, beyond "dot product between normalized high-dimensional concept vectors".

In terms of interpretability, the number itself had issues too -- our vectors conflated the background knowledge with the domain knowledge in a way that we could not subsequently unweave.

This makes it hard to know whether a high relatedness score is due to the user's data or the background knowledge.

similar issues still come up today with both LLM-based analysis where the model can interpret data unexpectedly, and with vector embeddings, whose vectors can have non-obvious distance relationships.

What's wrong with it?

A few things...

It's *ungrounded* – Only derived data is shown, not the underlying documents

Another problem is that our user interface shows only derived data. The original documents are nowhere to be seen, making it hard or impossible for the user to question the model.

While we did eventually add a documents sidebar there were other invisible modeling choices, such as deciding which words to include in the concept cloud and how to size them, which made it hard or impossible to interpret our visualizations with confidence, and could significantly impact the results of data analyses.

What's wrong with it?

A few things...

It's *fuzzy* – Hard to draw firm conclusions on the basis of ill-defined vector dot products

Another issue is the fact that we were directly reporting "internal" model parameters to the user.

These numbers, representing associations between concepts, can be extremely useful as input features to a machine learning algorithm.

But they are not good tools for human thought -- in my experience, they are in practice impossible to use correctly for understanding or decision-making, even if it is theoretically possible.

Decision are discrete -- which product development should be prioritized for the next iteration of our product? -- but our reporting could not be easily used to confidently justify those conclusions since there was no way to sharpen the scores into something closer to a higher-level decision within the system.

This applied to the scores we reported to the user, the color highlighting, and the positions along the x-axis or in the default layout – everything is vaguely meaningful but imprecise and bore a complex relationship to the ground truth documents underlying the analysis.

So, what to do?

Possible solution: *tags*

My idea was to use tagging, which can be seen as a form of binary classification, as a firmer basis for interpretable data analysis.

By tags I mean the same kind of thing as hashtags or tags on old-school blogs.

Tagging

- Tag documents as "amazing" if they express amazement, or "audio quality", "annoying", ...

  Crucially, each tag represents a binary decision!

The idea was that our system would help the user interactively create curated document sets through a tagging process, and would then explore the relationships between tags as a more grounded form of analysis.

So if you have a bunch of product reviews, you could, for example, use our system to tag a subset of them as related to "Audio". Or perhaps for some tags the appropriate unit of analysis would be a paragraph or sentence, allowing more targeted analysis.

Benefits

1. This is more grounded & interpretable
2. This is more concrete – a product review either has the tag or it doesn't.
3. This forces useful sharpening – what *should* it mean to tag a review as "awesome"?
4. Tags represent sets of documents. (Eg. You can explore tag-tag relationships by intersecting their document sets!)

So here are a bunch of the advantages of this kind of approach.

Compared to vector-based metrics ("A is 64% related to B"), tag-based metrics ("20% of the documents are tagged C") are more interpretable since you can sample or inspect specific documents with that tag to verify the conclusion yourself by sampling documents and spot-checking them.

And once you've tagged your documents, you can start exploring the relationships between them in a very concrete way since set operations (intersection, difference, segmentation) over tags become meaningful.

To me, '200 reviews are tagged both "Audio" and "Amazing"' is much more interpretable than '"Audio" and "Amazing" are 64% associated', but more on this on the next slide - this is also misleading! oh, 2014...

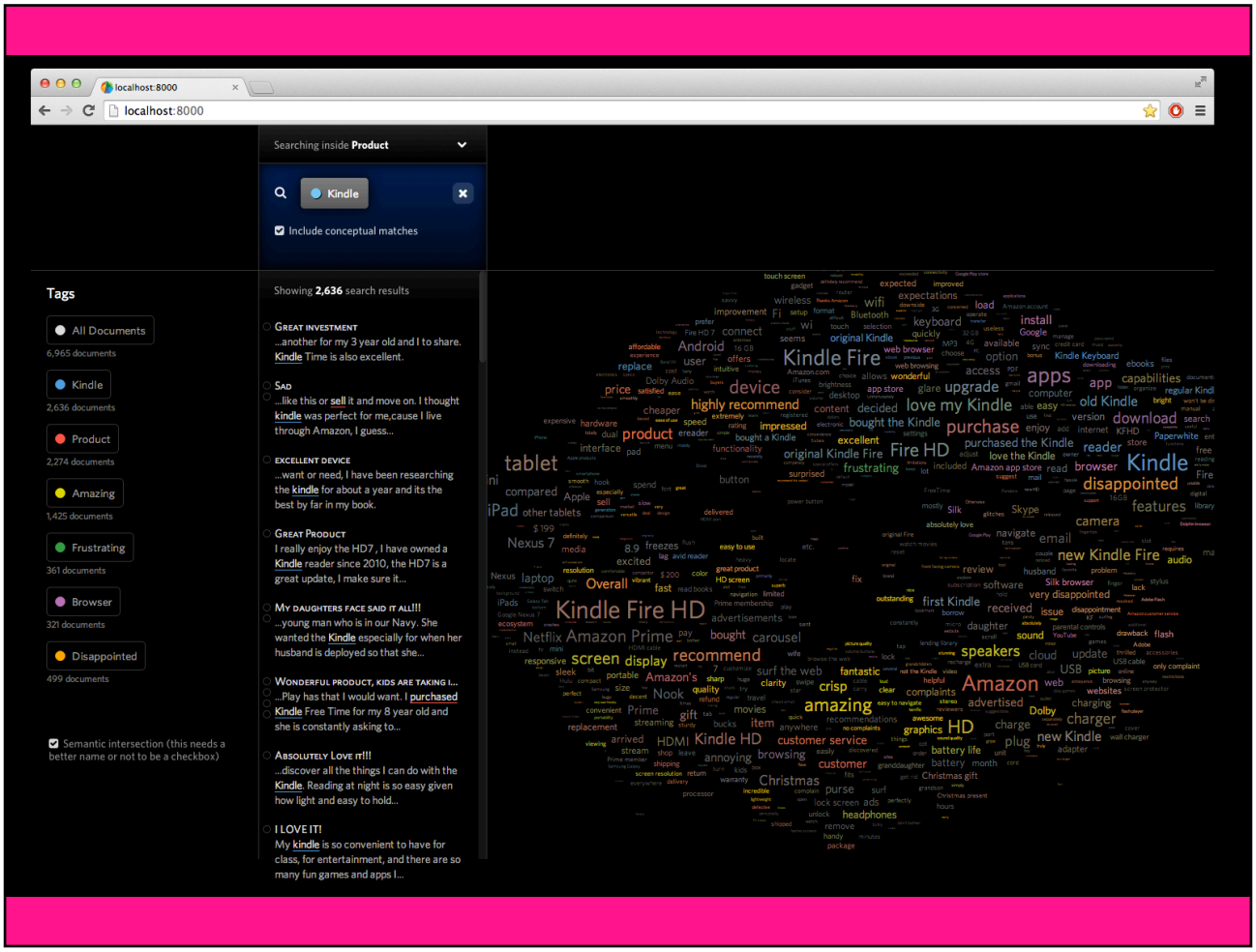
Challenges

- *Compositionality* – Analyzing two tags through their intersection can be fraught.
  - For example, reviews tagged "Amazing" and "Audio quality" do not necessarily say good things about the audio quality.
- *Tag Quality* – Auto-tagging at low cost & latency is possibly still an unsolved research problem!
  - Same for collaborative human/ML tagging workflows

However, this still has a lot of issues. One issue is that the precise meaning of tags becomes very important, and composing tags together can require care and attention to the granularity at which the tags are applied.

For example, reviews tagged "Amazing" and "Audio quality" do not necessarily say good things about the audio quality.

And in 2014, the quality of tags was also a big problem for us because it was difficult to dynamically and flexibly apply user-specified tags without a lot of erroneous classifications. This is part of the reason why this prototype never made it out to production, though I still think it's a good idea if it can be made to work.



So at the time, this is what my prototype of a solution looked like.

It has a list of tags on the left, corresponding to the topics we saw before. So you still have "Amazing", "Browser", "Disappointed".

And then next to that, crucially, we actually show the documents. And we show not just the documents, but actual snippets from the documents.

In fact, we show multiple snippets from each document, if there are multiple locations that are considered matches to one or more tags, with individual phrases highlighted as the justification for why a particular tag was applied to a document.

This lowers the cost of spot checks since your eye is drawn immediately to the place where there's a potential misclassification. This is very useful in the face of unreliable AI systems.

Moving along from 2014 to 2025...

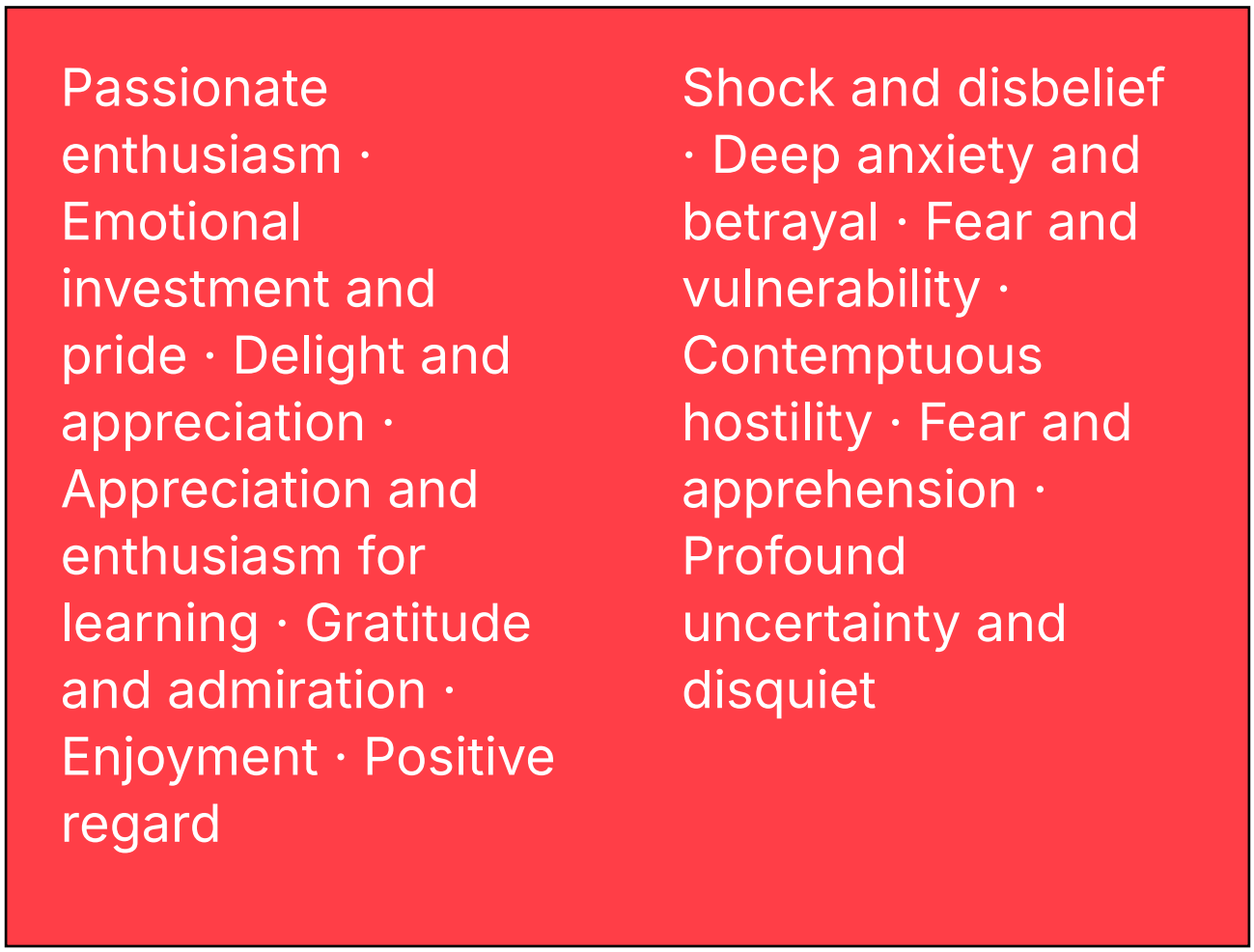
Podcast Vibes

Analyzing emotions in podcasts

So that was all a prelude to my podcast project, which I've been working on for the last few weeks. I'll go through this pretty quickly, because I guess this is probably going quite long now. But there's a lot of heritage from my experiences a decade ago that apply to analyzing the emotions in podcasts.

Thesis: Podcasts have vibes.

My basic thesis is that podcasts have vibes. By which I mean, every podcast inhabits a different part of the emotional landscape.



For example, here are some emotions from a few different podcasts.

My basic thesis is that people who spend time listening to stuff on the left probably end up in a different emotional space compared to people who listen to stuff on the right.

What's important is the connection between emotion and its subject matter.

And it not only matters what emotions a podcast is putting out -- I also really care what that emotion is being associated with.

For example, if the person speaking is disgusted about something, is it an event? A place? Or is it a person?

Podcasts inform how people *feel* about a subject

I think that that pretty much everyone, to one level or another, adopts their opinions and emotional valences from their surroundings.

In this way podcasts and other forms of news reporting provide not only information, but an *emotional interpretation* for complex events. (Kinda like reaction videos.)

Can we get at those feelings with language models?

It turns out LLMs are surprisingly good at this

So I ran a whole bunch of transcripts through Claude to extract out subject-emotion pairs, and it turned out to be ridiculously good at it.

(The only big issue I've noticed is that it will give incomplete results, ie. drop subject-emotion pairs even though they were present in the podcast.)

Subject	Emotion	Context	Evidence
Beans	Passionate devotion	Shows speaker's first impression of Steve's emotional connection to his work	"I first met Steve last year when he was the keynote speaker at Edible Communities Edible Institute, and I was captivated by his passion for beans, yes, but also by how Rancho Gordo works with and supports small farms and farmers here in the US and throughout North, Central, and South America, where many varieties of beans originated and grow."
The Bean Book	Deep personal investment	Reflects the speaker's commitment to creating something definitive in his field	"It's been a real labor of love as you could imagine. We've been self publishing books the last six or seven books we've done. And when Ten Speed asked us to do this and they wanted the definitive beam book, I thought we have to do it."
Professional publishing experience	Enthusiastic appreciation	Highlights the joy found in professional collaboration	"But working with a real professional publishing house was really the treat. And, honestly, working with the photographer, Ed, I've learned so much about photography that, we've already used started using a lot of those techniques in our own in house photography."
Previous career path	Self-criticism and frustration	Reveals personal struggles before finding his true calling	"I have to keep running reinventing myself with every job. I was getting really frustrated, and I thought you are a screw up, and it's not gonna work."

Here's a screenshot of some of the results from a podcast episode about food, specifically beans. You can see the whole thing **here**.

Every quote is grounded to the transcript using Claude's citations feature, which is fantastic.

And in my data pipeline, I go further and tie the transcript all the way back to timestamps in the original audio, so we have end-to-end provenance and attribution.

(In my local prototype, I can view a transcript and its emotional labeling, and click on any sentence to hear the relevant segment of the podcast as supporting evidence.)

The Prompt

The document you've been provided is an excerpt from a podcast transcript. I'm interested in the subjects being discussed, and how the speakers feel about those subjects, particularly the *value judgments* made by the speakers and what these imply about their speakers' views of the world.

Please organize every emotionally-loaded subject discussed in the transcript into a table in Markdown format, with one row for each (subject, emotion) pair. Each pair should be between a subject (person, place, or thing) and an emotion (feeling, quality, or vibe) that the person feels specifically about the subject. If there is more than one emotion associated with a subject, make a new row for each emotion.

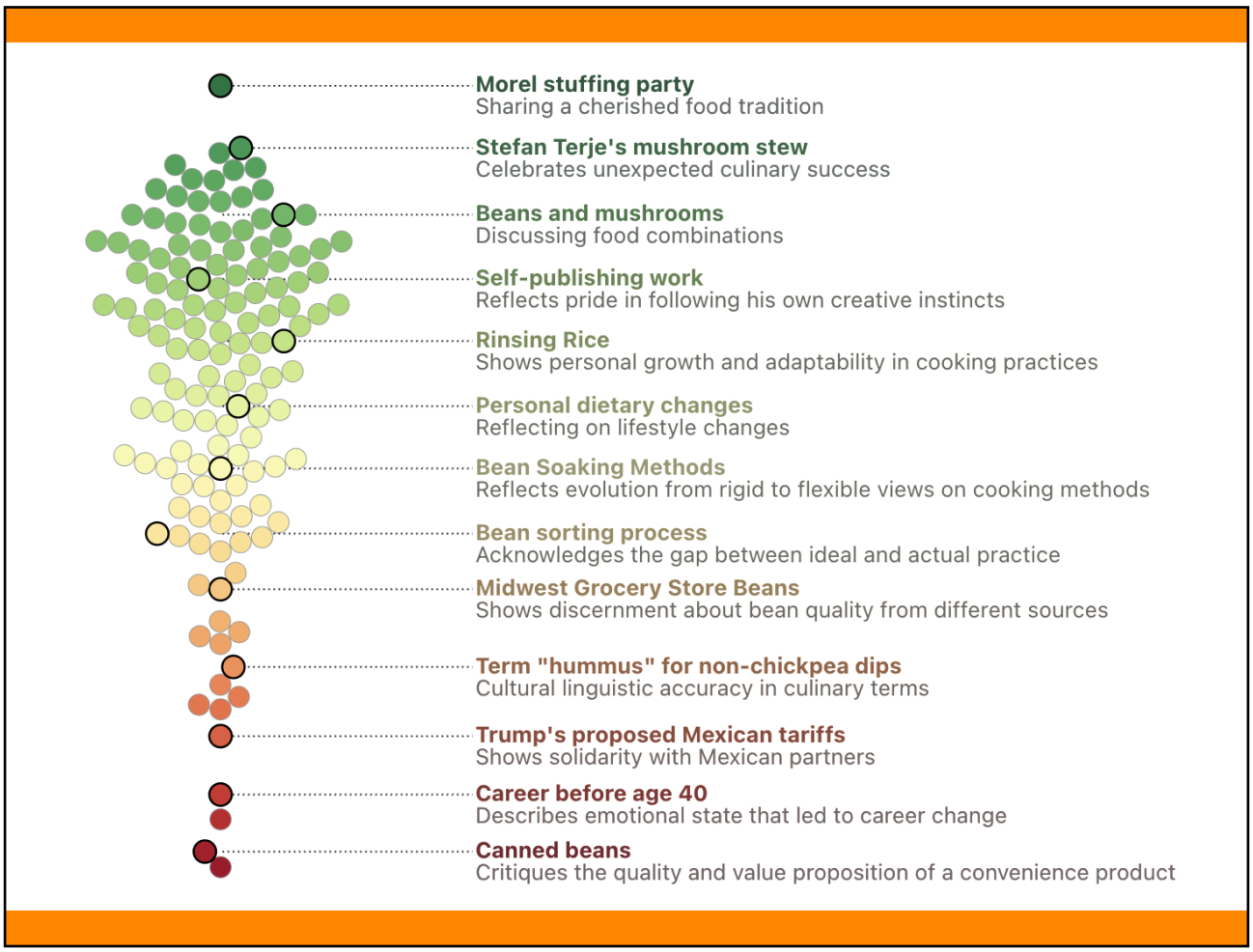
1. The first column should have the subject being discussed. Keep the subject short, and ground it in the source material.
2. The second column should have the emotion of that person towards the subject. Include only emotions that indicate how the person feels *about* the subject. Be expressive, nuanced, generous and richly emotive in your description, so long as your purple prose is supported by the text.
3. The third column should cite supporting quotes to support your assertions.
4. The fourth and final column should be a short remark that frames the supporting quotes in the larger context of the conversation. Do not refer directly to the speakers.

Here is what is most important:

- Completeness: Please include every single subject that has the speaker has an emotion about, whether positive or negative, offhand, direct, or implied.
- Correctness: Do not include emotions that are not attributable to specific subjects.
- Speakers only: Do not include emotions of people other than those who are speaking. For example, if the speakers reference somebody else's emotions, those should not be included in the table.
- Speaker meta-attitudes: Sometimes a speaker's emotion can be inferred from the way they talk about somebody else's emotion, so feel free to include those.
- Citation: Please cite the source documents to support your assertions.
- Richness: Use evocative and nuanced emotional descriptions that stay true to the words uttered by the speakers.

If there are no entries, then return an empty table consisting of the header alone.

Please return just the table.



The table is really useful but also quite large.

What if we want to see an overview?

One way to do this is, like the Concept Cloud, organize the emotions positive-to-negative emotion axis.

The green stuff at the top is positive, the red stuff at the bottom is negative, and the yellow is kind of in the middle.

I'm particularly amused at the position of canned beans at the very end of the negative spectrum... Though, of course, this axis itself is subjective.

More on this: Podcast Vibes Prototyping

Steve:	Exact yes.	
	And in fact, you know, we have this bean club, which, I sorted out sort of a joke because there were so many wine clubs in Napa.	Mild disdain (Wine clubs in Napa) Playful irony (Bean club)
	I thought, ugh, there's another wine club.	Mild disdain (Wine clubs in Napa) Playful irony (Bean club)
	Wouldn't it be funny to do a bean club?	Pride and enthusiasm (Bean club) Proud delight (Bean club)
	Well, there's twenty two thousand members in it, and we're about to go up to twenty six.	Pride and enthusiasm (Bean club) Proud delight (Bean club)
	And they have a a there's two dedicated Facebook groups, and they're the happiest, most interesting places on the Internet.	Pride and enthusiasm (Bean club) Proud delight (Bean club) Delight and satisfaction (Facebook groups)
	And people don't fight, and we talk about really interesting things.	Proud delight (Bean club) Delight and satisfaction (Facebook groups)
	Like and it goes from, I have this bean, what should I do with it?	Genuine enjoyment (Bean discussions)
	To, I made this elaborate cassoulet, and I think I want a bigger bean or a smaller bean.	Genuine enjoyment (Bean discussions)
	And it's really fun.	Proud delight (Bean club) Genuine enjoyment (Bean discussions)
	That's what I would say.	
	And the wait list we're working on, but it is thirty thousand people waiting to get into it.	Pride and enthusiasm (Bean club) Frustrated pride (Bean club waitlist)
	We just can't expand it as fast as we want.	Pride and enthusiasm (Bean club) Frustrated pride (Bean club waitlist)
	And we have a warehouse where we do the fulfillment.	Pride and commitment (Long-term workers)
	So we have, workers who've been there eight to ten years in the community.	Pride and commitment (Long-term workers)

And here's what I'm currently playing around with locally as a way to represent the underlying transcript in its original order together with these LLM-based classifications and annotations.

In my local prototype, I also have the ability to selectively add or remove particular speakers from the transcript, and re-rank the sentences from their original order, e.g. from positive-to-negative, or negative-to-positive, or other criteria.

I like presenting the annotations to the side of the original data because often the information you're dealing with has a fundamental structure that people are already familiar with, and using that as an anchor is a very effective way to make an interface feel intuitive -- it can fit into the existing mental schema of the person using it.

More advanced visualizations will inevitably need to reorganize and filter the data, or push up the level of abstraction.

But I've always found it useful to have something where the new data is treated as an annotation and the original structure is preserved.

Possible futures

Small picture

- Comparative analysis of the same topic across podcasts
- "Daily Podcast Pulse" – visualize all of the news podcasts for a given day

Big picture

- Collaborative human/ML auto-tagging at interactive speeds (unsolved research problem?)
- Turn this into a tool for lightweight, flexible analysis of text datasets

Auto-tagging is still hard if you want to do it within a reasonable cost or latency budget (for interactive exploration).

Another tricky issue is finding a data representation for subject matter that would enable easily identifying "the same topic" between podcasts. The current subject descriptions are often too specific for this purpose and need additional context to disambiguate. I don't know if asking the LLM to generate a sentence summary that encapsulates all of this context would work, or whether this would need something more like a hierarchical meaning representation to do the kinds of flexible topic tagging that I imagine would be really useful.

A tool I would love to have one day is something I think of as **spreadsheets, but for text**.

Spreadsheets are great with numbers, but don't deal well with freeform text.

Lots of real-world datasets -- product reviews, podcast transcripts, chat messages, email -- are combinations of text and structured data where the text is really the star of the show.

I would love to have a UI that lets me quickly add tags and explore their relationships, all the while doing lots of spot checks to make sure that my tags really mean what I think they mean.

Extra: The Data Pipeline

Parse RSS feeds into Podcasts & Episodes

To process an episode:

1. Transcribe it
2. Extract subjects & emotions w/ a moving window over sentences.
3. Deduplicate entries using vector similarity.

I built this in Go using River (job queue) and Postgres.