Lorenzo Naranjo November 11, 2026

Language Models in Finance

Much of what moves financial markets arrives as text: central bank statements, earnings calls, annual reports, news. For decades economists turned text into numbers by counting words. Language models now read text in a way that captures meaning, not just vocabulary, and the tools you use to write code are built on them. This chapter explains how text becomes data, what a language model is, how finance has used text, and where language models fail in ways that matter for financial research.

Counting words

The simplest way to measure the tone of a document is to count words from a list. A *dictionary* lists positive or negative words, and the tone of a document is, for example, the share of its words that are negative.

The difficulty is that meaning depends on context. Loughran and McDonald (2011) show that a general-purpose dictionary misreads financial text: in a large sample of annual reports, "almost three-fourths of the words identified as negative by the widely used Harvard Dictionary are words typically not considered negative in financial contexts." Words such as "tax," "cost," "capital," "board" and "liability" are on the general list of negative words, but in a 10-K they are just the vocabulary of accounting. Loughran and McDonald built word lists for finance, which are now standard. Even a good dictionary has limits. Counting words cannot tell "rates will not rise" from "rates will rise" unless rules for negation are added, which researchers do for simple cases such as a positive word preceded by "not." And no word list recognizes that two sentences with no words in common mean the same thing.

From words to vectors

To compare documents, we represent each one as a vector of numbers. In the *bag of words* representation, the vector has one entry per word in the vocabulary, counting how

often the word appears; word order is ignored. Two documents are similar when their vectors point in similar directions, measured by the *cosine similarity*

$$\cos(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}'\mathbf{b}}{\|\mathbf{a}\| \, \|\mathbf{b}\|}, \tag{1}$$

which is 1 when the vectors point the same way and 0 when they share no direction at all. With word counts, which are never negative, it lies between 0 and 1.

A bag of words treats "inflation eased" and "price pressures moderated" as unrelated, because they share no words. An *embedding* is designed to do better. It is a vector, typically with a few hundred entries, produced by a neural network trained so that texts with similar meanings get nearby vectors. The entries have no meaning on their own; only the distances between vectors do. With embeddings, (1) aims to measure similarity of meaning rather than of vocabulary.

It does so imperfectly, especially with the small models that run on a laptop. Such models capture what a text is about better than which way it points. With the model used in the notebook for this session, "inflation eased" is closer to "inflation elevated," its opposite (cosine 0.63), than to "price pressures moderated," which means the same thing (0.40), and "rates will rise" and "rates will not rise" score 0.87. Embeddings are good at finding documents about the same subject and poor at telling hawkish from dovish.

Language models

A language model is a neural network trained to predict the next piece of text, called a *token* (a word or part of a word), from the text that came before. Trained on an enormous amount of writing, a model that predicts the next token well has to represent grammar, facts, and a good deal of reasoning about the world, because all of these help predict what comes next.

The networks behind today's models are *transformers*, introduced by Vaswani et al. (2017). Their key idea is *attention*: to build the representation of each token, the network looks at every other token in the text and decides how much weight to give each one. In "the Committee decided to raise the target range," the representation of "raise" draws

on “Committee” and “target range,” which is what lets the model know which raise is meant. Stacking many layers of attention lets the model capture relations across whole documents.

A model trained only to predict text continues documents; it does not answer questions. The assistants you use are further trained on examples of instructions and good responses, and on human judgments of which answers are better. The embeddings of the previous section come from the same family of networks: a transformer that builds an internal representation of a text, condensed into one vector and trained so that texts with similar meanings get similar vectors.

Text in finance

Central bank communication is a good example of why text matters. Gürkaynak et al. (2005) show that monetary policy announcements move asset prices through two channels: the change in the policy rate, and what the statement signals about the future path of policy, which they find has a much greater effect on longer-term Treasury yields. Lucca and Trebbi (2009) score the content of FOMC statements automatically and find that “longer-dated Treasuries mainly react to changes in policy communication,” while short-term yields respond to the rate decision itself. Shapiro and Wilson (2022) measure the tone of FOMC transcripts with the finance dictionary of Loughran and McDonald (2011) and use it to estimate what the Fed cares about.

Language models make such measurement much easier. They can classify the tone of a statement, extract the numbers from an earnings release, summarize a 10-K, or compare this month’s statement with last month’s, across thousands of documents. The notebook for this session does the last of these with embeddings.

Where language models fail

They state falsehoods fluently. A language model produces likely text, and a likely-sounding citation, number or fact can be false. The chapter on how these notes are built gives examples from preparing this course. Every claim that matters has to be checked against its source.

They can know the future. A model trained on text up to 2025 has read about the market's reaction to events in 2020. If you ask it in a backtest to judge whether a 2020 news story was good or bad for a stock, it may be drawing on what happened next, which no investor could have known at the time. This *look-ahead bias* can make a strategy look profitable when it is not. A clean test uses only text, and models, from before the period being predicted.

Their answers depend on the question. Rewording a prompt can change a classification, and the same prompt can give different answers on different runs. Results that rest on a model's judgment should be checked for how much they change when the prompt does.

Data go where you send them. Text sent to a model run by a company, such as the assistants you use online, leaves your computer; a small model running on your own machine, like the one in the notebook, keeps it there. Use the tools WashU approves for the kind of data involved, and never send confidential information to a tool that is not approved for it.

Practice Problems

These problems give you a chance to practice the concepts introduced in this chapter. Try to solve each one on your own before expanding the solution.

Solutions to all problems can be found at https://lorenzonaranjo.com/class-materials/fintech/.

Problem 1 (Cosine Similarity of Word Counts). Two short statements use a vocabulary of four words: "inflation," "elevated," "eased," "labor." Their word counts are $\mathbf{a} = (2, 1, 0, 1)$ and $\mathbf{b} = (2, 0, 1, 1)$. Compute their cosine similarity, and explain what the number misses about the two statements.

Problem 2 (A Backtest with a Language Model). A student uses a language model released in 2025 to rate the tone of every FOMC statement since 2015, and finds that a strategy that buys Treasury bonds after statements the model rates as dovish earned high returns from 2015 to 2024. Which statement is most accurate?

a. The strategy would have earned these returns in real time, because the statements were public when they were released.
b. The model may have been trained on text describing how markets reacted to each statement, so its ratings may contain information investors did not have at the time; the result is not evidence of a usable strategy.
c. Look-ahead bias cannot arise, because the model only reads the statement.
d. The returns prove that language models understand monetary policy better than bond traders.

Problem 3 (A Dictionary That Misreads Finance). A general-purpose dictionary counts "liability," "tax," "cost" and "capital" as negative words. Select all statements that apply.

a. Measured with this dictionary, a bank's 10-K will look more negative than its content justifies.
b. Loughran and McDonald found that most words flagged as negative by a general dictionary in 10-Ks are not negative in a financial context.
c. Any dictionary, however carefully built for finance, measures tone perfectly.
d. Counting words, without a rule for negation, cannot distinguish "rates will rise" from "rates will not rise."

References

Gürkaynak, Refet S., Brian Sack, and Eric T. Swanson. 2005. "Do Actions Speak Louder Than Words? The Response of Asset Prices to Monetary Policy Actions and Statements." *International Journal of Central Banking* 1 (1): 55–93. https://www.ijcb.org/journal/v1n1/do-actions-speak-louder-words-response-asset-prices-monetary-policy-actions-and.

Loughran, Tim, and Bill McDonald. 2011. "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, And 10-Ks." *Journal of Finance* 66 (1): 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x.

Lucca, David O., and Francesco Trebbi. 2009. *Measuring Central Bank Communication: An Automated Approach with Application to FOMC Statements*. Working Paper No. 15367. https://www.nber.org/papers/w15367.

Shapiro, Adam Hale, and Daniel J. Wilson. 2022. "Taking the Fed at Its Word: A New Approach to Estimating Central Bank Objectives Using Text Analysis." *Review of Economic Studies* 89 (5): 2768–805. https://doi.org/10.1093/restud/rdab094.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, et al. 2017. "Attention Is All You Need." In "Advances in Neural Information Processing Systems." Special issue, *Advances in Neural Information Processing Systems* 30. https://proceedings.neurips.cc/paper/7181-attention-is-all.