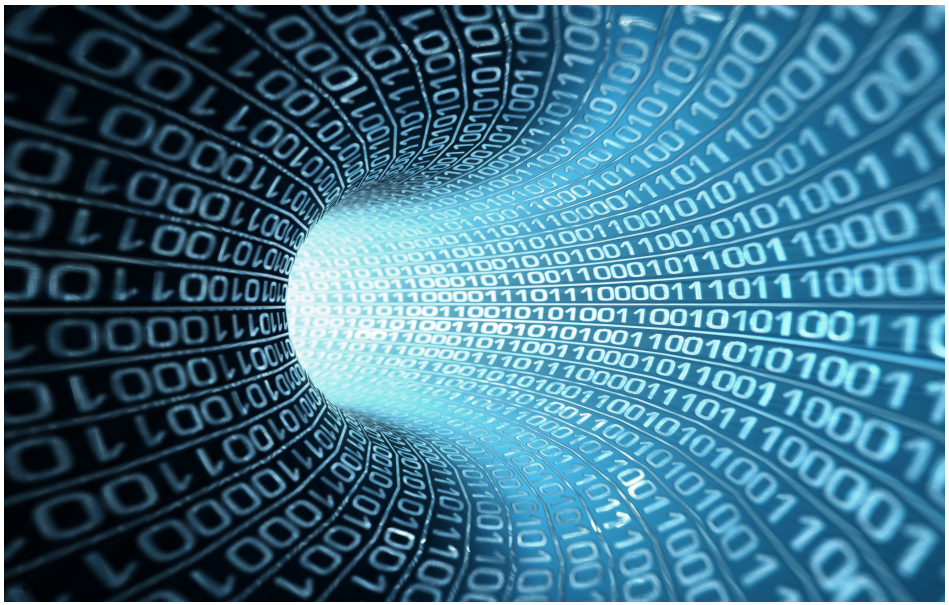
AI and Data Science Consulting

John H. Muller

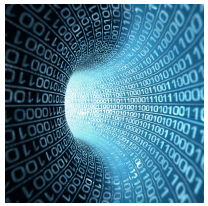
November 13, 2025



Outline

Introduction

About Me



- Computer Scientist (PhD), data scientist and finance quant,
- Over 15 years of experience in industry
  - Financial Services
  - Quantitative Investing
  - Retail
  - Image and video compression
- Passion for making my work matter.

Introduction

Skills, Experience and Tools

- Computer Science: algorithms and data structures
- Data Science
  - Supervised learning, clustering, regression, tree based methods
  - Neural Networks, Deep Learning and Generative AI
- Coding: Python and the python ecosystem
  - pandas, numpy, scikit-learn, pytorch ...
- SQL
- LangChain, Langgraph, CrewAI, Hugging Face

What: Compare 3 methods of classifying on a medical dataset.
For details on the dataset see Kaggle
https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset

Variables

1. id: unique identifier
2. gender: "Male", "Female" or "Other"
3. age: age of the patient
4. hypertension: 0 if the patient doesn't have hypertension, 1 if the patient has hypertension
5. heart_disease: 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease
6. ever_married: "No" or "Yes"
7. work_type: "children", "Govt_jov", "Never_worked", "Private" or "Self-employed"
8. Residence_type: "Rural" or "Urban"
9. avg_glucose_level: average glucose level in blood
10. bmi: body mass index
11. smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown"*
12. stroke: 1 if the patient had a stroke or 0 if not

- logistic regression
- gradient boosting
- random forests

Accuracy and loss on train, validation and test datasets. Loss is binary cross-entropy loss.

acc	loss	ds
0.950610	0.156825	train
0.948946	0.173313	val
0.953815	0.135607	test

Figure 2.1: Logistic Regression

acc	loss	ds
0.964761	0.094885	train
0.958935	0.119844	val
0.959839	0.104992	test

Figure 2.2: RandomForest

acc	loss	ds
0.950888	0.151802	train
0.948946	0.168683	val
0.953815	0.136938	test

Figure 2.3: Gradient Boosting

What: forecast the monthly **change** in non-farm payroll.
Data Source: BLS via St. Louis Fed site FRED.

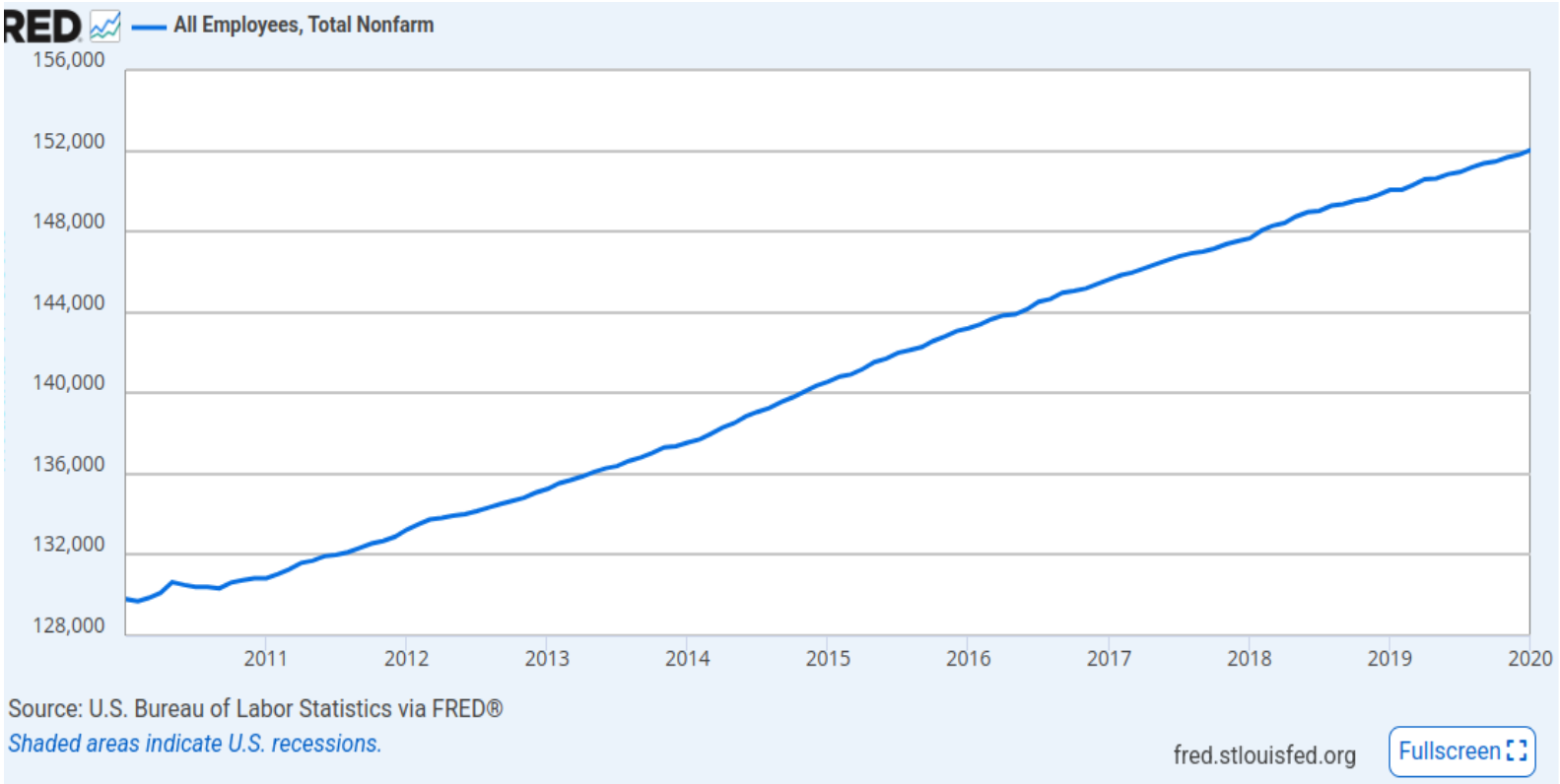


Figure 2.4: NonFarm Payroll level, 2010 - 2020

Not always a steady increase.

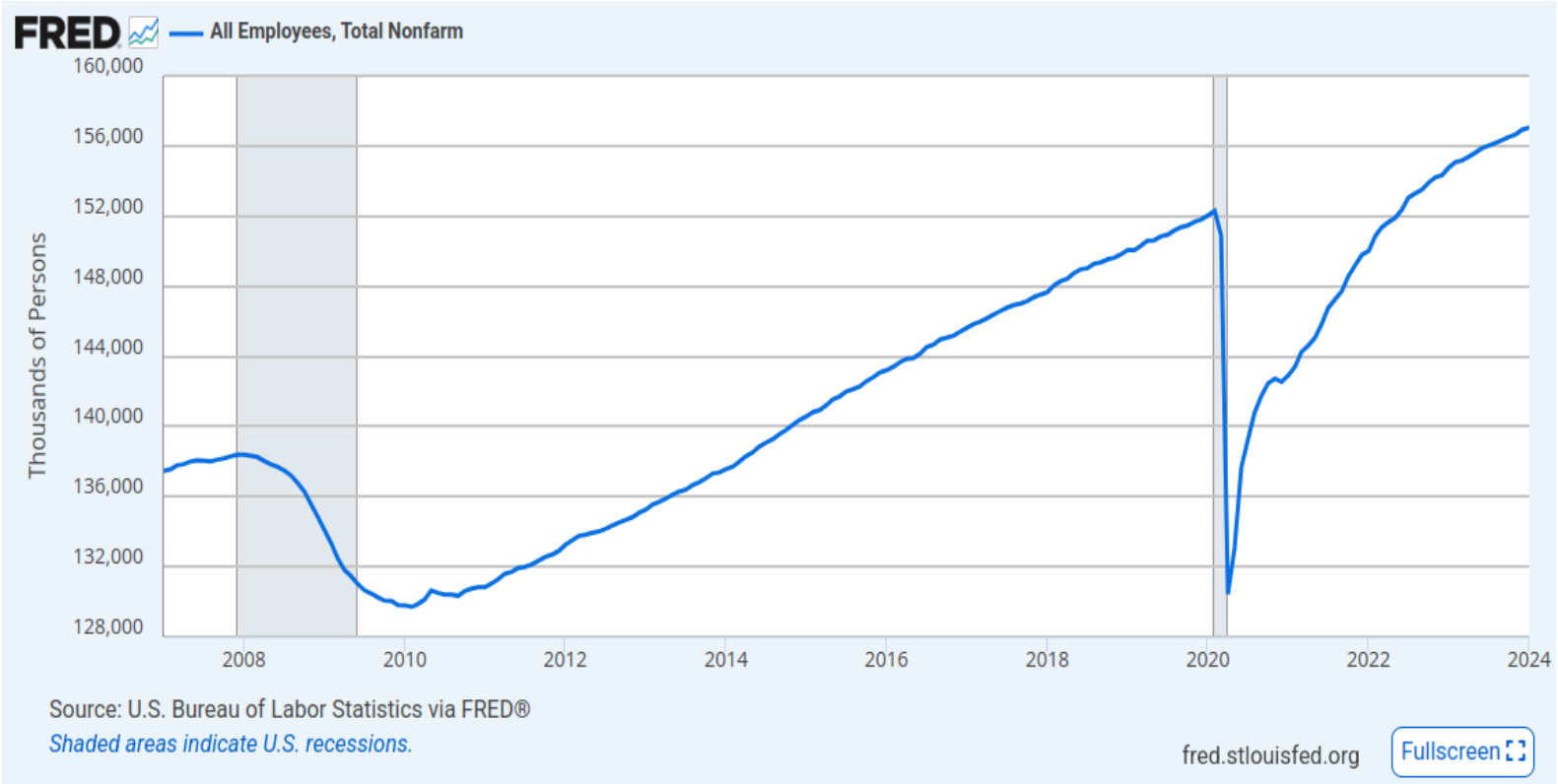


Figure 2.5: NonFarm Payroll level, 2008 - 2023

Changes in the level look less predictable.

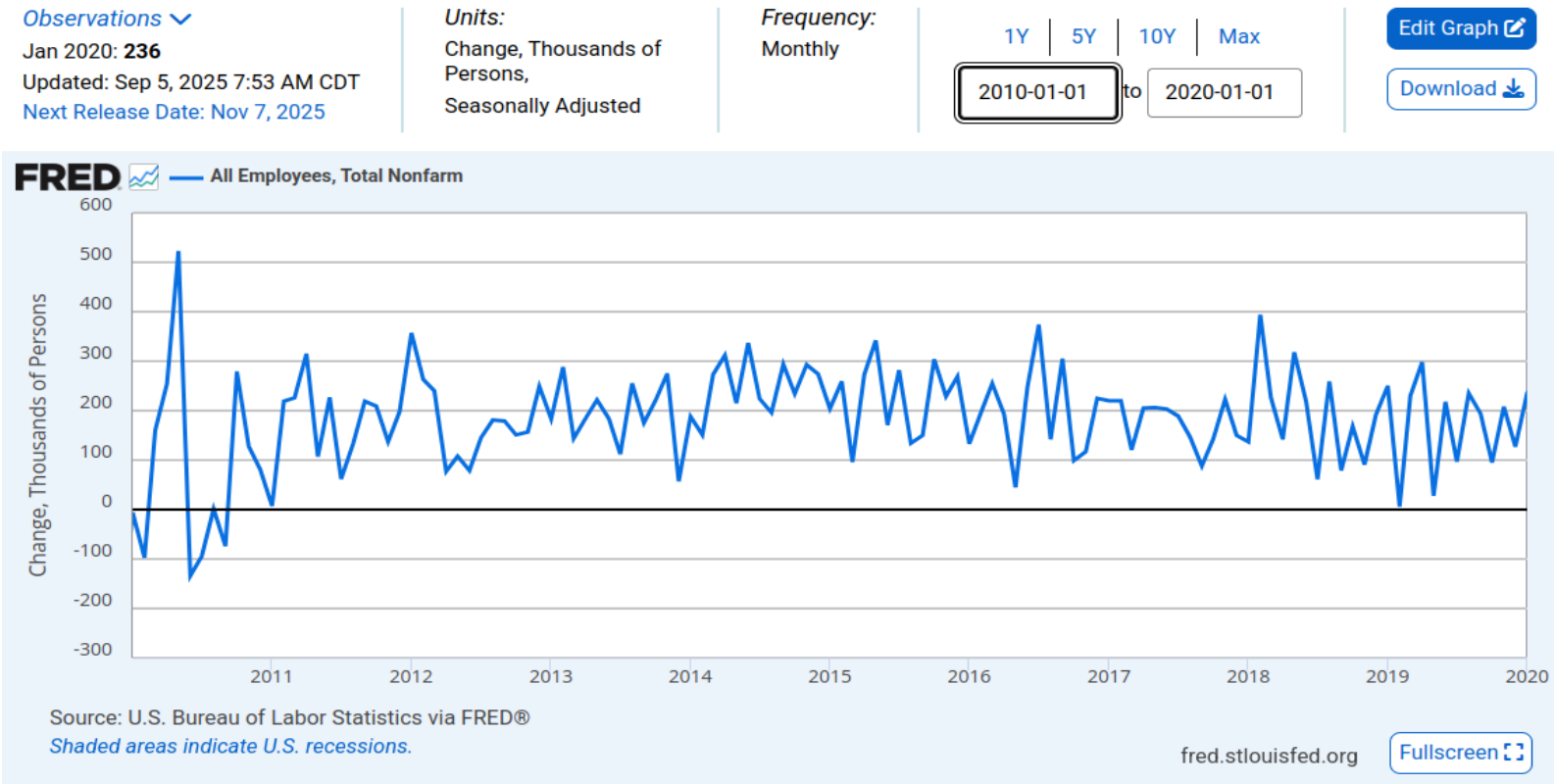


Figure 2.6: NonFarm Payroll, changes

What: forecast the next month's change in non-farm payroll.
How: regress the monthly change on

1. One or more lagged values.
2. Unemployment rate.
3. Civilian Labor force.
4. Other data that might be helpful:
   - GDP (quarterly)
   - unemployment claims (weekly)
   - ADP payroll changes (monthly)
   - Google trend values for "job", "layoff", ... (hourly)

I will start with the first 3 above.

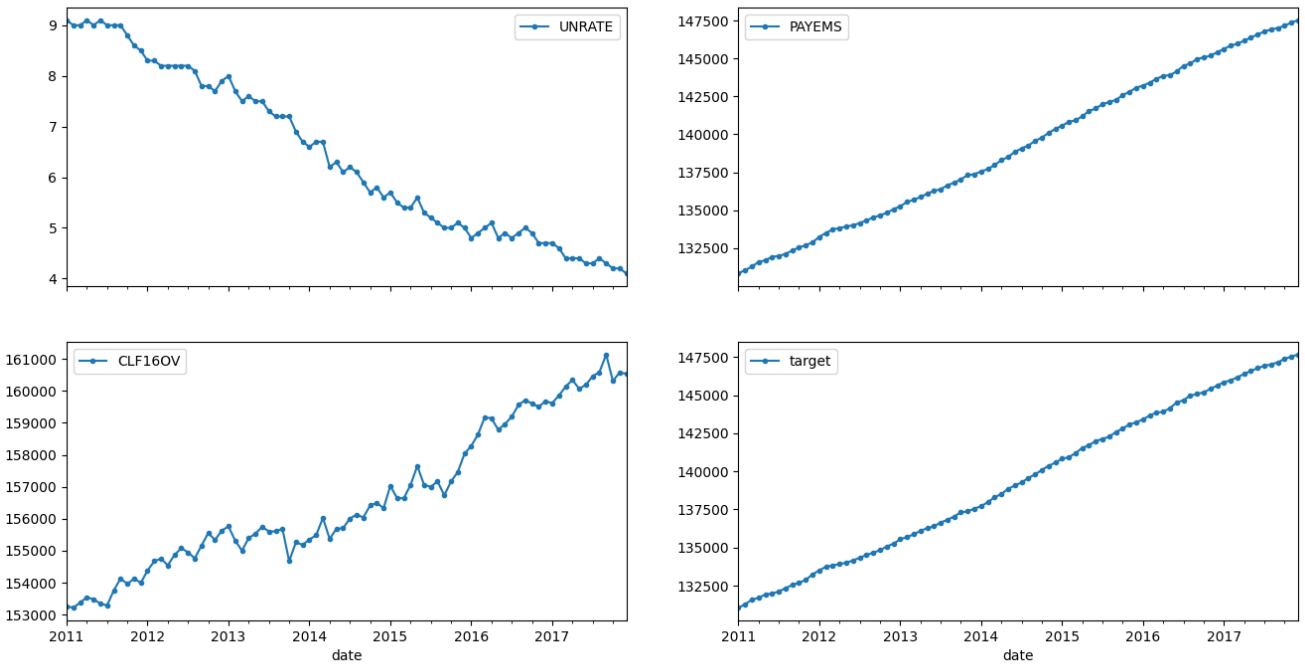


Figure 2.7: target and input variables, level

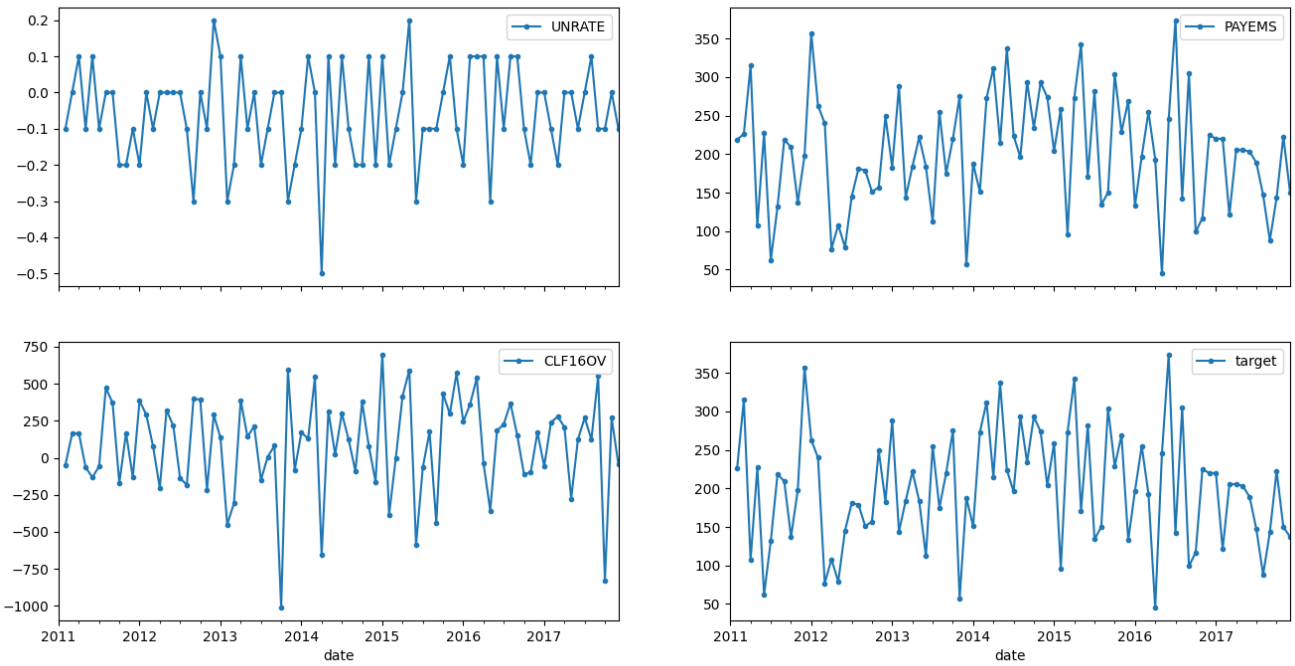


Figure 2.8: target and input variables, changes

How well does it work? Is it **useful**? See the true vs. predicted values below. Traders often react to whether the number beat the expected number. We can use 200 as a proxy for expectation, so we accurately predicted above or below about half the months.

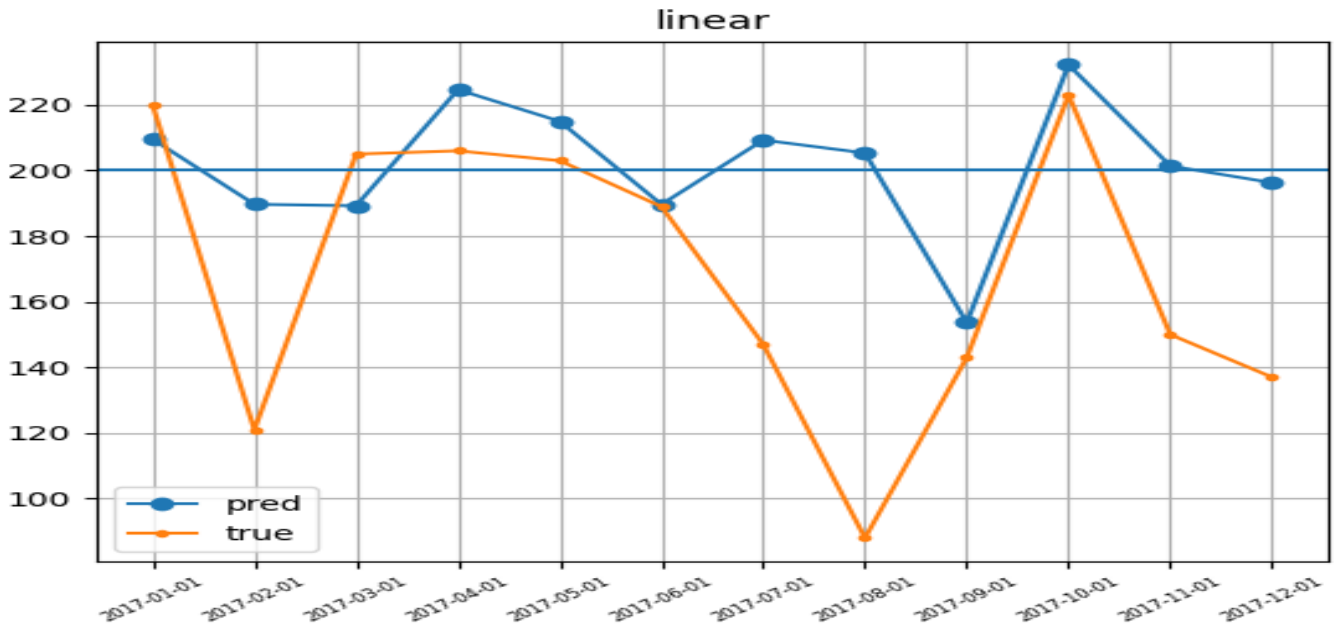


Figure 2.9: results using a linear model

What: predict a viewer's rating for a movie.
How: Embedding NN for both viewer and the movie.
The similarity between viewer and rating determines the predicted rating.

Pytorch model

```
class MF(nn.Module):
    def __init__(self, n_users, n_movies, emb_size=100):
        super(MF, self).__init__()
        self.n_users = n_users
        self.n_movies = n_movies
        self.user_emb = nn.Embedding(n_users, emb_size)
        self.movie_emb = nn.Embedding(n_movies, emb_size)

        # initializing the matrices with a positive number
        # supposed to help generally will yield better results
        self.user_emb.weight.data.uniform_(0, 0.5)
        self.movie_emb.weight.data.uniform_(0, 0.5)

    def forward(self, users, movies):
        m = self.movie_emb(movies)
        u = self.user_emb(users)
        return (u * m).sum(1)  # taking the dot product
```

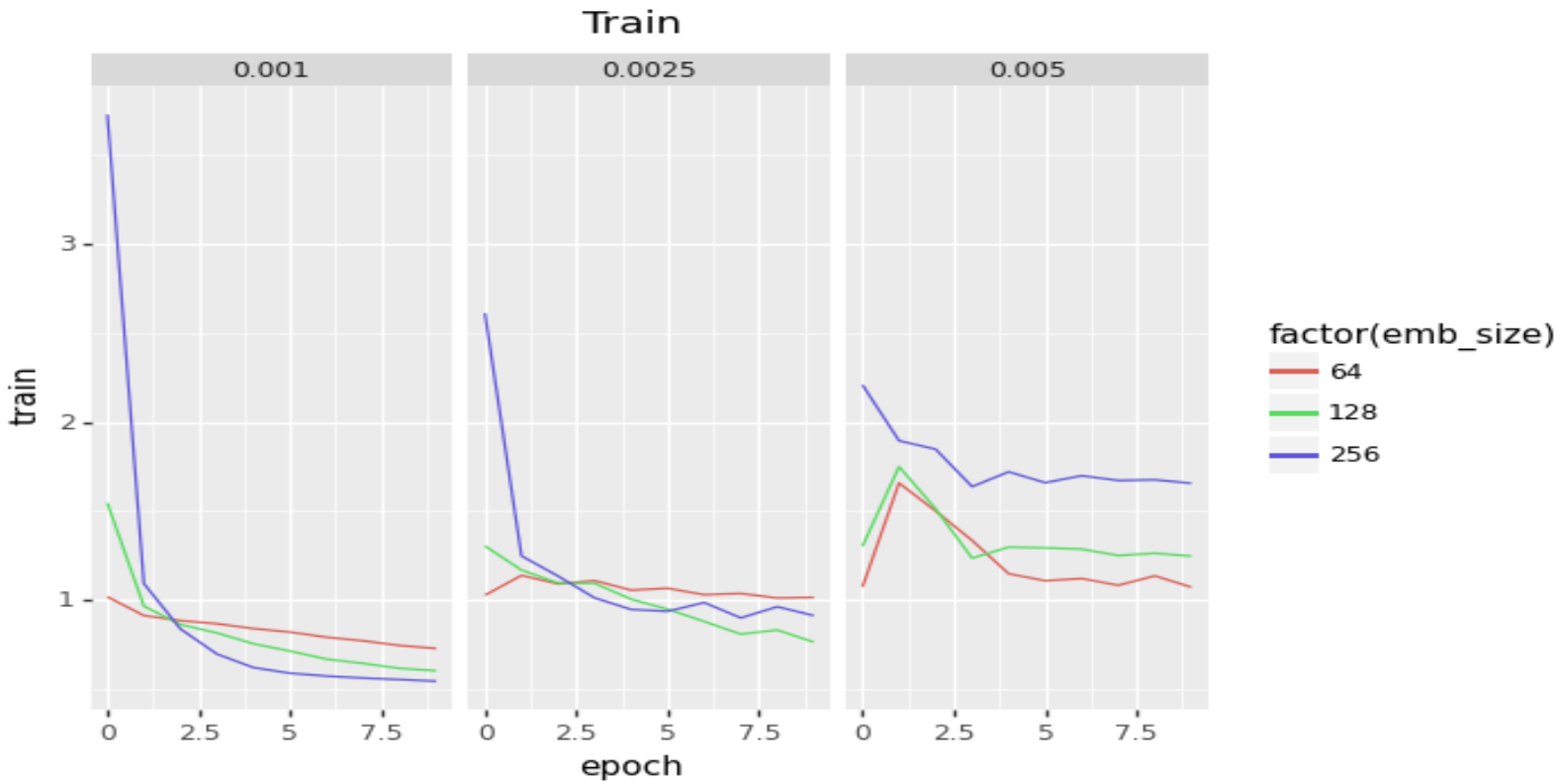
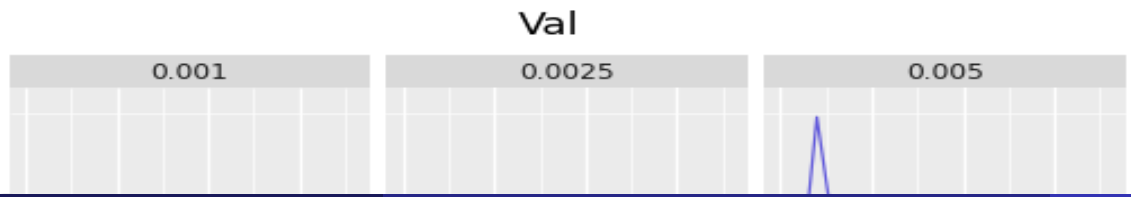


Figure 2.10: Interface to gather 5 basic facts



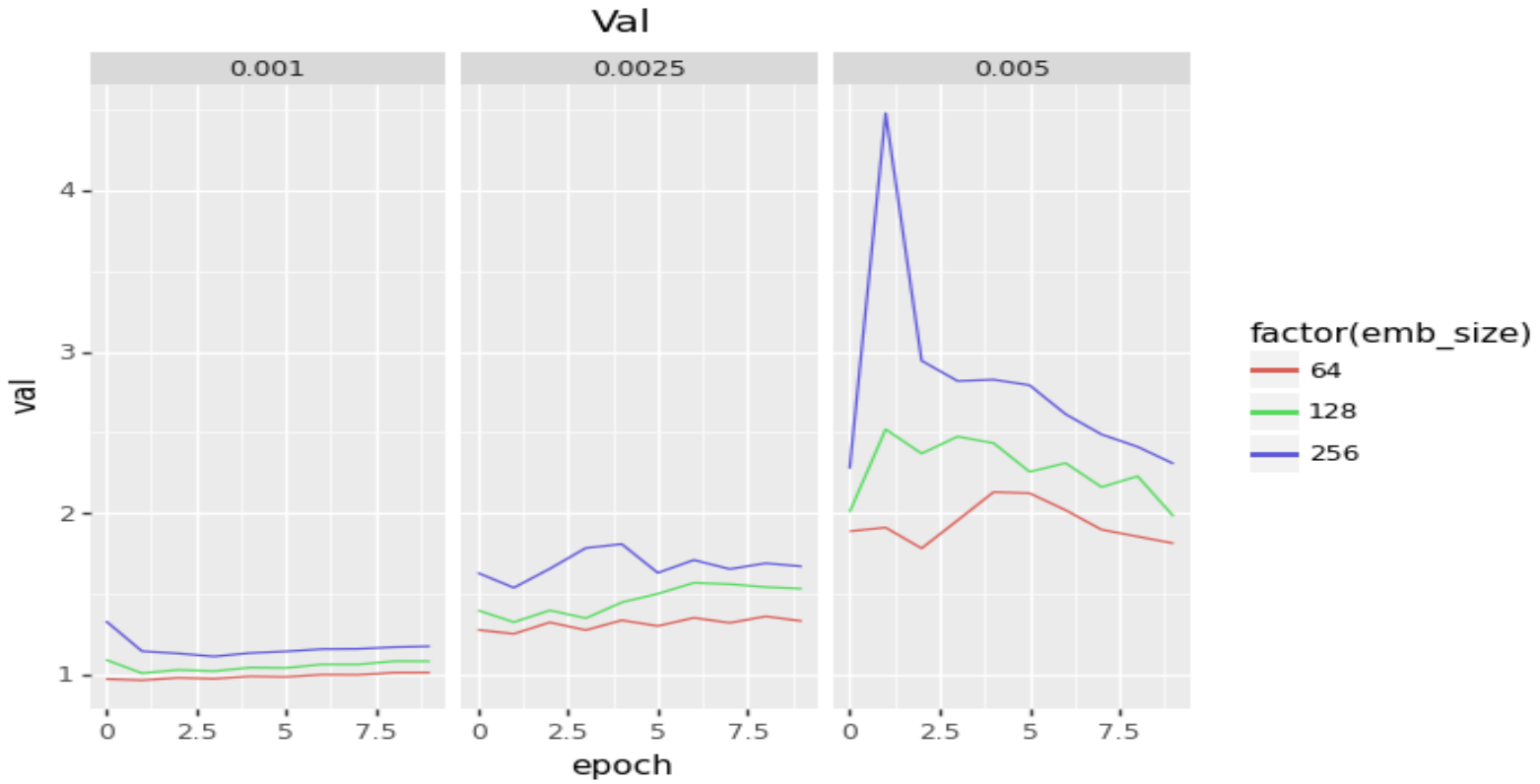


Figure 2.12: Interface to gather 5 basic facts

Test error was 1.23 with 64 dimensional embeddings

What: Given a minimal set of *facts*, generate a narrative or story based on the facts.

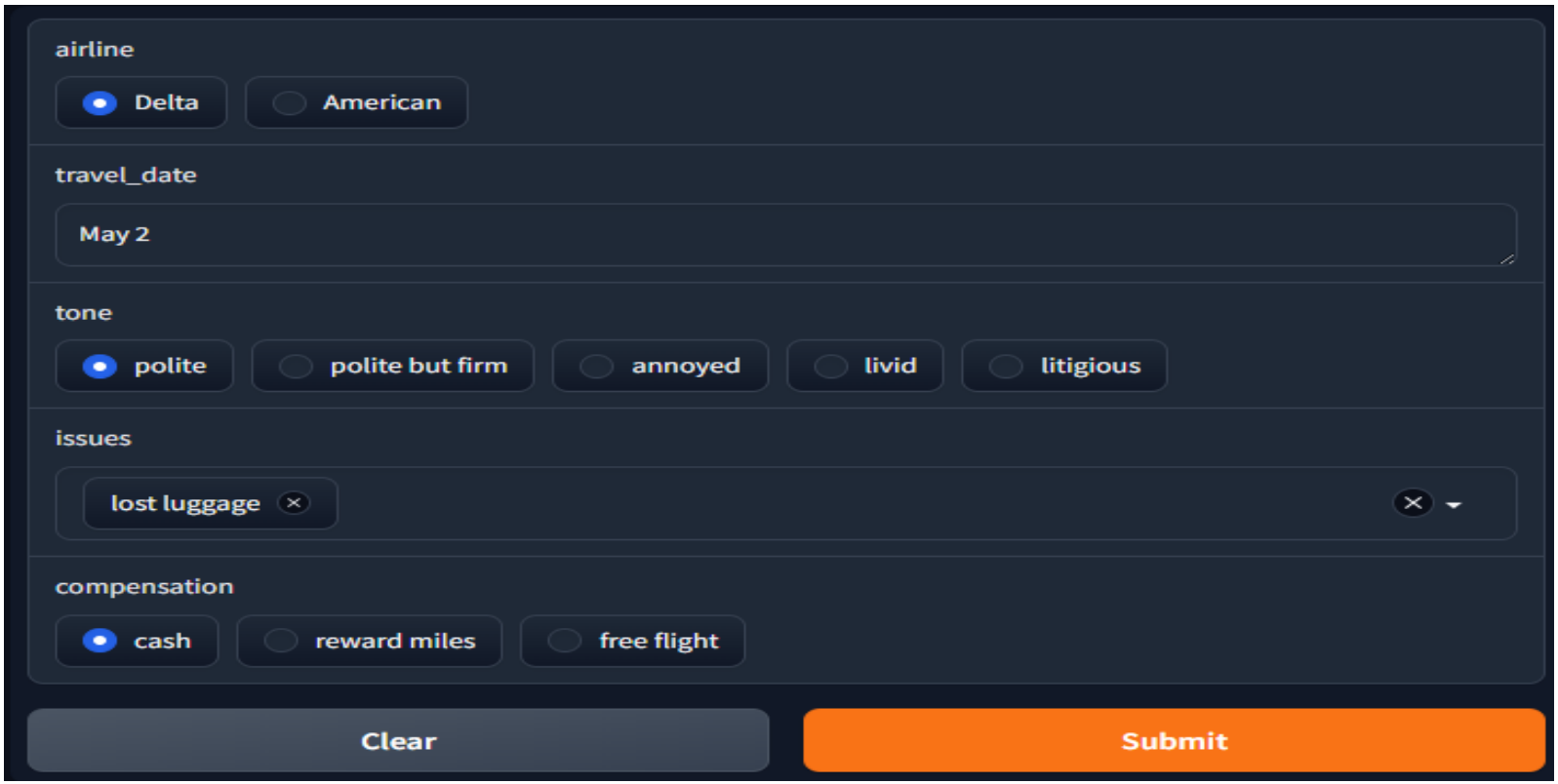


Figure 2.13: Interface to gather 5 basic facts

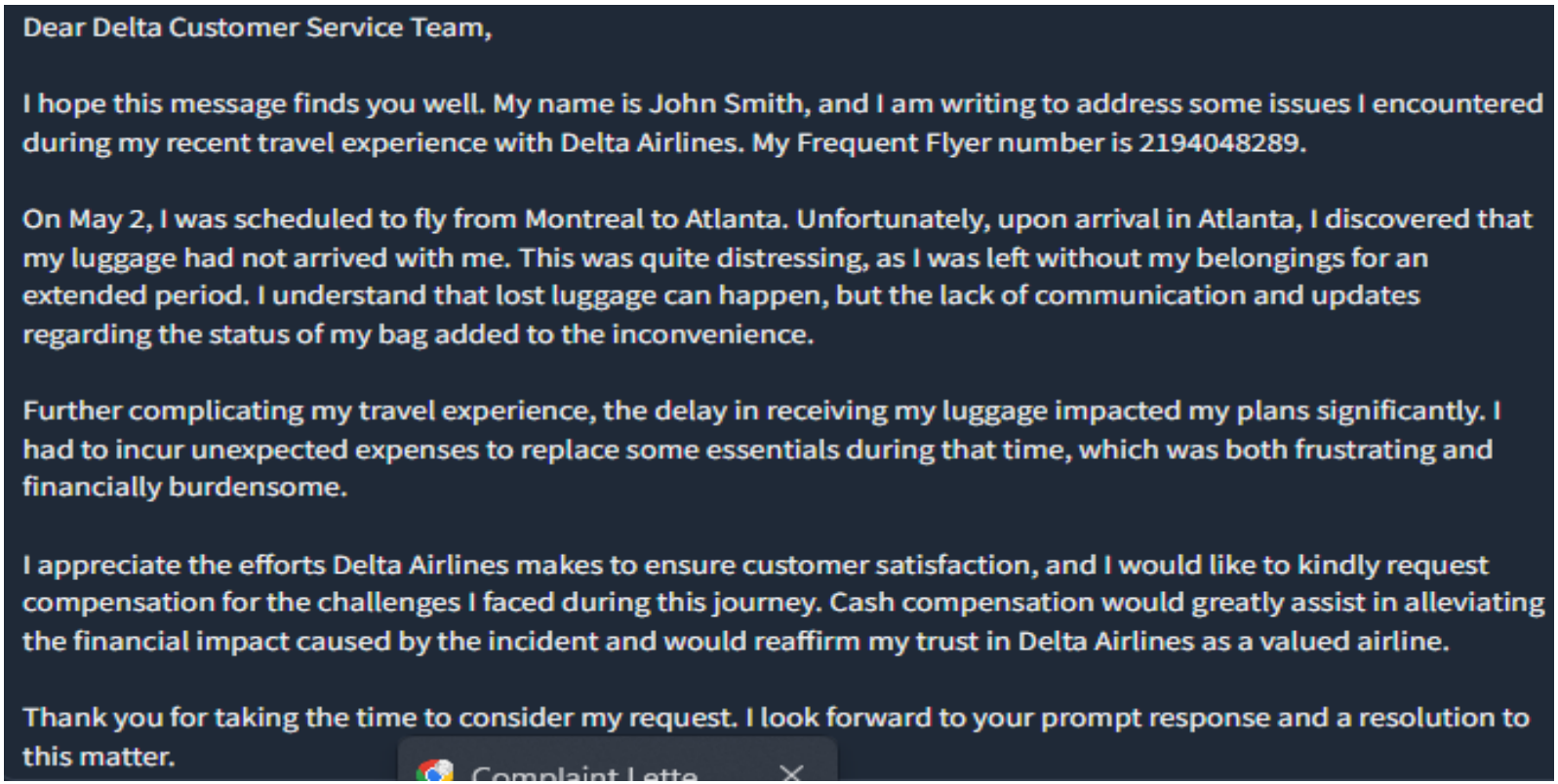
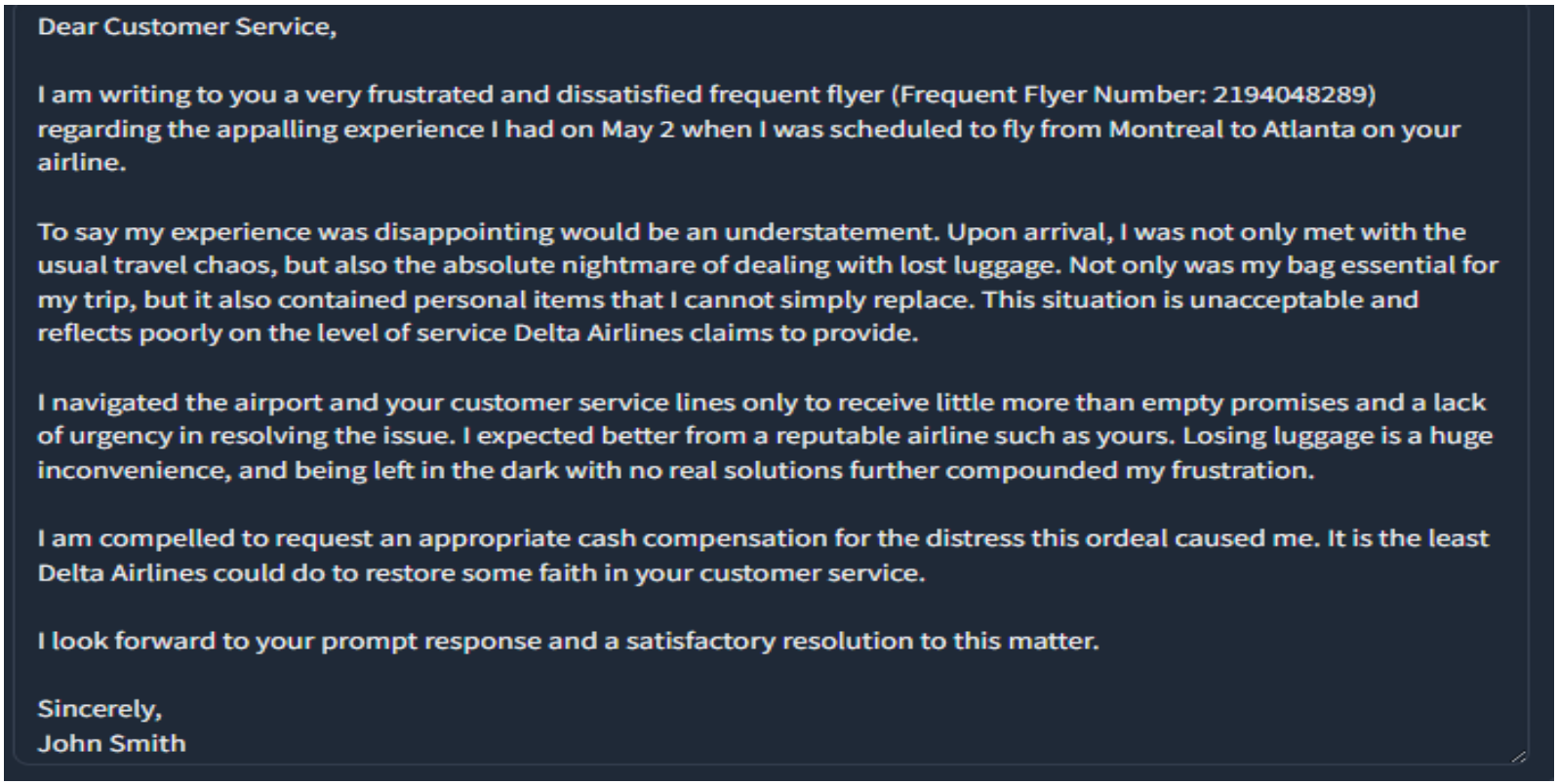


Figure 2.14: resulting "polite" letter


Dear Customer Service,

I am writing to you a very frustrated and dissatisfied frequent flyer (Frequent Flyer Number: 2194048289) regarding the appalling experience I had on May 2 when I was scheduled to fly from Montreal to Atlanta on your airline.

To say my experience was disappointing would be an understatement. Upon arrival, I was not only met with the usual travel chaos, but also the absolute nightmare of dealing with lost luggage. Not only was my bag essential for my trip, but it also contained personal items that I cannot simply replace. This situation is unacceptable and reflects poorly on the level of service Delta Airlines claims to provide.

I navigated the airport and your customer service lines only to receive little more than empty promises and a lack of urgency in resolving the issue. I expected better from a reputable airline such as yours. Losing luggage is a huge inconvenience, and being left in the dark with no real solutions further compounded my frustration.

I am compelled to request an appropriate cash compensation for the distress this ordeal caused me. It is the least Delta Airlines could do to restore some faith in your customer service.

I look forward to your prompt response and a satisfactory resolution to this matter.

Sincerely,
John Smith

Figure 2.15: resulting "livid" letter

What: Condense a lengthy passage of several pages down to several paragraphs. Click the link below for a demo.

▸ Link

Please note a few things:

- You will probably have to restart the app
- Be patient, I am using limited processing

Conclusion and next steps

Let me help you solve your problems.
Next steps?

Conclusion and next steps

Thank you!

John H. Muller

jmuller.ics88@gtalumni.org

+1 (617) 669-2204