Homework report

Aleksandr Makarov
aleksandr.makarov@ut.ee
Student, sometimes employee
University of Tartu

Chat-GPT3.5
https://chat.openai.com/chat
Model
OpenAI

Abstract

This is all about the struggles and suffering since the bigger the suffering, the bigger the outcome(maybe). - by me.
I have never seen latex before in my life. I have also started this homework a bit earlier and struggled too much, I should have waited for the lecture on Thursday. I am really grateful for this course since otherwise, I wouldn't have had such an intense experience of struggling together with other people(this isn't a joke, this is indeed supporting).

1 Main part

In this report, we will analyze the Data Thinking Zulip chat data using logistic regression, linear regression, and embeddings. First, we will clean the data and put it into a Polars DataFrame, and then compute summary statistics of the dataset. We will then proceed to apply the three techniques to analyze the data and provide insights.

We started by loading the chat data from the provided JSON file and extracted the content and sender of each message. We then created a dictionary with column names and values and created a Polars DataFrame. We did not perform any further data cleaning or preprocessing as that was beyond the scope of this analysis. In this analysis, we use logistic regression to predict the sender of messages in the Data Thinking Zulip Chat data based on the embeddings generated using Word2Vec. Logistic regression is a popular classification algorithm that estimates the probability of a binary event occurring. It is used to model the relationship between a set of independent variables and a dependent binary variable. In our case, the independent variables are the Word2Vec embeddings, and the dependent variable is the sender of the message (represented as a binary variable). The logistic regression equation used in this analysis is: $P(y = 1|X) = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_p X_p)}}$

where y is the binary dependent variable (i.e., the sender of the message), $X_1, X_2, ..., X_n$ are the independent variables (i.e., Word2Vec embeddings), $\beta_0, \beta_1, \beta_2, ..., \beta_n$ are the coefficients (i.e., weights) to be estimated, and e is the base of the natural logarithm. Visualization sucks(Fig. 1), and so do I.

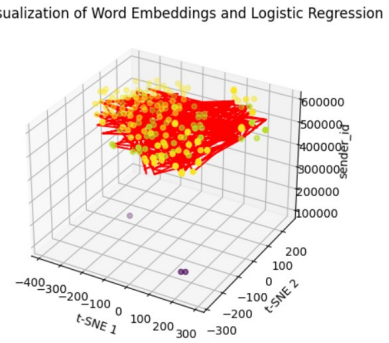

Figure 1: My corrupted imagination together with copilot and chat-gpt.

Linear Regression:

In this analysis, we use linear regression to predict the sender of messages in the Data Thinking Zulip Chat data based on the embeddings generated using Word2Vec. Linear regression is a popular regression algorithm that estimates the relationship between a dependent variable and one or more independent variables. In our case, the dependent variable is the sender of the message, and the independent variable is the Word2Vec embeddings. I tried my best to guess what was meant by visualizing linear regression for embeddings. Since it has too many dimensions, I tried to do 3d thing because I did not see any other options. The outcome is visible in figure 2

The linear regression equation used in this analysis is: $P(y = 1|X) = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_p X_p)}}$

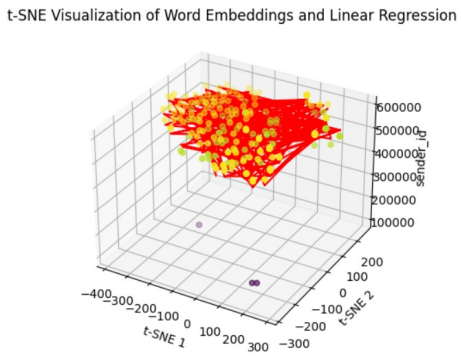

Figure 2: My corrupted imagination together with copilot and chat-gpt.

where y is the dependent variable (i.e., the sender of the message), $X_1, X_2, ..., X_n$ are the independent variables (i.e., Word2Vec embeddings), $\beta_0, \beta_1, \beta_2, ..., \beta_n$ are the coefficients (i.e., weights) to be estimated.

Embeddings: In this analysis, we generate embeddings for the messages in the Data Thinking Zulip Chat data using Word2Vec. Word2Vec is a popular algorithm for generating word embeddings, which are dense vector representations of words in a high-dimensional space. These embeddings are used to represent the meaning of words in a numerical format, which can be used in various natural language processing tasks. In our case, we use Word2Vec to generate embeddings for the messages in the Zulip Chat data, which are then visualized using t-SNE.

The Word2Vec model used in this analysis is trained on the text data in the 'content' column of the Polars DataFrame. The text data is preprocessed as needed (e.g., removing stop words, tokenizing text, etc.), and the embeddings for each message are generated using the first word in the message. These embeddings are then used to train the logistic and linear regression models, and also visualized using t-SNE fig. 3.

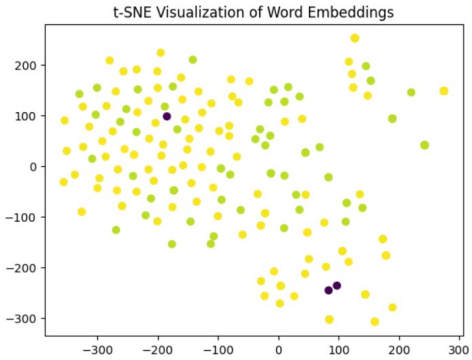

Figure 3: My corrupted imagination together with copilot and chat-gpt.