

# Decoding Congressional Stock Trades: An Industry and Committee-Focused Analysis with Graph Neural Networks and Large Language Models

Suyeol Yun \*

December 5, 2023

## Abstract

In this study on congressional stock trading, I refine the estimation of excess returns by focusing on the congressperson-ticker-timing level, departing from the traditional congress-aggregate level, calendar-based yearly approach methods which are less specific. This enhanced estimation provides a more detailed and relevant analysis for individual trades. Additionally, I significantly advance the understanding of influential factors beyond traditional methods that relied on manual feature design and regression on tabular data. By employing a Graph Neural Network (GNN), adept at machine learning in graph-structured data, I effectively train the model to learn how to aggregate information channeled through the network, identifying industry sectors and committee assignments as pivotal factors in congressional stock trading. To address the interpretability challenges of the GNN, I initially use GNNExplainer, and subsequently introduce a Large Language Model (LLM) for predictive analysis and theorizing. Although the LLM does not yet equal the GNN in predictive power, it provides crucial interpretability, enabling deeper insights into the motivations and behaviors in congressional stock trading. This comprehensive approach emphasizes the importance of specific industry and committee assignments in these transactions and underscores the necessity for more stringent scrutiny and regulatory frameworks to ensure ethical financial practices.

---

\*Master Student, Department of Political Science, MIT. Email: [syyun@mit.edu](mailto:syyun@mit.edu)

# 1 Introduction

In April 2020, Senator Ron Wyden began buying stocks in semiconductor companies such as Broadcom, Applied Materials, and KLA Corporation, initiating these purchases on April 6th, 2020, and then selling them a year later on either April 6th or April 16th, 2021. This sequence of transactions demonstrates a deliberate pattern of multiple buys followed by sales, highlighting not only strategic decision-making but also the significance of precise timing. The critical nature of this timing is accentuated by a key legislative event: just days before Senator Wyden sold his semiconductor stocks, on April 1st, 2021, President Biden announced a substantial investment plan for the U.S. chip industry. The synchronization of Wyden’s stock sales with this major legislative announcement suggests that members of Congress may have access to industry-level information that informs their trading decisions, enabling them to time their actions in alignment with events that could impact the market. This scenario emphasizes the vital role of timing in congressional stock trading, indicating that legislators might leverage sensitive, industry-specific information to coordinate their trading activities with legislative developments, thus raising important questions about the interplay of legislative insight, personal financial benefit, and the anticipation of market-moving events.

Following the examination of Senator Wyden’s trading activities, it becomes evident that the academic exploration of congressional stock trading is divided into two primary strands with distinct objectives. The first strand, exemplified by the works of Ziobrowski et al. (2004, 2011) and Eggers and Hainmueller (2013), focuses on estimating excess returns, quantifying the financial gains that congress members might realize through their stock transactions. This research investigates whether members of Congress achieve returns exceeding standard market benchmarks, potentially indicating an informational advantage from access to non-public information. The second strand, as seen in the work of Eggers and Hainmueller (2014), delves into identifying the predictors or driving factors behind congressional stock trading, aiming to uncover the various influences on congress members’ stock trading decisions, including specific events, legislative activities, and personal connections. Together, these strands offer a multifaceted view of congressional stock trading, one quantifying financial outcomes and the other investigating the decision-making processes and influences behind these trades.

The case of Senator Wyden’s 2020 trading activities brings to light a critical shortcoming in the prevailing research on congressional stock trading, particularly in the context of excess return studies by

authors like Ziobrowski et al. (2004, 2011) and Eggers and Hainmueller (2013). These studies predominantly focus on the collective trading performance of Congress as a whole, determining an average excess return for all members. However, such an approach may not reveal the full picture, especially regarding our primary goal of understanding whether individual Congress members can achieve excess returns from specific transactions involving certain firms or industries. The generalization inherent in these studies, which emphasize aggregate data, potentially obscures the instances where individual members might realize significant excess returns from trades in particular sectors or companies.

Another significant limitation of the excess return strand of congressional stock trading literature relates to the treatment of timing in stock transactions. This body of work, Ziobrowski et al. (2004, 2011); Eggers and Hainmueller (2013), primarily employs a calendar time-based synthetic portfolio approach (Shanken, 1996; Annaert et al., 2009). This method involves replicating Congresspersons' stock transactions, buying and selling stocks based on their decisions and subsequently assessing the portfolio a year later. However, this approach fails to adequately capture the crucial element of timing in congressional stock trading. It overlooks the strategic significance of when these trades are executed, particularly in relation to legislative events and market movements. The importance of timing has been underscored by researchers like Schweizer (2011) and (Tahoun, 2014), who highlight that the precise timing of transactions in relation to external events can be a key determinant in the profitability and strategic nature of stock trades by Congress members. By adhering to a rigid, calendar-based analysis, existing studies may not fully appreciate the timing strategies employed by individual Congresspersons, which can offer deeper insights into how legislative information and market anticipations are potentially utilized in their trading decisions.

Building on the limitations identified in the existing research on excess return studies of congressional stock trading, I propose a distinct approach: estimating excess returns at the congressperson-ticker-timing level. This method involves constructing a sequence of buys followed by sales for specific ticker transactions of a congressperson, thereby assessing the excess return of these transactions over a particular period. This approach captures the "life-cycle" of a transaction from its initiation to closure, offering a more detailed analysis than previous broad averages.

The results of this method are dual in nature. Firstly, they align with the findings of studies like those by Eggers and Hainmueller (2013), confirming that on average, Congress members' stock transactions yield mediocre returns. However, more significantly, this approach allows for the identification of abnor-

mally high returns from specific transactions. The discovery of these outliers is crucial as it necessitates a deeper exploration into the second strand of research, which seeks to identify the explanatory factors driving these specific transactions.

The notable work of Eggers and Hainmueller (2014) represents a pivotal contribution to the second strand of inquiry in congressional stock trading research, focusing on identifying the driving factors behind these transactions. Eggers and Hainmueller (2014) sought to unravel the reasons behind the generally mediocre profitability of congressional stock transactions, a finding initially presented in their 2013 study (Eggers and Hainmueller, 2013). Eggers and Hainmueller (2014) hypothesized that transactions with a “political connection,” such as local ties or connections to firms through campaign contributions, might outperform those without such connections. Interestingly, their findings supported this hypothesis, indicating that politically connected transactions tended to yield better returns than those lacking such connections.

However, a critical aspect of Eggers and Hainmueller (2014) also revealed a lack of evidence to support the notion that Congress members disproportionately invest in companies directly linked to their committee assignments. This finding poses a significant puzzle, presenting a conflict not only with individual cases like Senator Ron Wyden’s trading activities but also with well-established understandings of congressional committees. This discovery stands in stark contrast to the established findings regarding committees’ specialization in various topics, as conferred by statutory jurisdiction (Myers, 2009) or bill referral (King, 1994). Congressional members are known to specialize in these areas (Asher, 1974a), using their unique knowledge and expertise to influence legislative outcomes (Boros and Fenno, 1968; Gilligan and Krehbiel, 1989; Kiewiet and McCubbins, 1991; Krehbiel, 1992; Curry, 2019). In the case of Senator Wyden, his role as Chairman of the Senate Finance Committee and his involvement in reviewing the Chips Act, which aimed to incentivize domestic chip production, could potentially influence the stock prices of related companies.

Therefore, the lack of evidence found by Eggers and Hainmueller (2014) for a link between committee assignments and stock trading activities calls for further investigation. Their study utilized a regression analysis with binary indicators for firm connections, using these to predict the portfolio weight of specific firms based on whether they lobbied a committee on which a member served or provided campaign contributions. However, this approach has two significant limitations.

Firstly, the exclusion of industry-level information in Eggers and Hainmueller’s analysis overlooks a

critical aspect of congressional stock trading. The importance of industry-specific data is exemplified by cases like that of Senator Ron Wyden. Instances where multiple companies within a specific industry engage in lobbying activities while legislation relevant to that industry is under review can significantly deliver industry-specific information. This was apparent in situations similar to Senator Wyden’s, where as a committee chair, he was involved in legislation affecting industries such as semiconductor. In such scenarios, members like Wyden, deeply engaged in the legislative process, might gain insights that could influence their trading decisions.

Secondly, this binary measure overlooks the fact that lobbying activities by various companies within the same industry can collectively provide a richer tapestry of industry-level information related to legislation. Congresspersons, especially those on relevant committees, might be privy to this aggregated industry-specific knowledge, which could inform their decisions to buy different stocks, offering a more informed perspective on potential legislative impacts on the industry. The current binary model, focusing on isolated instances of firm lobbying, fails to account for this broader context where a congressperson might be influenced not just by a single firm’s lobbying efforts but by the cumulative information arising from an entire industry’s lobbying activities. This limitation highlights the need for a more comprehensive methodological approach that can properly capture and measure the complexities of these multi-layered interactions, taking into account the full scope of industry lobbying and its potential impact on congressional stock trading.

In response to the limitations noted in Eggers and Hainmueller (2014), this research introduces innovative solutions as follows:

First, to establish the industry-level correlation with congressional stock trading, the paper develops a new measurement that evaluates the similarity in NAICS code distribution between congresspersons’ stock portfolios and the committees they are assigned to, compared to those they are not. This is achieved using cross-entropy, revealing that the industry-level distributions in congresspersons’ stock trading closely resemble the industries of their assigned committees. This finding emphasizes that such industry-level information should not be omitted in analyses focused on understanding what influences congressional stock trading decisions.

Second, instead of employing a simplistic binary indicator, this study adopts the use of Graph Neural Networks (GNN). The GNN approach enables the neural network to learn the appropriate aggregation of information to accurately predict stock transactions between congresspersons and tickers. This method

involves the neural network continually updating the numerical representations of nodes and their aggregation within a heterograph that represents the complex network of congressional activities. These activities include which firms are lobbying on which bills, the assignment of bills to various committees, and the committee assignments of congresspersons. This heterograph is notably high-dimensional, with 55,700 nodes and 264,633 edges. Despite these daunting dimensions, the use of GNN in this study demonstrates that it is possible to successfully design predictive modeling over such a high-dimensional graph input.

In this study, a Graph Neural Network (GNN) model was trained to predict stock transactions between congresspersons and tickers, achieving an accuracy of 0.81 and an AUC-ROC score of 0.89. These results suggest that the information encapsulated in the heterograph, representing the complex network of congressional activities, is substantial in explaining congresspersons’ stock choices. An ablation study further evaluated the contributions of different edge types within this heterograph to the model’s predictive accuracy. By systematically removing specific edge types and calculating their Shapley values, it was revealed that a congressperson’s committee assignments and firm-level lobbying on bills were key to the model’s predictive performance. This finding contrasts with Eggers and Hainmueller (2014), who argued that there was no significant evidence to link firm-level lobbying on bills and a congressperson’s committee assignments with the composition of their stock portfolio.

Furthermore, to address the black-box nature of GNN predictions (Dayhoff and DeLeo, 2001; Buhrmester et al., 2021; Olden and Jackson, 2002), this research incorporates the use of GNNExplainer (Ying et al., 2019). This tool identifies the most critical nodes and edges that are pivotal in explaining why the model predicts the existence of a transaction (edge) between a congressperson and a ticker. Providing examples of this interpretative process, I will demonstrate how GNNExplainer brings greater transparency to the decision-making process of the GNN model, thereby reinforcing its trustworthiness and understandability.

Finally, I introduce a novel approach using a recently emerging Large Language Model (LLM) Agent (Shinn et al., 2023; Wang et al., 2023; Park et al., 2022), capable of performing prediction and theorization simultaneously. This method employs the LLM Agent for the same link prediction task as the GNN, focusing on identifying “buy-sell” transactional connections between legislators and stock tickers within the heterograph. The Agent not only predicts these links but also concurrently generates and refines hypotheses or theories that are generalizable to other predictive instances. This process mirrors human cognitive process, as the Agent endeavors to discern which information within the graph is crucial

for accurately predicting transactional relationships. Significantly, since all this process is conducted in natural language, it allows for the interpretation and understanding of the general principles produced by the Agent, enhancing the transparency and accessibility of its predictive reasoning.

In sum, this research emphasizes the crucial role of industry-level information, particularly as it flows through lobbying activities on bills and committee assignments, in shedding light on congressional stock trading, offering a perspective that contrasts with previous studies. Adopting Graph Neural Networks (GNN) for its ability to learn feature representations in a heterograph and understand information aggregation in networks representing congressional activities, this approach addresses GNN’s interpretability challenges by integrating GNNExplainer. Additionally, the innovative use of a Large Language Model (LLM) Agent for both prediction and theorization opens new avenues in political science, demonstrating the LLM’s potential as a reasoning module for theory building in this field.

## 2 Estimating Excess Returns of Congressional Stock Trading<sup>1</sup>

The broader literature on insider trading has long explored the information advantages that certain individuals, such as corporate insiders or well-connected investors, may possess when trading in the stock market. For example, Jeng et al. (2003) estimated returns to insider trading from a performance-evaluation perspective, while Ivković and Weisbenner (2005) studied the information content of the geography of individual investors’ common stock investments. These studies highlight the importance of understanding the impact of information asymmetry and potential insider trading in financial markets.

Despite the extensive research on insider trading in general, the application of this approach to congressional stock trading has been limited. In the context of congressional stock trading, previous studies, such as those conducted by Ziobrowski et al. (2004), Ziobrowski et al. (2011) and Eggers and Hainmueller (2013), have predominantly used calendar-time based portfolio approaches (Hoechle and Zimmermann, 2007) to estimate excess returns. This involves creating synthetic buy and sell portfolios (Shanken, 1996; Annaert et al., 2009) that mimic congresspersons’ stock purchases and sales but sell or buy such stocks after a year. This approach, however, neglects the importance of transaction timing, which is a crucial aspect of insider trading (Tahoun, 2014; Schweizer, 2011). It does not account for the short-term fluctuations in stock prices that congresspersons might anticipate based on their access

---

<sup>1</sup>Reproducible code for this section is available at <https://github.com/syyunn/efd/blob/main/anlys/cashout/fifo-rd-fed-ppssss-include-etf.py>

to privileged information. Congresspersons could potentially profit from these expected fluctuations by strategically timing their transactions using their privileged knowledge.

In addition, averaging excess returns across congresspersons may not capture the full extent of insider trading within the inner circle of Washington D.C. politics. Schweizer (2011) provided anecdotal evidence of politicians and their friends profiting from insider stock tips, while Lenz and Lim (2009) studied corruption and wealth accumulation in Congress, and Jerke (2010) and Bainbridge (2010) examined the use of political intelligence for insider trading with several anecdotes. These case studies suggest that certain congresspersons might engage in insider trading with specific firms or industries, which would be overlooked in an aggregate analysis.

In this section, therefore, I aim to address these limitations by estimating the excess returns at the congressperson-ticker level, with a focus on the life cycle of each buy/sell chain of specific tickers consecutively transacted by a congressperson. This approach offers a more granular analysis of potential insider trading among U.S. members of Congress, allowing us to better evaluate whether widespread insider trading exists at the congressperson-ticker level. By doing so, we build upon both the general insider trading literature and the existing research on congressional stock trading, contributing to a more comprehensive understanding of the potential information advantages leveraged by politicians in the stock market and the importance of transaction timing.

## 2.1 Data

To estimate excess returns at the congressperson-ticker level, I first needed to compile comprehensive data on the stock transactions of U.S. members of Congress. I obtained this data by scraping the Senate Financial Disclosure website<sup>2</sup>, which provides detailed information about the stock transactions made by congresspersons, including the date, ticker symbol, and the amount of each transaction.

The resulting dataset consists of 25,023 transactions, spanning a period from January 1, 2014, to August 5, 2022. These transactions involve 74 distinct Senators and 2,114 distinct tickers. Among these tickers, around 40% (832) are individual company-level tickers, such as AAPL for Apple Inc. and AMAT for Applied Materials Inc., while the remaining 60% (1,282) are ETFs or mutual funds, like QQQ for Nasdaq-100 index funds or IHI for U.S. Medical Devices ETF. This prevalence of industry-level security transactions suggests that Congresspersons often trade based on broader industry trends, rather than

---

<sup>2</sup><https://efdsearch.senate.gov/search/home/>



focusing solely on specific firms. This observation urges a recalibration of current literature (Ziobrowski et al., 2011, 2004; Eggers and Hainmueller, 2013, 2014), which tends to focus predominantly on firm-level stock trading behavior. Instead, it highlights the necessity of broadening our perspective, embracing a more comprehensive unit of analysis that extends beyond the firm-level, to include industry-level stock trades.

For each transaction, I added the Volume Weighted Average Price (VWAP) in USD acquired from a commercial stock data API <sup>3</sup>. VWAP is a widely used trading benchmark (Madhavan, 2002; Białkowski et al., 2008; Duffie and Dworczak, 2021) that represents the average price at which a security is traded throughout the day, weighted by the volume of each trade. By using VWAP, I obtained a representative price for each stock transaction, taking into account the varying trading volumes and prices during the entire trading day. However, the Senate Financial Disclosure data is range-censored in terms of the “amount”, which represents the value of the stock transaction for that date. The amount is reported as one of the following ranges in Table 1.

Amount Range (USD)
1,001 - 15,000
15,001 - 50,000
50,001 - 100,000
100,001 - 250,000
250,001 - 500,000
500,001 - 1,000,000
1,000,001 - 5,000,000
5,000,001 - 25,000,000

Table 1: Range of the min-max amount of each stock transaction.

RFC first_name	RFC last_name	RFC ticker	RFC asset_name	RFC trans_date	RFC amount_min	RFC amount_max	123 vwap
John	Hoeven	QCOM	QUALCOMM Incorporated	2017-03-02	100,001	250,000	[NULL]
David A	Perdue , Jr	[NULL]	Alliant Energy Corp CMN	2015-10-21	15,001	50,000	[NULL]
Benjamin L	Cardin	VO	Vanguard Mid-Cap ETF	2021-07-23	1,001	15,000	238.6192
Pat	Roberts	BAC	Bank of America Corporation	2018-07-05	1,001	15,000	279392
Patrick J	Toomey	IWF	iShares Russell 1000 Growth ETF	2021-01-14	1,001	15,000	241.4834
Timothy M	Kaine	ODVYX	Oppenheimer Developing Markets Y	2015-07-13	1,001	15,000	[NULL]
Kamala D	Harris	BSV	Vanguard Short-Term Bond ETF	2017-02-28	1,001	15,000	[NULL]
Steve	Daines	[NULL]	AMERICAN TAX EXEMPT BOND FUND	2014-10-22	1,001	15,000	[NULL]
Sheldon	Whitehouse	PANW	Palo Alto Networks, Inc.	2017-01-11	1,001	15,000	[NULL]
Mark R	Warner	ANGIX	Angel Oak Multi-Strategy Income Instl	2017-05-01	15,001	50,000	[NULL]
A. Mitchell	McConnell, Jr.	VFIAX	Vanguard 500 Index Fund Admiral Shares	2018-03-23	15,001	50,000	[NULL]
Sheldon	Whitehouse	TRBCX	T. Rowe Price Blue Chip Growth Fund	2018-05-07	15,001	50,000	[NULL]
Mark R	Warner	DBLTX	DoubleLine Total Return Bond Fund Class I	2018-06-01	15,001	50,000	[NULL]
Ron L	Wyden	BLL	Ball Corporation	2020-05-07	50,001	100,000	65.1487
John W	Hickenlooper	QRTEA	Qurate Retail, Inc. - Series A Common Stock	2021-05-10	250,001	500,000	14.2215
Mark R	Warner	AGG	iShares Core U.S. Aggregate Bond ETF	2021-03-05	1,001	15,000	114.2362
Christopher A	Coons	MSAIX	Invesco American Value Fund Class Y	2020-09-10	15,001	50,000	[NULL]
Robert J	Portman	WCMIX	WCM Focused International Growth Fund Institutional	2019-07-11	1,001	15,000	[NULL]
Christopher	Murphy	[NULL]	Aggressive Managed Allocation Age 4-7	2015-12-21	1,001	15,000	[NULL]
A. Mitchell	McConnell, Jr.	PRSCX	T. Rowe Price Science & Tech	2017-12-15	1,001	15,000	[NULL]
Rick	Scott	SHV	iShares Short Treasury Bond ETF	2019-03-21	250,001	500,000	110.4938

Figure 1: **Senator’s Stock Transactions Data (Compiled)** The table shows the compiled stock transactions data which includes the name, ticker, date, amount min/max, and VWAP for each transaction.

<sup>3</sup><https://polygon.io/stocks>

It is important to note that not all of the transactions have a clear ticker because some assets are not publicly traded on an exchange (See “NULL” values in ticker field in Figure 1). Additionally, not all transactions have VWAP values, as not all tickers or asset names have available stock price data from the data provider (See “NULL” values in vwap column in Figure 1). This may lead to some limitations in the analysis, but the dataset still provides a rich source of information for understanding potential insider trading among U.S. members of Congress. An excerpt of a few rows of such compiled transaction data is provided in Figure 1.

## 2.2 Uncanny Timing of Congressional Stock Trading

Firstly, I gained insights into the mechanisms behind their trading decisions by reviewing news articles. For example, there were several media reports<sup>45</sup> regarding Ron Wyden’s semiconductor stocks trading. I searched for Senator Ron Wyden’s stock transactions with a NAICS code beginning with 334, which indicates computer and electronic product manufacturing. I found that three different tickers (AMAT, AVGO, KLAC) of the transactions that met this condition have a commonality in that they all started on the same date, April 6th, 2020, and ended on either April 6th or April 16th, 2021. Furthermore, all of them follow a similar pattern of multiple purchases followed by sales after certain critical points, such as *Purchase – Purchase – ... – Purchase | Sales – Sales – ... – Sales* as shown in Fig 2.

On April 1st, 2021, President Biden announced a plan to invest \$50 billion to boost the U.S. chip industry<sup>6</sup>. After this announcement, Senator Ron Wyden sold all of his semiconductor stocks. This suggests that members of Congress may have access to not only legislative information but also the publicization of such information that can potentially move the stock market beforehand. This enables them to not just design their portfolio, but also determine when to buy and when to sell, with some anticipation of specific events and their impact on the market.

As shown in the example of Ron Wyden, importance of timing in the context of political insider trading is emphasized by scholars like Tahoun (2014) and Schweizer (2011). Accordingly, this research proposes to integrate such timing considerations into the analysis of excess returns. Specifically, we will

---

<sup>4</sup>Theo Wayt, “US Sen. Ron Wyden boosts chipmakers while his wife buys their shares”, New York Post, May 20, 2021, <https://nypost.com/2021/05/20/us-sen-ron-wyden-boosts-chipmakers-while-his-wife-buys-their-shares/>

<sup>5</sup>Alicia Parlapiano, Adam Playford, and Kate Kelly, “These 97 Members of Congress Reported Trades in Companies Influenced by Their Committees”, The New York Times, Sept. 13, 2022, <https://www.nytimes.com/interactive/2022/09/13/us/politics/congress-members-stock-trading-list.html>

<sup>6</sup>Alex Leary and Paul Ziobro, “Biden Calls for \$50 Billion to Boost U.S. Chip Industry”, The Wall Street Journal, March 31, 2021, <https://www.wsj.com/articles/biden-urges-50-billion-to-boost-chip-manufacturing-in-u-s-11617211570>

	ABC fi	ABC la	ABC ticker	ABC trans_type	ABC trans_date	ABC trans_type	123 amount_min	123 amount_max
1	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
2	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
3	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
4	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
5	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
6	Ron L	Wyden	AVGO	Purchase	2020-04-06	Purchase	15,001	50,000
7	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
8	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
9	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
10	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
11	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
12	Ron L	Wyden	AVGO	Purchase	2020-06-04	Purchase	15,001	50,000
13	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
14	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
15	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
16	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
17	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
18	Ron L	Wyden	AVGO	Purchase	2020-06-23	Purchase	15,001	50,000
19	Ron L	Wyden	AVGO	Purchase	2021-03-04	Purchase	1,001	15,000
20	Ron L	Wyden	AVGO	Purchase	2021-03-04	Purchase	1,001	15,000
21	Ron L	Wyden	AVGO	Purchase	2021-03-04	Purchase	1,001	15,000
22	Ron L	Wyden	AVGO	Purchase	2021-03-04	Purchase	1,001	15,000
23	Ron L	Wyden	AVGO	Purchase	2021-03-04	Purchase	1,001	15,000
24	Ron L	Wyden	AVGO	Sale (Partial)	2021-03-30	Sale (Partial)	1,001	15,000
25	Ron L	Wyden	AVGO	Sale (Partial)	2021-03-30	Sale (Partial)	1,001	15,000
26	Ron L	Wyden	AVGO	Sale (Partial)	2021-03-30	Sale (Partial)	1,001	15,000
27	Ron L	Wyden	AVGO	Sale (Partial)	2021-03-30	Sale (Partial)	1,001	15,000
28	Ron L	Wyden	AVGO	Sale (Partial)	2021-03-30	Sale (Partial)	1,001	15,000
29	Ron L	Wyden	AVGO	Sale (Full)	2021-04-06	Sale (Full)	100,001	250,000
30	Ron L	Wyden	AVGO	Sale (Full)	2021-04-06	Sale (Full)	100,001	250,000
31	Ron L	Wyden	AVGO	Sale (Full)	2021-04-06	Sale (Full)	100,001	250,000
32	Ron L	Wyden	AVGO	Sale (Full)	2021-04-06	Sale (Full)	100,001	250,000
33	Ron L	Wyden	AVGO	Sale (Full)	2021-04-06	Sale (Full)	100,001	250,000

Figure 2: **Senator Ron Wyden’s stock transactions for Broadcom Inc. (Ticker: AVGO)**  
The transactions exhibits a pattern of multiple purchases followed by sales after certain critical points, spanning from April 6th, 2020 to April 6th, 2021.

adopt an approach that recognizes the life-cycle of transactions, starting from consecutive purchases and ending with consecutive sales. This methodology will be detailed further in Section 2.3, ensuring that our analysis is cognizant of both the decision to invest in specific stocks and the timing of these decisions.

### 2.3 Purchases-then-Sales: Sub-sequences of Congressional Stock Transactions

Based on the observation introduced in Section 2.2, I partitioned each transaction sequence into sub-sequences, where each sub-sequence consists of consecutive purchase transactions followed by consecutive sale transactions, all arranged in chronological order as illustratively shown in Figure 3. This kind of sub-sequence partitioning is based on the assumption that if congressional investments involve insider trading—using privileged knowledge—there should be a timing of both the beginning and end of the investment that is driven by a certain event (Cziraki et al., 2021; Sivakumar and Waymire, 1994).

Specifically, the event of interest would be one that, upon being publicized, moves the stock market into a different phase. In the case of insider trading, we would expect to see a pattern where a congressperson accumulates a long position in a stock ahead of a positive event and subsequently monetizes that position by selling the stock after the event becomes public and positively impacts the stock price. Conversely, a congressperson may sell a stock ahead of a negative event and avoid losses when the event becomes public and negatively impacts the stock price.

It is important to note that in this analysis, we are only considering the case of congresspersons taking long positions and subsequently selling those positions, as this is the type of transaction that is reported in Financial Disclosure reports. In these reports, there are no “stock-shorting” transactions, which involve betting against a stock and profiting from its decline. As such, our partitioning approach focuses on identifying sub-sequences of consecutive purchases followed by consecutive sales, which may reflect the use of privileged knowledge to take advantage of market-moving events and realize profits from long positions.

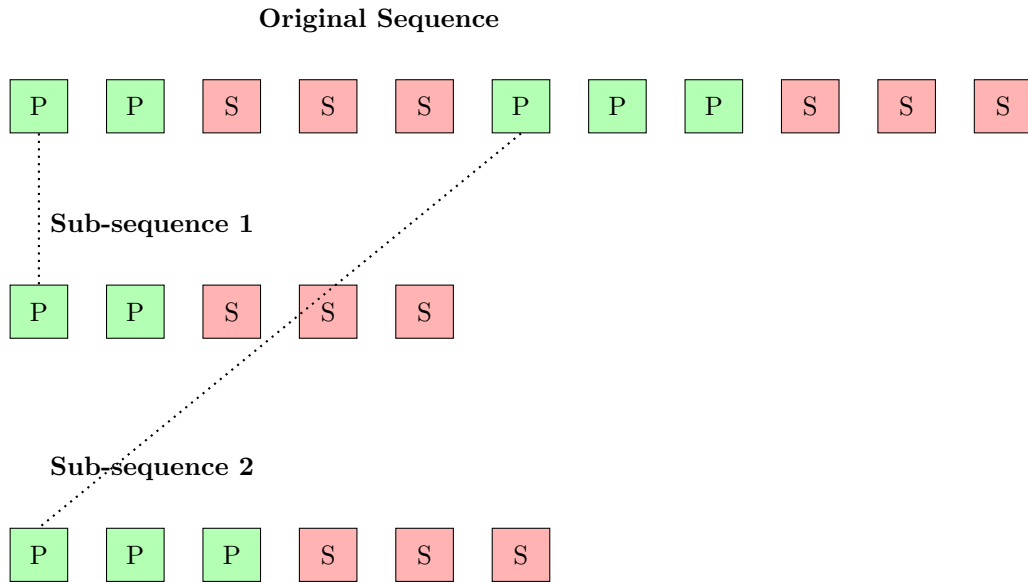


Figure 3: Partitioning of transaction sequences into sub-sequences based on consecutive purchase and sale transactions.

Through the partitioning process, I obtained a total of 435 sub-sequences spanning across 358 unique combinations of Senator-Ticker pairs. Each sub-sequence represents a long position taken by a senator in a specific stock and is characterized by a start date and an end date. The start date corresponds to the date of the first purchase transaction in the sub-sequence, and the end date corresponds to the date of the last sale transaction in the sub-sequence. The duration of each sub-sequence, measured in days,

represents the length of time the senator held the long position. On average, senators held such long positions for approximately 340 days, and around 65% of these positions were held for less than a year. The frequencies of durations for all subchains are illustrated in Figure 6.

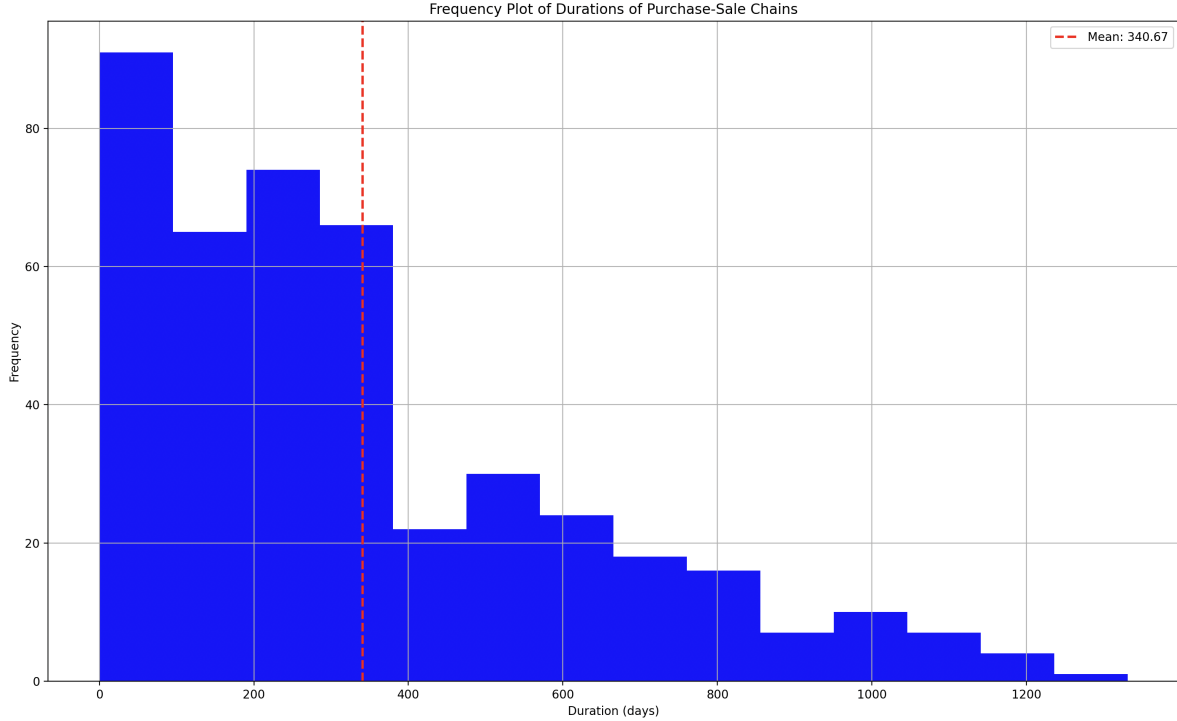


Figure 4: **Frequencies of durations of long positions held by congressmen at the Senator-Ticker level** The durations are measured in days and represent the length of time between the start and end dates of each subchain. The mean duration of holding such long positions is approximately 340 days. Notably, around 65% of these long positions are held for less than a year

## 2.4 Estimating Excess Returns of Sub-sequences

Estimating the excess return of the 435 sub-sequences, which were acquired following the procedure described in Section 2.3, presented a methodological challenge due to the nature of the Finance Disclosure data. The data provides only the “minimum and maximum” range of amounts spent on purchasing or selling each ticker on a specific day as illustrated in Table 1, rather than exact transaction amounts. To address this challenge and estimate the excess returns for each Purchase-Sale sub-sequence, the following approach was taken:

1. **Random Sampling of Transaction Amounts:** For each transaction (purchase or sale) within a sub-sequence, an amount was randomly sampled from a uniform distribution with support equal to the minimum and maximum range of the transaction amount provided in the data.

**2. Estimation of Shares Bought or Sold:** The sampled amount was divided by the volume-weighted average price (VWAP) of the stock on the corresponding transaction date to estimate the number of shares bought or sold by the congressperson.

**3. Creating Settled Pairs:** Within each sub-sequence, settled pairs of buy-sell transactions are identified. A settled pair consists of one unit of a buy transaction matched with one unit of a subsequent sell transaction. The pairing process is based on a first-in and first-out principle, meaning that stocks purchased earlier are matched first to sales, before those bought later. This ensures that the sale always occurs after the purchase. Multiple settled pairs can be created within a single sub-sequence.

**4. Computing Profit Return Rate for Each Settled Pair:** For each settled pair, the profit return rate is calculated as the relative profit or loss from the buy-sell transaction. The profit return rate is computed using the formula:

$$\text{Profit Return Rate} = \frac{\text{Sale Price} - \text{Purchase Price}}{\text{Purchase Price}} * 100$$

where “Sale Price” is the price at which the stock was sold, and “Purchase Price” is the price at which the stock was purchased.

**5. Penalizing the Profit Return Rat with Fed Reserve Rate:** The profit return rate for each settled pair is then penalized by the average Federal Reserve Rate during the holding period of that specific pair. The holding period is defined as the time interval between the purchase date and the sale date of the settled pair. The penalized return, or “excess return” for each pair is calculated as:

$$\text{Excess Return} = \text{Profit Return Rate} - \text{Average Federal Reserve Rate}$$

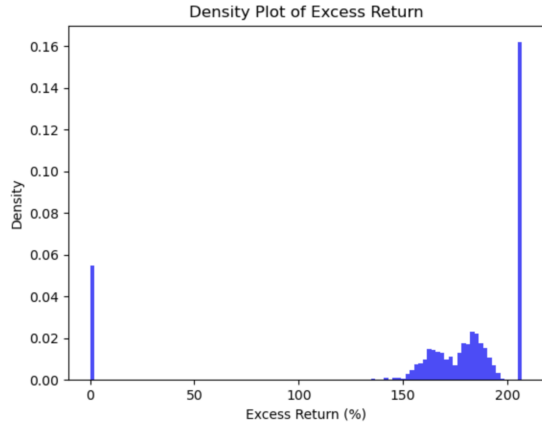
The Average Federal Reserve Rate represents the risk-free rate of return that could have been earned from a risk-free investment during the same holding period.

**6. Averaging Excess Returns:** The final excess return for the entire sub-sequence is computed by averaging the excess returns of all the individual pairs of settled buy-sell transactions within the sub-sequence. This approach provides a comprehensive measure of the excess return for the sub-sequence, accounting for profit ratio above the risk-free rate of return for each holding period.

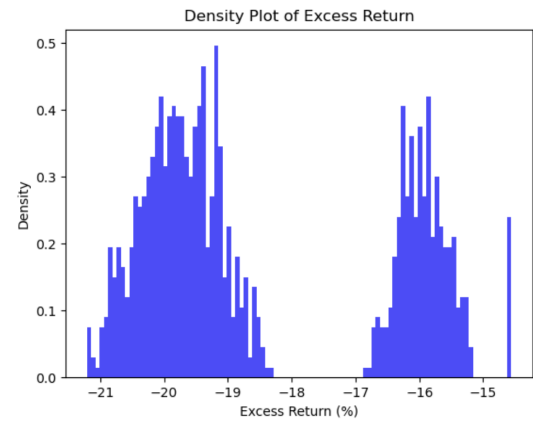
By following this approach, the excess return of each sub-sequence was estimated while accounting

for the limitations of the available data. Using the Federal Reserve Rate as the baseline for comparison is a more conservative approach because it represents a risk-free rate of return (Bauer and Rudebusch, 2014; Sarno and Thornton, 2003) that is typically higher than the interest rates offered by most savings accounts. As a result, the excess return is measured against a higher baseline, potentially lowering the final estimation result and providing a more cautious assessment of the excess return earned by the congressperson.

To assist in understanding the concepts explained previously, I am presenting the estimated excess return distributions for Senator Ron Wyden’s sub-sequences involving two different companies: Applied Materials Inc. (AMAT), which provides manufacturing equipment, services, and software to the semiconductor industry, and Marriott International Inc. (MAR), a global hotel brand. These distributions were computed using the random sampling method explained earlier, where the randomness is inherited from the uniform random sampling of transaction amounts from the provided minimum and maximum ranges. It is important to note that each sub-sequence is uniquely identified not only by the congressperson-ticker level but also by the start and end dates of the sub-sequence.



(a) Ron Wyden’s excess returns from transactions involving Applied Materials Inc. (AMAT) from April 2020 to April 2021.



(b) Ron Wyden’s excess returns from transactions involving Marriott International Inc. (MAR) from May to August 2020.

Figure 5: Estimated Excess return distributions of Senator Ron Wyden’s transactions for AMAT and MAR

In Figure 5, the mean of the excess return distributions for Senator Ron Wyden’s sub-sequences involving AMAT (Applied Materials Inc.) and MAR (Marriott International Inc.) are 166.30% and -18.43%, respectively. As we can see, even the same senator sometimes achieves great excess returns while also experiencing failures. In a similar vein, I collected the mean values of the excess return distributions for all 435 subsequences, which are presented in Figure 6.

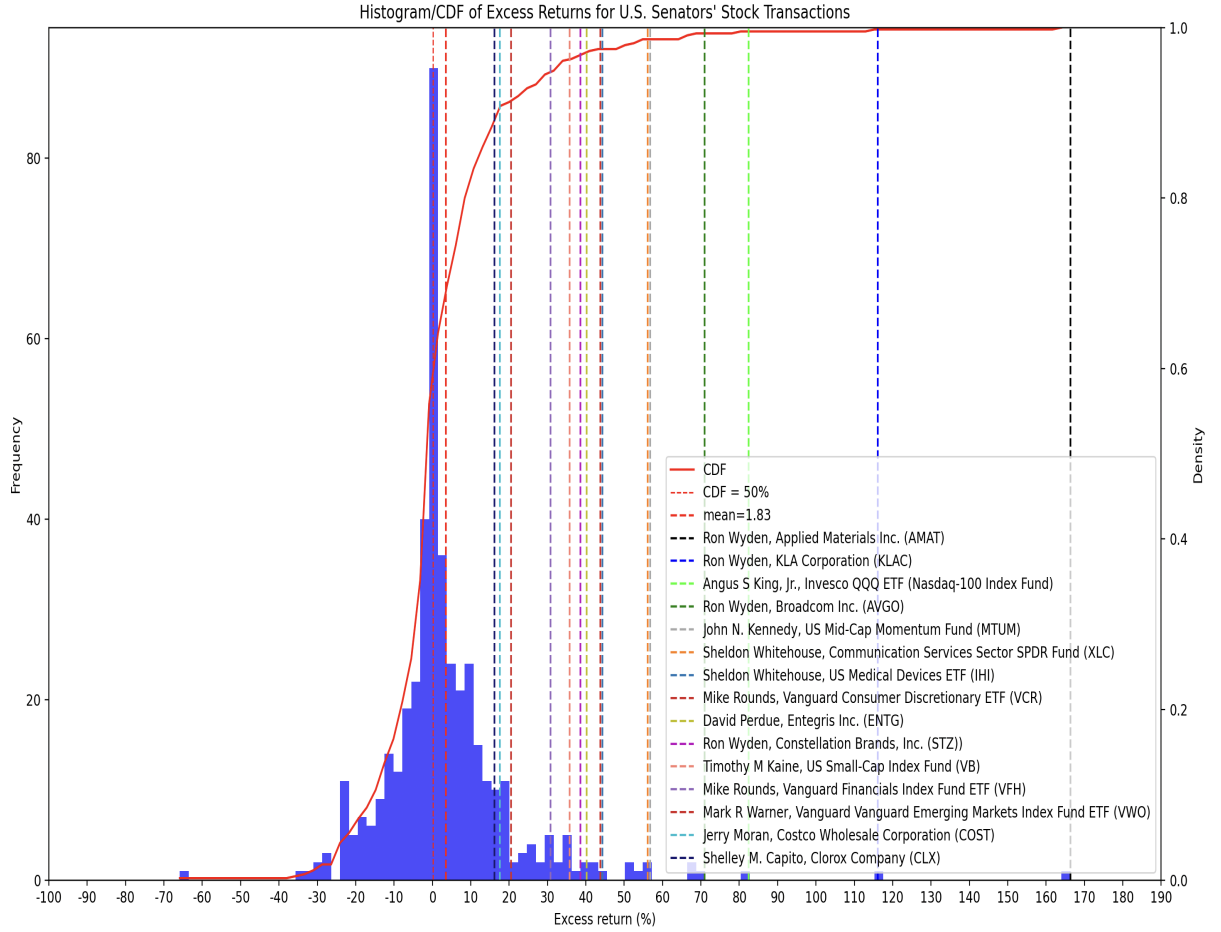


Figure 6: Distribution of Senators' Mean of Estimated Excess Returns Distribution.

In Figure 6, among the mean estimated excess returns, Ron Wyden's semiconductor-related stocks like Applied Materials Inc (AMAT), KLA Corp. (KLAC), and Broadcom Inc. (AVGO) are highly ranked, scoring from 80% to 166% of excess returns. These transactions have already been spotlighted by the media. This suggests that this method can reveal such dubious transactions spotlighted by the media, ranking them as acquiring high-performing excess returns.

What is noticeable is that the cumulative density of the excess return distribution reaches 0.5 when the excess return is 0. It means that not all transactions of Senators are always successful, but it's more like random whether they actually acquire positive excess profit. This finding aligns more with the results from Eggers and Hainmueller (2013) than those of Ziobrowski et al. (2004) and Ziobrowski et al. (2011), suggesting that Senators are more like mediocre investors who align with the literature about failing individuals as investors, as explained in broad finance literature such as Barberis and Thaler (2003), and Barber and Odean (2000).

However, one thing that is important to notice is that the distribution in Figure 6 scores a skewness of



2.804, which means the tail on the right side of the distribution is longer. This indicates that compared to the case of Senators losing money, at least in situations where they are acquiring excess return from it, they may be more related to some privileged knowledge that can back up the performance of such stock transactions. The presence of the long tail suggests that, in some cases, Senators might be involved in transactions that benefit from privileged knowledge, as Eggers and Hainmueller (2014) suggest, which could be originating from their political connections, including connections established through Political Action Committee (PAC) donations or district-level affiliations with the firms in question.

While the current finding of significant outliers with excess returns suggest possible privileged-information driven stock trading at the congressperson-ticker level, it is essential to delve deeper into the fundamental relationship between congressional activities and stock transactions. In order to gain a better understanding of the underlying factors that drive congressional stock trading behavior and the potential impact of privileged information on investment decisions among lawmakers, we need to explore an approach that quantifies the predictive power embedded in the congressional activities and assesses the extent to which these activities are associated with stock transactions.

It is worth noting that this approach is not entirely novel, as Eggers and Hainmueller (2014) have already studied the connections between congressional-related activities and the specific firm's stock transactions of a congressperson. Their research examined factors such as PAC contributions, lobbying and geographical connections based on district, and congressperson's committee membership and firm-level lobbying, to determine whether these factors could predict a congressperson's stock transactions.

Given that the innate difficulty in identifying the intention behind such transactions (Ziobrowski et al., 2011, 2004; Eggers and Hainmueller, 2013, 2014), due to the limited information available or the more fundamental challenge of distinguishing between a congressperson's private and public life (Buchanan and Tollison, 1984), it is essential to examine how dynamically these transactions are connected with congressional activities in terms of information. One possible avenue for this exploration is to represent congressional activities as a graph, which can effectively capture the complex relationships between various actors and actions (Henaff et al., 2015; Kaushik et al., 2002) within the legislative process. Graph-based representations are well-suited for modeling the interconnected nature of congressional activities, taking into account not only the individual actions of Congress members but also the broader context of committee memberships, lobbying efforts by firms on specific bills of their interest, the referral of bills to specific committees, and the assignment of Congresspersons to certain committees.

Eggers and Hainmueller (2014) studied the impact of such factors, particularly committee membership and firm-level lobbying on bills, on stock transactions at a binary level, considering whether or not this information existed for each congressperson-stock pair. However, congressional activities are more complex and interconnected, with various entities involved in these relationships simultaneously rather than in isolation. For example, multiple firms in the semiconductor industry, such as Intel, Qualcomm, Broadcom, Apple, and IBM, participate in lobbying efforts for bills related to their sector, like the CHIPS Act (H.R.4346 117th Congress) or FABS Act (S.2107 117th Congress). These activities are governed by specific congresspeople within particular committees, and all this information collectively can form the detailed context in which a congressperson transacts stock.

In addition, as explained in Section 2.1, a congressperson’s securities transactions are not limited to individual firm levels. In fact, 60% of these transactions involve exchange-traded funds (ETF) or mutual funds that target a wide range of specific industries such as wireless communication, medical devices, or mid-cap or small-cap companies. Therefore, the full context of a congressperson’s stock transactions extends beyond individual companies to encompass broader industry trends and movements.

In light of this, the next section will introduce a newly compiled dataset that captures congressional activities as a whole, in the form of graph-structured data. I will then demonstrate how this graph-structured data can be useful, for example, by directly computing the similarity between a committee’s industry-level specialization and the industry-level distribution of a congressperson assigned to that committee in Section 4. Additionally, I will present a method for modeling the predictive task, which can directly take graph-structured data as input using Graph Neural Networks in Section 5. An array of analyses using graph-structured data will enhance our comprehension of the intricate relationships between congressional activities and stock transactions. This approach will offer more profound insights into the potential influence of privileged information acquired through congressional activities on lawmakers’ investment decisions.

### 3 Graph-Structured Data for Representing Congressional Activities<sup>7</sup>

---

<sup>7</sup>Reproducible code for this section is available at <https://github.com/syyunn/gnnex/blob/main/data/graph.ipynb>

The data utilized in the following sections forms a large, complex network that is categorized as a heterograph. This structure captures congressional activities through different types of nodes and edges, thereby encapsulating the multi-faceted nature of these activities. This heterograph encompasses information on congressional activities, such as committee assignments, bills being lobbied by firms, bill assignments to committees, and firms classified under specific NAICS codes. The detailed specifications of the node types can be found in Table 2, while the edge types are described in Table 3. Different types of nodes and their relationships, captured by different types of edges, are provided in Figure 7. The process of data collection from disparate sources and the subsequent disambiguation and merging of entities are elaborated upon in Appendix A. Additionally, Appendix B provides a detailed explanation of a more modern approach to extracting structured data from collected financial disclosure PDFs. In this approach, I utilized a Large Language Model (LLM) to aid in the extraction process. The specifics of how the LLM was employed are discussed in detail within the appendix.

Table 2: Heterograph (Nodes)

Node Type	N	Period	Source
Firm (Ticker)	4,202	-	Lobbyview & Finance Disclosure
Bills	47,767	110-117th Congress	Lobbyview
Congressperson	2,431	113-118th Congress	Lobbyview & Finance Disclosure
Committee	556	-	Lobbyview
NAICS code	744	-	naics.com
Total	55,700	-	-

Table 3: Heterograph (Edges)

Edge Types	N	Period	Source
Congressperson- Buy/Sell- Firm (Ticker)	24,675	[2013-01-24, 2023-03-08]	Finance Disclosure
Firm (Ticker) - Lobby On - Bill	148,487	[2016-01-02, 2022-02-24]	Lobbyview
Ticker- Classified as - NAICS Codes	4,147	-	Finance Disclosure & naics.com
Bill- Referred to - Committee	75,626	[2016-01-05, 2021-12-17]	Lobbyview
Congressperson- Assigned to - Committee	11,698	115-117th Congress	Finance Disclosure & Lobbyview
Total	264,633	-	-

To provide a more concrete understanding of the data, Figure 8 displays a subgraph related to Senator Ron Wyden’s transaction in Trip Advisor stock (Ticker: TRIP). This subgraph illustrates the relationships between Senator Ron Wyden’s congressional activities, including his membership in the Senate Finance Committee, his involvement with a specific bill related to airport improvements, and the economic sectors represented by NAICS codes, thereby providing insights into how these activities could potentially influence or be influenced by his stock transactions.

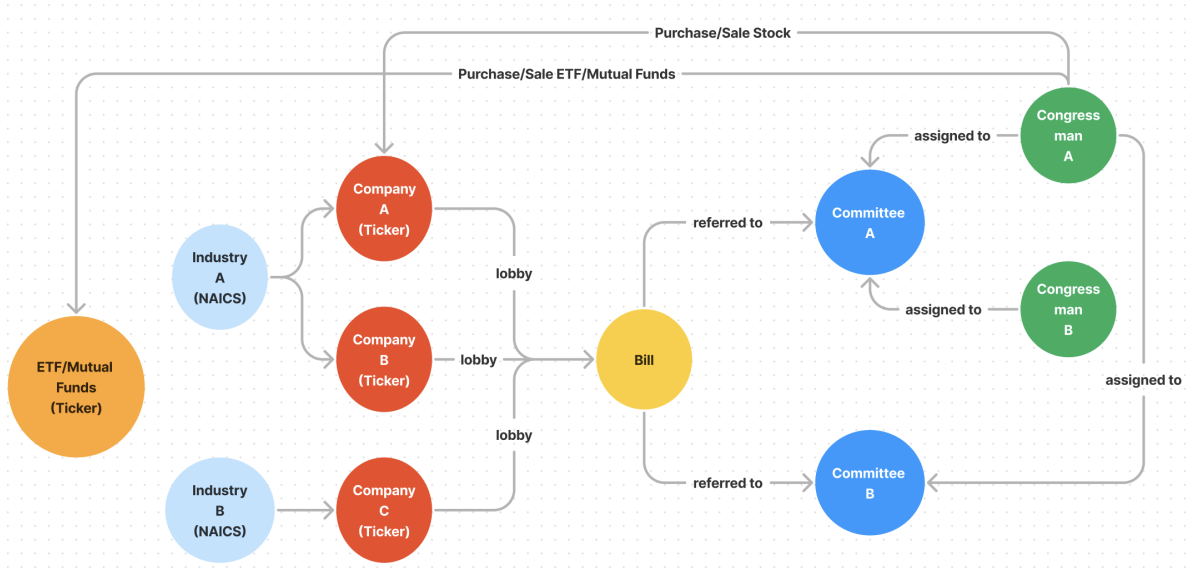


Figure 7: **Illustration of Heterograph** The graph-structured data used in this study includes various types of nodes and edges that represent different entities and interactions within the congressional activities and investment behavior of Congresspersons.

## 4 Industry-level Similarities between Congresspersons and Committees<sup>8</sup>

In this section, I aim to examine the relationship between committee assignments and stock trading behavior among congress members, addressing the puzzle arising from Eggers and Hainmueller (2014)’s findings. Despite their discovery of the role that political connections play in profitable transactions, they found no evidence that committee membership influenced investment decisions. This observation is strikingly contrary to the preponderance of literature highlighting the importance of committee assignments and the specialized knowledge they confer.

Numerous studies, such as Patterson (1970), King (1994), and Asher (1974b), have focused on the role of committee assignments in shaping legislative outcomes, the impact of bill referral on committee specialization, and members’ specialization in topics related to committees’ jurisdiction, respectively. Further research reinforces this notion of committee assignments as platforms for leveraging congressperson’s knowledge and expertise, as shown in Boros and Fenno (1968); Gilligan and Krehbiel (1989); Kiewiet and McCubbins (1991); Krehbiel (1992); Curry (2019). The results of Egger and Hainmueller (2014) thus present a unique counterpoint, as they seem to contradict this prevailing understanding of committee

<sup>8</sup>Reproducible code for this section is available at <https://github.com/syyunn/efd/blob/main/anlys/cycle/main9-transactions-desc-house-included.ipynb>

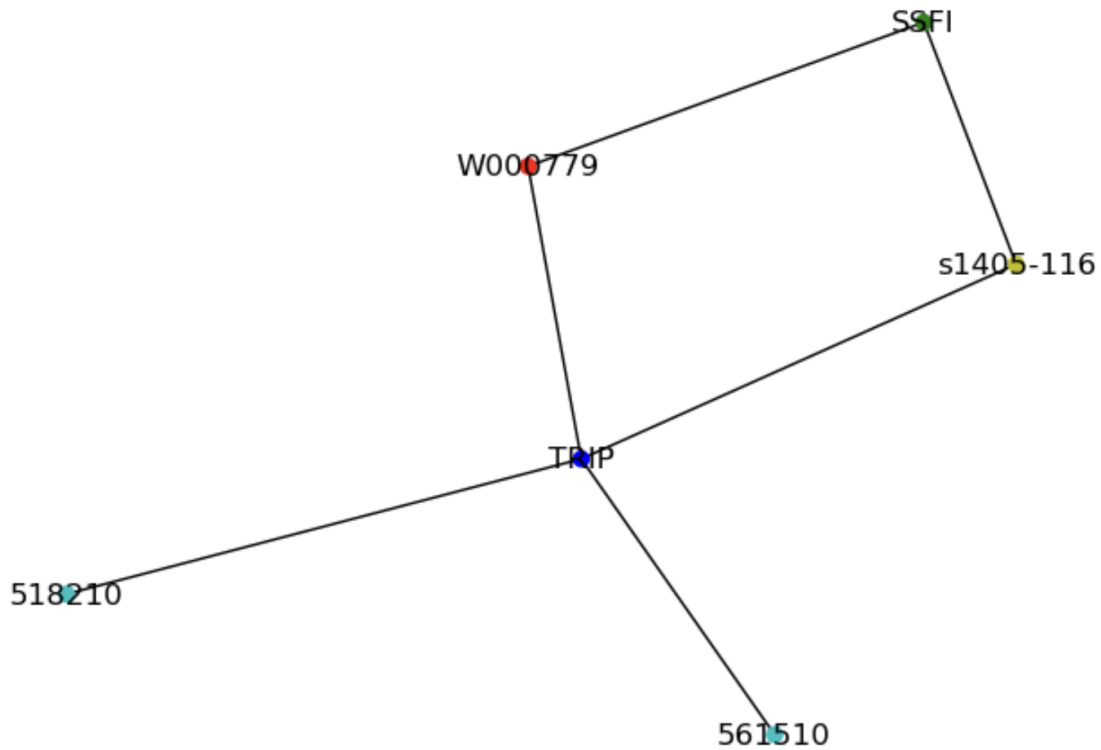


Figure 8: **A subgraph illustrating the congressional network related to the transaction of Senator Ron Wyden’s Trip Advisor stock.** The node labeled W000779 corresponds to Ron Wyden’s bioguide-id, which is a unique identifier provided by Congress for each senator. SSFI represents the Senate Finance Committee, of which Ron Wyden is a member. S1405-116 is a bill in the 116th Congress that revises requirements for the airport improvement program and pilot program for passenger facility charges at nonhub airports. The node labeled 518210 represents the NAICS Code for Data Processing, Hosting, and Related Services, while 561510 represents Travel Agencies.

roles in the legislative process.

Committee assignments provide congresspersons with unique access to information and resources related to specific industries and policy areas. As members of these committees, they are privy to the latest policy developments (Price, 1978), market trends, and regulatory changes (Weiss, 1989) that could potentially affect the performance of companies in the industries they oversee. The specialized knowledge and insights gained from participating in committee activities could inform their investment decisions and potentially influence their stock trading behavior. For example, they might be more inclined to invest in companies within their committee’s jurisdiction due to their deeper understanding of the industry dynamics and future prospects. Furthermore, their committee roles could enable them to establish connections with industry stakeholders and gain access to non-public information that could potentially offer them an edge in making investment decisions.

This section aims to provide a more robust statistical examination of the correlation between committee specialization and the stock trading patterns of Congress members. Specifically, it probes the degree to which committee specialization mirrors the industry distribution of a Congress member’s stock portfolio. This industry-focused analysis is justified by several observations and hypotheses:

1. As evidenced in Section 2.1, approximately 60% of the stock transactions by Senators involve ETFs or Mutual Funds. This indicates that congressional trading doesn’t merely operate at the firm-level but extends to the industry-level, suggesting a possible connection between industry-specific knowledge and transaction behaviors.

2. On the data side, the graph-structured data introduced in Section 3 allows for the direct aggregation of industry-level committee specialization by tracing the link from firms lobbying on bills to the committees these bills are assigned to.

3. As shown in Figure 6, many of the investments with high excess returns are industry-specific ETFs or Mutual funds. For instance, Senator Ron Wyden’s collection of semiconductor stocks like AMAT, AVGO, and KLAC; Senator Mike Rounds’ Financial Sector ETF; and Senator Sheldon Whitehouse’s Medical Device ETFs. These observations underscore the need for an industry-level analysis that goes beyond firm-level stock trading.

In addition, the legal framework also favors an industry-level analysis. Following the STOCK Act 2012 and the corresponding Senate Ethics manual (Boxer et al., 2012), Congress members are prohibited from trading based on information obtained through their official duties. This suggests that leveraging not just firm-specific, but industry-level information garnered from congressional activities for trading is legally restricted. This study will hence scrutinize whether there exists a discernible pattern between committee specializations and industry-level stock transactions.

## 4.1 Measuring Industry-level Specialization

In this subsection, I will discuss the measurement of committee specialization in terms of industry-level specialization. By aggregating NAICS codes for the bills lobbied by various industries and those bills being referred to specific committees, I aim to quantify the industry-level specialization of each committee.

As an example, we can expect the Senate Banking Committee to have a higher degree of jurisdiction over banking-related issues, such as regulations tied to LIBOR (London Interbank Offered Rate) rates.

This, in turn, would influence firms associated with NAICS codes related to banking (e.g., 52) to lobby the bills assigned to this committee more actively. This industry-level specialization can be effectively captured using the graph-structured data discussed in Section 3.

To capture this industry-level specialization, I aggregate the North American Industry Classification System (NAICS) codes of each firm that is lobbying on each bill. These codes categorize each firm according to its primary business activity. I then look at the bills assigned to each committee. This allows to identify which industries - as represented by their NAICS codes - are most active in lobbying bills that fall under the purview of specific committees. For instance, if a significant number of firms with the NAICS code for banking (52) are lobbying on bills assigned to the Senate Banking Committee, this would suggest a high degree of industry-level specialization of Senate Banking Committee on financial industry. This method of analysis, leveraging graph-structured data, allows us to explore the relationships between committee assignments, industries, and lobbying activities.

To represent the committee-level specialization, I create a discrete probability distribution for each committee by aggregating the NAICS code distributions of firms that lobby bills assigned to those committees. Specifically, I count the occurrences of each NAICS code associated with firms that lobby bills referred to a particular committee. This method highlights the committee's industry focus, as it captures the concentration of lobbying efforts by firms within specific industries. Once we have the count-based frequency plot for each committee, we can easily convert it into a probability distribution function (PDF) by normalizing the counts with the total occurrences of all NAICS codes.

Similarly, to aggregate the NAICS code distribution of firms involved in each congressperson's stock transactions, we can simply count the occurrences of NAICS codes associated with all stock transactions executed by the congressperson (both purchases and sales). This provides a clear picture of the industries in which they invest, as the NAICS codes are derived from the firms whose stocks are being bought or sold by the congressperson in their stock transactions. We then normalize the count-based frequency plot with the total occurrences of all NAICS codes to obtain a PDF representing the congressperson's industry preferences.

As we've seen in Section 2.4, the data used for estimating Senator's excess return shows that around 60% of tickers are Exchange-Traded Funds (ETF) or mutual funds, which are not single firms but representing certain industries or the stock market in general. For example, in Figure 6, one of the highest excess returns, like the 45% profit shown by Sen. Sheldon Whitehouse, is the ticker IHI, which

is an ETF that invests in the US medical device market. Sen. Sheldon Whitehouse supported the Biden plan to fully utilize the Defense Production Act, increase the supply of necessary medical equipment and supplies<sup>9</sup>. As shown, Senators do not always reflect their knowledge gained from congressional activities at the firm level but at the industry level. In either case, this measurement can effectively capture their level of specialization in certain industries in terms of their transaction patterns. By comparing the similarity between the two distributions, one from the committee and one from the congressperson, we can statistically test how similar they are to each other.

Subsequently, we measure the similarity between a congressperson’s stock transactions’ industry preference and a committee’s industry specialization. This similarity can be quantified using cross entropy. Cross-entropy is a useful statistical tool for determining the similarity between two distributions (Wu et al., 2018; Mao et al., 2013), making it an ideal choice for comparing the distributions of NAICS codes for committee assignments and stock transactions. In this regard, for a given congressperson  $i$  and committee  $k$ , the cross entropy  $H_{ik}$  is computed as:

$$H_{ik} = - \sum_j P_{i,j} \log Q_{k,j}$$

Here,  $P_{i,j}$  refers to the density of congressperson  $i$ ’s trades in industry  $j$  relative to all their transactions, and  $Q_{k,j}$  denotes committee  $k$ ’s specialization in industry  $j$  measured in the probability density.

A lower cross entropy  $H_{ik}$  suggests a higher similarity between the industry preference of congressperson  $i$ ’s stock transactions and the industry specialization of committee  $k$ . This value is calculated at the congressperson-committee level, thereby providing an industry-level similarity measure between them.

Figure 9 provides an example of this measurement for Sen. Sheldon Whitehouse’s case. It displays the NAICS code distributions for Sen. Whitehouse’s stock transactions, the Senate Finance Committee (SSFI), and the Senate Banking Committee (SSBK). The figure illustrates that the distribution of Sen. Whitehouse’s stock transactions resembles the distribution of the Senate Finance Committee more closely than that of the Senate Banking Committee, highlighting the connection between his transactions and his committee assignments.

In Figure 9, we can calculate the cross-entropy between Sen. Whitehouse’s stock transaction distribution and the distributions of Senate Finance Committee (SSFI) and Senate Banking Committee

---

<sup>9</sup>See <https://www.whitehouse.senate.gov/news/release/whitehouse-supports-biden-plan-to-fully-utilize-defense-production-act-increase-supply-of-necessary-medical-equipment-and-supplies>



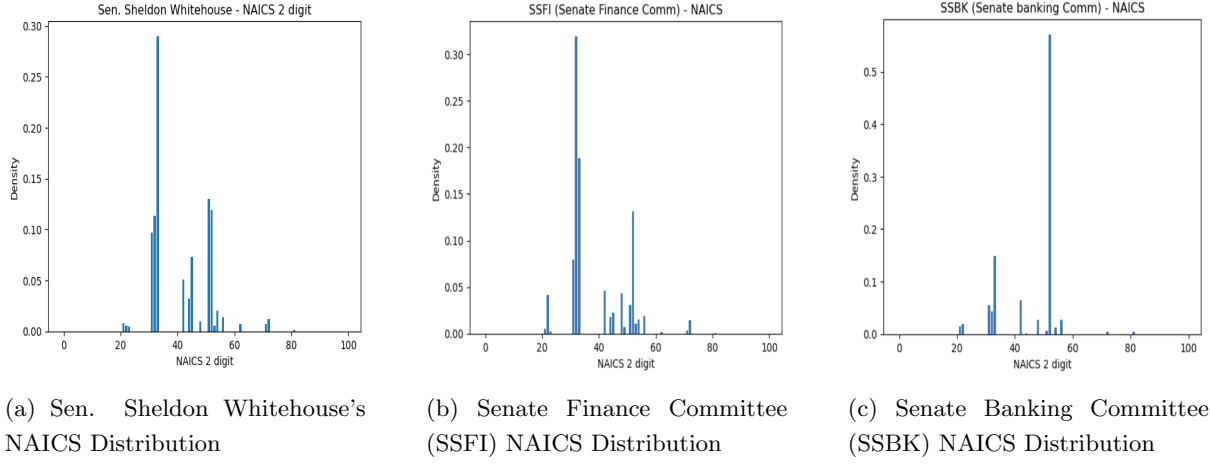


Figure 9: Comparison of NAICS code distributions for Sen. Sheldon Whitehouse, Senate Finance Committee (SSFI), and Senate Banking Committee (SSBK). The figure illustrates how the distribution of Sen. Whitehouse's stock transactions resembles the Senate Finance Committee's distribution more closely than that of the Senate Banking Committee.

(SSBK). The results are as follows:

$$\text{Cross entropy (Sen. Sheldon Whitehouse, SSFI)} = 0.816$$

$$\text{Cross entropy (Sen. Sheldon Whitehouse, SSBK)} = 3.311$$

These values indicate that Sen. Sheldon Whitehouse's investment portfolio is more similar to Senate Finance Committee in terms of NAICS code distribution, as a lower cross-entropy value represents a closer resemblance between the distributions. This reflects that Sen. Whitehouse's stock portfolio more closely resembles the industry distribution of his own committee, the Senate Finance Committee (SSFI), compared to the Senate Banking Committee (SSBK), to which he does not belong. Also, this suggests that the cross-entropy measure effectively captures the similarity between the industry-level specialization of a committee and the preferences reflected in a congressperson's stock portfolio.

## 4.2 Comparing Cross-Entropy Between Assigned and Unassigned Committees

In this subsection, I investigate whether there is a difference in the similarity between the industry distributions of Congress members' stock transactions and the industry distributions of their assigned and unassigned committees. To conduct this analysis, I computed the cross-entropy between the NAICS code distribution of stock transactions and committees for the 115th, 116th, and 117th Congresses. I

restricted the stock transaction dates to match each congressional term (e.g., for the 115th Congress, from January 2017 to January 2019) to ensure that only transactions during these periods were considered.

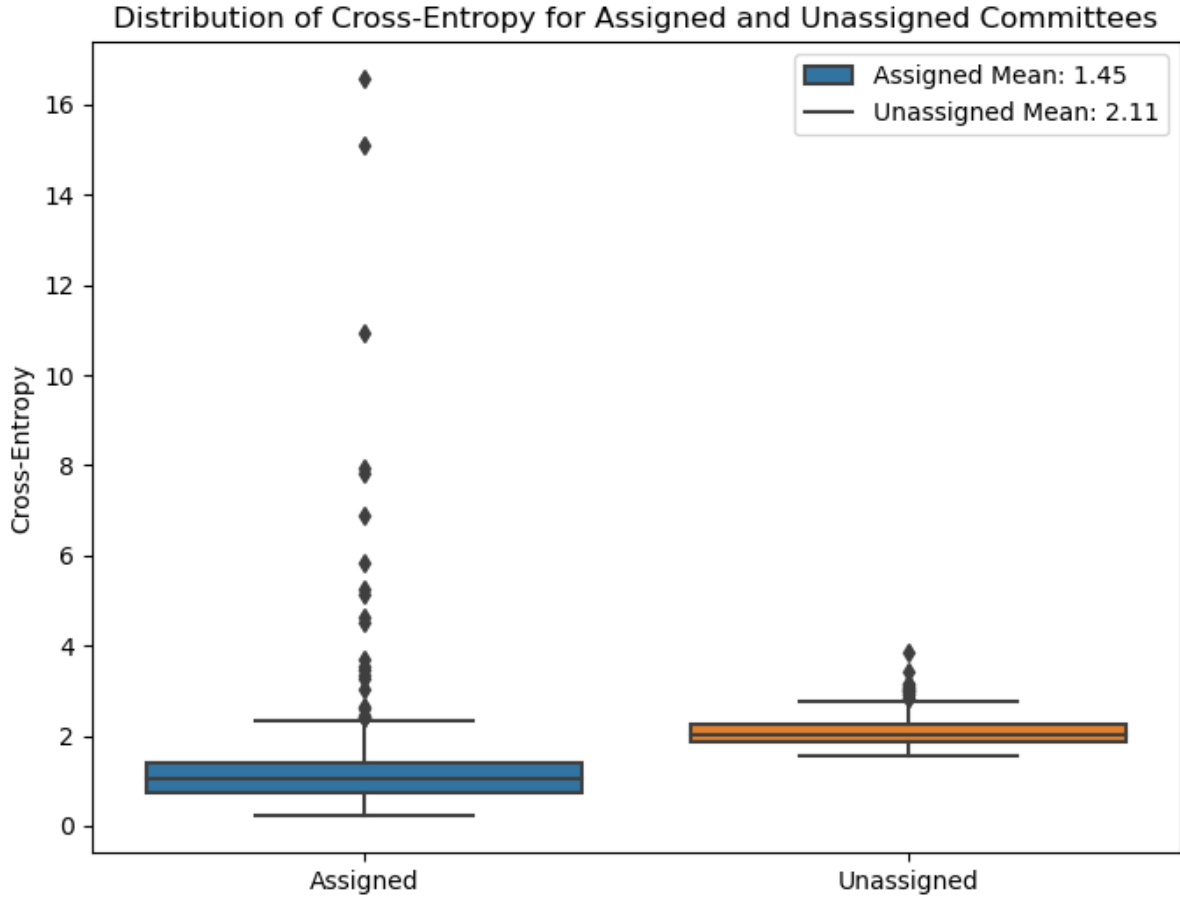


Figure 10: **Cross-entropy of assigned and unassigned committees:** The figure shows the comparison between the average cross-entropy of assigned and unassigned committees, with a sample size of 235 unique pairs of (Congressperson, Congressional year). The lower average cross-entropy for the assigned group suggests that the stock trading patterns of Congress members more closely resemble the industry distribution of their assigned committees compared to unassigned committees.

Senators and House Representatives are typically assigned to several committees during each congressional term. For each Congress member  $i$ , and for each congressional term  $t$ , I calculated the cross-entropy  $H_{i,k}^t$  between their stock transactions and each committee  $k$  (both assigned and unassigned). The objective is to ascertain whether the industry preferences in a Congress member's stock transactions align more closely with the industry specializations of their assigned committees than those of unassigned committees. To summarize these measurements, I calculated the mean cross-entropy value for each Congress member  $i$  across all their assigned committees (denoted as  $\bar{H}_{i,\text{assigned}}^t$ ), and the mean cross-entropy value across all their unassigned committees (denoted as  $\bar{H}_{i,\text{unassigned}}^t$ ) for each congressional term  $t$ . The means here are taken over all committees  $k$  that each Congress member  $i$  is assigned or not assigned to

respectively, for each term  $t$ .

The results are presented in Figure 10, with a sample size of 235 unique pairs of (Congressperson, Congressional year). The result indicates that the average cross-entropy of the assigned committees is significantly lower than that of the unassigned committees. This finding suggests that a Congressperson’s stock trading pattern aligns with the industry-level specialization of the committees they are assigned to, supporting the hypothesis that the stock trading patterns of Congress members more closely resemble the industry distribution of their assigned committees compared to unassigned committees. This result is directly opposite to the findings of Eggers and Hainmueller (2014), which conclude that there’s no evidence that members disproportionately invest in companies to which they are connected through their committee assignments. This result, on the other hand, suggests that there exists a clear resemblance between the industry-level bias in Congresspersons’ trading and their committees’ industry-level specialization.

There could be several reasons for this discrepancy. For example, the data range is different: Eggers and Hainmueller (2014) consider the period from January 1, 2004, to December 31, 2008, while in this case, we consider the 115th, 116th, and 117th Congresses, which correspond to the years 2017 to 2022. However, more fundamentally, the difference may reside in the measurement approach. Eggers and Hainmueller (2014) use a binary encoding to indicate whether a specific firm has engaged in lobbying behavior on a certain bill that is referred to a particular congressional committee. They then design a linear regression model to predict the weight of that specific firm’s stock in a congressperson’s entire investment portfolio. In this way, they investigate whether there is a relationship between a firm’s lobbying activities on bills assigned to a committee and the investment decisions of congresspersons who are members of that committee. However, this approach only captures a specific company’s lobbying behavior rather than industry-level information in its entirety. For example, multiple firms can lobby on the same bill assigned to a certain committee, then a congressperson on that committee can evaluate more broadly how this could impact the industry as a whole and selectively redesign their own portfolio. This means that there is no reason to assume that a congressperson would buy a specific stock that is being lobbied for — instead, it is more plausible to understand that such lobbying provides more detailed context about a specific industry, which the congressperson can utilize in their personal financial investment decisions.

Therefore, the cross-entropy approach, which directly measures the industry-level similarity between

a committee’s interests and a congressperson’s portfolio, provides a more intuitive understanding of the potential influences on a congressperson’s investment behavior. This approach allows us to discern a relationship between the industry-level specialization of committee assignments and the industry-level bias in Congresspersons’ stock trading, which contributes to our understanding of these influences.

However, it is also important to recognize that confounding factors could affect both committee assignments and the resemblance of stock trading to the committee’s industry-level specialization. For example, scholarly consensus suggests that committee assignments are often based on a congressperson’s existing knowledge and expertise (Boros and Fenno, 1968; Gilligan and Krehbiel, 1989; Kiewiet and McCubbins, 1991; Krehbiel, 1992; Curry, 2019). Consequently, it’s possible that the observed correlation between a congressperson’s trading patterns and their committee’s industry specialization may not entirely result from knowledge gained through congressional activities, but may also be influenced by pre-existing expertise and interests. This introduces a confounding factor, making it challenging to disentangle the effects of pre-existing expertise from the potential influence of congressional committee assignments on trading decisions. The relationship between a congressperson’s industry specialization, their committee assignments, and their stock trading patterns thus warrants further exploration.

In addition to our primary findings, it’s important to acknowledge an interesting observation from Figure 10: the variance in average cross-entropy for assigned committees is considerably higher than that for unassigned committees. This suggests that the stock trading behavior of a subset of Congresspersons deviates significantly from the industry specialization of their assigned committees. This pattern leads to another interesting finding: The conventional political science theory posits that the primary motivation for congressional members is re-election (Mayhew, 1975; Fenno, 1977). From this perspective, Congresspersons might demonstrate caution in selecting stocks to trade to avoid the perception of conflict of interest with their assigned committees’ focus industries. Essentially, they may strive to prevent any appearance of using insider knowledge gained from committee assignments for personal financial gain, which could negatively affect their chances of re-election. This finding underscores the importance of considering not just direct legislative and financial interests but also political strategy and public perception in understanding the investment behavior of Congresspersons.

## 5 Predicting Congressional Stock Transactions using Graph Neural Networks

In the previous section, therefore, I discussed the limitations of the linear prediction model used by Eggers and Hainmueller (2014), which employed a binary encoding of lobbying and committee assignments to predict the weight of a specific firm’s stock in a congressperson’s portfolio. I pointed out that this model did not fully capture the complex interactions between different entities involved in congressional activities, particularly at the industry level. However, I acknowledge that the model was an attempt to explain congressional stock transactions using potentially explanatory components such as district, PAC, lobbying, and committee assignments.

Now, it’s important to note that while the cross-entropy approach in Section 4 revealed a clear resemblance between Congresspersons’ stock trading behavior and their assigned committees’ industry-level specialization, this approach does not directly answer whether this resemblance originates from knowledge gained through congressional activities or from the Congresspersons’ expertise and experience before their congressional tenure.

In this section, I propose to use a graph neural network (GNN) (Zhou et al., 2020; Wu et al., 2020; Scarselli et al., 2008; Zhang et al., 2019) to predict congressional stock transactions using the information embedded in the congressional activities captured in the data explained in Section 3. The GNN approach is uniquely equipped to handle this task because it can model the complex relationships among various entities involved in congressional activities, all of which are naturally structured as a graph<sup>10</sup>. By using GNN, we can design a model that directly consumes the congressional graph that captures legislative-related activities of different entities, thereby enabling us to test the predictability of congressional trading behavior based on these activities. This will help us isolate the influence of congressional activities on stock trading from pre-congressional expertise and other confounding factors, which is a significant step forward in our understanding of the interplay between committee assignments, congressional activities, and stock transactions. By leveraging a graph representation of the relationships between firms, bills, committees, and congresspersons, we can train a GNN to predict whether a congressperson is likely to buy a particular stock. The advantage of using a Graph Neural Network (GNN) over a traditional

---

<sup>10</sup>Reproducible code for this section is available at [https://github.com/syyunn/gnnex/blob/main/hetero/train\\_kfold\\_auto.py](https://github.com/syyunn/gnnex/blob/main/hetero/train_kfold_auto.py)

approach such as including industry dummy variables in a regression model can be attributed to several reasons:

Firstly, the advantage of using a GNN over traditional regression models, including those using binary indicators like in Eggers and Hainmueller (2014), lies in its superior feature-learning capability. Unlike manual feature design, GNN doesn't rely on predefined indicators. It dynamically learns to aggregate neighborhood information, thus developing accurate vector representations of congresspersons and tickers. This approach enables GNN to more precisely predict buy-sell relationships between entities. In contrast, manual feature design methods, such as binary indicators, often oversimplify complex relationships and fail to capture the multifaceted dynamics present in the data. GNN, on the other hand, adapts to the intricacies inherent in congressional stock trading data, offering a more effective modeling approach.

Secondly, GNNs can better handle the heterogeneous and high-dimensional nature of our data. A traditional approach using industry dummy variables is limited in its ability to handle high-dimensional categorical variables. Furthermore, this approach treats each industry as a separate and independent entity, ignoring any potential correlations or dependencies between industries. On the other hand, GNNs can handle high-dimensional data and also account for the interconnectedness of entities.

Finally, GNNs are capable of learning and evolving over time, allowing them to adapt and improve their predictions as new data comes in. This is especially useful in our context, where the behavior and preferences of Congresspersons, the focus of committees, and the performance of industries and stocks can change over time.

## 5.1 Designing a Classifier with Graph Neural Networks

To predict congressional stock transactions using a graph neural network (GNN) approach, I design a binary classifier that takes as input a graph  $G$ , a congressperson and a ticker (stock symbol). The classifier, denoted as  $f(G, \text{congressperson}, \text{ticker})$ , will output a binary prediction of either 0 or 1, indicating whether an edge (a buy or sell relationship) exists between the given congressperson and the ticker.

The hidden representations (Rauber et al., 2016; Das et al., 2020) of the congressperson and the ticker, denoted as  $h_{\text{congressperson}}$  and  $h_{\text{ticker}}$  respectively, are obtained as outputs of the GNN model. The main task in this approach is to train the GNN model to learn a computational graph that generates “good” representation of the congressperson and the ticker,  $h_{\text{congressperson}}$  and  $h_{\text{ticker}}$ , which involves

how to effectively encode the information embedded in the network to perform the downstream task of binary classification (Féraud and Clérot, 2002).

To design the classifier, a probabilistic modeling approach is used that comprises of a sigmoid function applied to the logit, which is the output of the model. The logit of the model is obtained by passing the representation learned by the GNN,  $h_{congressperson}$  and  $h_{ticker}$ , to an MLP (Multi-layer perceptron) (Gardner and Dorling, 1998; Tang et al., 2016) that maps the representations of the congressperson and the ticker to a single logit. In other words, the MLP takes as input the representations of the congressperson and the ticker learned by the GNN, and outputs a logit that will be used to compute the probability of the existence of edge between them. MLP is simply an affine transformation over the concatenation of two representations,  $h_{congressperson}$  and  $h_{ticker}$ , followed by a non-linear activation function (Lu and Lu, 2020), which is ReLU (Agarap, 2018) in this case.

Formally, the logit of the classifier is defined as:

$$\text{logit} = \text{MLP}(x) \text{ where } \text{MLP}(x) = \text{ReLU}(Ax + b)$$

$$x = \text{concat}(h_{congressperson}, h_{ticker})$$

$$A \in \mathbb{R}^{(d+d) \times 1}$$

$$b \in \mathbb{R}^1$$

- **logit**: This is the output of the model. It's a transformed version of the probability that a congressperson would invest in a particular stock. In this binary classification problems, the logit (also known as log-odds) is the logarithm of the odds  $p/(1-p)$  where  $p$  is the probability of a positive event that congressperson trades such stock.

- **MLP**: This stands for Multi-Layer Perceptron, a type of artificial neural network. In this case, it's a function that takes the concatenated embeddings of a congressperson and a ticker as input and produces a logit as output.

- **$h_{congressperson}$ ,  $h_{ticker}$** : These are the vector embeddings of a congressperson and a ticker symbol, respectively. Each vector embedding represents the congressperson or ticker in the learned feature space. The embeddings are of dimension  $d$ .

- **$\text{ReLU}(Ax + b)$** : This is a Rectified Linear Unit activation function applied to a linear transformation

of the input. ReLU is defined as  $\text{ReLU}(x) = \max(0, x)$  and is used to introduce non-linearity into the model.

- **concat**: This is a function that concatenates (joins together) two vectors. Here, it concatenates the embeddings of a congressperson and a ticker symbol into a single vector of shape  $1 \times 2d$ , where  $d$  is the dimension of the individual embeddings. This shape is designed to be compatible with the weight matrix  $A$  which is of shape  $2d \times 1$ .

- **$A$** : This is a weight matrix for the linear transformation in the MLP. It's of shape  $(d+d) \times 1$ , meaning it takes a vector of size  $2d$  and transforms it to a vector of size 1.

- **$b$** : This is a bias term for the linear transformation in the MLP. It's added to the output of the linear transformation before the ReLU activation is applied.

- **$d$** : This represents the dimensionality of the vector embeddings of the congressperson and ticker.  $d + d$  is therefore the dimensionality of the concatenated input vector.

The sigmoid function is then applied to the logit to obtain a probability value:

$$\text{prob} = \sigma(\text{logit})$$

where  $\sigma(x)$  is the sigmoid function. The probability value indicates the likelihood of an edge existing between the given congressperson and the ticker. If the probability value is above a certain threshold, we predict that an edge exists between them, otherwise we predict that there is no edge.

Then remaining task is how to design a GNN model that can effectively learn the representations of the congressperson and the ticker,  $h_{\text{congressperson}}$  and  $h_{\text{ticker}}$ , respectively, which can be used to train the classifier. In the following section, I will discuss the design of the GNN model.

## 5.2 Design of the Graph Neural Network Architecture

To obtain the representations  $h_{\text{congressperson}}$  and  $h_{\text{ticker}}$ , I use a GNN approach that is designed to handle the complexity and dynamics of the congressional graph. The GNN approach is based on the idea of message passing and updating (Zhou et al., 2020; Wu et al., 2020), which is a process of aggregating information from the neighbors and updating the representation of each node accordingly.

In the case of the congressional graph, I use an edge-conditioned convolution GNN model (Gilmer



et al., 2017; Simonovsky and Komodakis, 2017), which takes into account the edge attributes, such as the date, to better capture the complex relationships in the graph. The message passing, aggregation and updating in this model is defined as:

$$\mathbf{h}'_i = \Theta \mathbf{h}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{MLP}(\mathbf{e}_{i,j}) \cdot \mathbf{h}_j$$

where  $\mathbf{h}_i$  and  $\mathbf{h}_j$  are the representations of nodes  $i$  and  $j$ , respectively,  $\mathbf{e}_{i,j}$  is the edge attribute between nodes  $i$  and  $j$ ,  $\mathcal{N}(i)$  is the set of neighbors of node  $i$ ,  $\Theta$  is a learnable matrix of size  $d \times d$ , where  $d$  is the dimension of the representation space, and  $\mathbf{MLP}$  takes the edge attribute  $\mathbf{e}_{i,j}$  as input and outputs a weight matrix of size  $d \times d$ . This weight matrix is then multiplied with the representation  $\mathbf{h}_j$  of the neighbor node  $j$  to obtain a message  $\mathbf{m}_{i,j} = \mathbf{MLP}(\mathbf{e}_{i,j}) \cdot \mathbf{h}_j$ . In the updating step, the message from each neighbor node is aggregated by summing them up, and the resulting sum is added to the current representation  $\mathbf{h}_i$  of node  $i$  multiplied by the learnable parameter matrix  $\Theta$  to obtain the updated representation  $\mathbf{h}'_i$ .

In the case of the congressional graph, the edge attribute  $\mathbf{e}_{i,j}$  represents the relationship between nodes  $i$  and  $j$  at a specific date, which is represented as the elapsed time from a reference date (in this case, January 1, 2016). However, in our case, we have different types of edges, which means that  $\mathbf{MLP}(\mathbf{e}_{i,j})$  should be differently defined for different types of edges. This is because parsing the information of start and end dates should be considered differently across different edge types. For example, committee assignments of a congressperson that occurred over a specific congressional year should be considered differently from the date information that a certain firm lobbied on a certain bill. To account for this, I used the expanded version of the above formula:

$$\mathbf{h}_i^{(l+1)} = \Theta^{(l)} \mathbf{h}_i^{(l)} + \sum_{j \in \mathcal{N}(i)} \mathbf{MLP}_k^{(l)}(\mathbf{e}_{i,j}^{(k)}) \cdot \mathbf{h}_j^{(l)}$$

Here,  $l$  represents a layer, and we can expand the expressivity of such message passing and updating process by stacking up the repeated layers of this operation. This allows the model to learn a more complex representation of each node, which is essential for capturing the intricate relationships in the congressional graph. Experimentally, I found that using 2 layers of message passing and updating was sufficient to learn the best representation of each node and used this configuration for the GNN model.

In conclusion, our GNN aims to learn the optimal parameter set that defines  $\Theta^{(l)}$  and  $\mathbf{MLP}_k^{(l)}$  to

output the best representations  $\mathbf{h}_i^{(l)}$  and  $\mathbf{h}_j^{(l)}$ , which helps to perform the downstream task successfully. In this case, the downstream task is generating the best logit in the prediction head,  $\text{MLP}(\mathbf{h}_{\text{congressperson}}, \mathbf{h}_{\text{ticker}})$ . It is also important to note that the representations  $\mathbf{h}_i^{(l)}$  and  $\mathbf{h}_j^{(l)}$  are initialized randomly before they are provided into the first layer of the message passing and updating process. Through multiple rounds of message passing and updating, the GNN is tuned to output the best representation of each node that scores the best performance as possible in binary classification of edge existence.

## 5.3 Training & Evaluation of the GNN

### 5.3.1 Dataset Preparation

In the context of our GNN architecture, the goal is to predict the existence of edges between two nodes, a task commonly known as link prediction. To train the GNN for this task, the dataset must be prepared for training and evaluation (test). The dataset consists of a total of 24,675 edges, which represent the relationship (congressperson, buy-sell, ticker).

To create a balanced dataset for the link prediction task, the dataset is divided into a train and test set with an 8:2 ratio, resulting in 19,740 instances for training and 4,935 instances for testing. The network is then trained using the 19,740 instances and its performance is evaluated on the 4,935 test instances.

In addition, to ensure a balanced dataset, the same number of randomly sampled negative edges (Yang et al., 2020) is prepared. These negative edges are created by randomly selecting pairs of nodes (congressperson and ticker) that do not have a connection in the original dataset. This results in a total of 39,480 edges for training and 9,870 edges for testing. Including both positive and negative examples in the training process helps the model to better differentiate between true and false existence of edges between congressperson and ticker nodes, improving its ability to predict links in the graph.

### 5.3.2 Training of the GNN

For the training of the GNN, a two-layer GNN architecture, as  $l = 2$ , is employed. Additionally, node embeddings  $h_i$  are represented as vectors in a 64-dimensional space ( $h \in \mathbf{R}^{64}$ ). This hyperparameter is also set experimentally.

In order to measure the performance of the model during the training process, binary cross-entropy loss is used as the loss function. Binary cross-entropy loss is particularly suitable for binary classification

problems (Ruby and Yendapalli, 2020), such as link prediction (Zhang and Chen, 2018), where the goal is to differentiate between the presence and absence of a connection between two nodes. This loss function quantifies the difference between the predicted probabilities and the true labels, and penalizes the model for incorrect predictions. Formally, the binary cross-entropy loss for a set of samples is defined as:

$$L = - \sum_{i=1}^N (y_i \cdot \log(p_i) + (1 - y_i) \cdot \log(1 - p_i))$$

where  $L$  represents the total binary cross-entropy loss.  $N$  is the total number of samples.  $y_i$  is the true label for the  $i$ th sample (1 for the presence of a connection, and 0 for the absence of a connection).  $p_i$  is the predicted probability of a connection existing between two nodes for the  $i$ th sample.

By minimizing the binary cross-entropy loss, the GNN learns to accurately predict the existence or non-existence of links in the network, ultimately improving its performance on the link prediction task. For the minimization, the Adam optimizer (Kingma and Ba, 2014) with stochastic gradient descent (SGD) is utilized. SGD is an iterative optimization algorithm that updates the model’s parameters based on a random sample (or minibatch) of training data in each iteration (Amari, 1993). This approach helps in converging faster and reduces the impact of noisy gradients, thus improving the optimization process. Adam is an adaptive learning rate optimization algorithm, combining the advantages of two other popular optimization methods, AdaGrad and RMSProp (Kingma and Ba, 2014). This optimizer is well-suited for large-scale problems and is known for its ability to efficiently handle noisy and sparse gradients, making it a suitable choice for training GNNs.

To obtain a more robust estimation of the model’s performance and uncertainty, a 5-fold cross-validation (Hastie et al., 2001) is performed. In this approach, the entire dataset is randomly split into five equal-sized chunks. For each fold, one chunk is used as the test set, while the remaining chunks are combined to form the training set. This process is repeated five times, with each chunk being used once as the test set. This technique allows for a better understanding of the model’s performance across different subsets of the dataset and provides uncertainty statistics of overall prediction performance.

### 5.3.3 Evaluation & Ablation Study

In this study, we conducted a link-prediction (Zhang and Chen, 2018) task to predict the existence of an edge between a congressperson and a ticker, symbolizing the trade relationship - whether the given

congressperson would sell or buy a particular stock. This task was performed using a variety of edge types, and the performance was evaluated using two metrics: accuracy and Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

The results of this evaluation are depicted in Figures 11 and 12. With all edge types included, the model achieved an accuracy of approximately 81% and an AUC-ROC of 0.89. These results indicate that the model was generally effective at predicting the stock transactions of congresspersons.

In addition, to further understand the importance of each edge type, I conducted an ablation study, where I systematically removed each edge type from the training and testing data and observed the resulting performance drop. The most significant drop in performance was observed when the edge type ('congressperson', 'assignment', 'committee') was removed. This resulted in a decrease in accuracy from 81% to 67%, and a decrease in AUC-ROC from 0.89 to 0.76. This suggests that the ('congressperson', 'assignment', 'committee') edge type carries significant information for predicting a congressperson's stock transactions.

In comparison, the removal of other edge types, such as ('bill', 'assigned\_to', 'committee'), or ('ticker', 'lobbies\_on', 'bill'), resulted in less dramatic performance drops. This further underscores the relative importance of the ('congressperson', 'assignment', 'committee') edge type in this prediction task.

To further quantify the importance of each edge type, I employed the concept of Shapley values (Winter, 2002; Hart, 1989; Littlechild and Owen, 1973), a concept borrowed from cooperative game theory. In this context, each edge type can be considered as a player in a cooperative game, where the "payout" is the performance of the model.

$$\varphi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S))$$

Here,  $\varphi_i(v)$  is the Shapley value for edge type  $i$ , representing the average marginal contribution of edge type  $i$  to the performance of the model, considering all possible combinations of edge types.  $N$  is the set of all edge types, not the total number of edges. In our case, there are four edge types, so  $N$  is 4.  $S$  is a subset of  $N$  that does not include edge type  $i$ ,  $|S|$  is the number of edge types in subset  $S$ , and  $n$  is the total number of edge types.  $v(S \cup i)$  and  $v(S)$  represent the performance of the model when edge type  $i$  is added to and excluded from the subset  $S$  of edge types, respectively. This means that the Shapley value indicates how much each edge type contributes to the performance of the model, which in our case

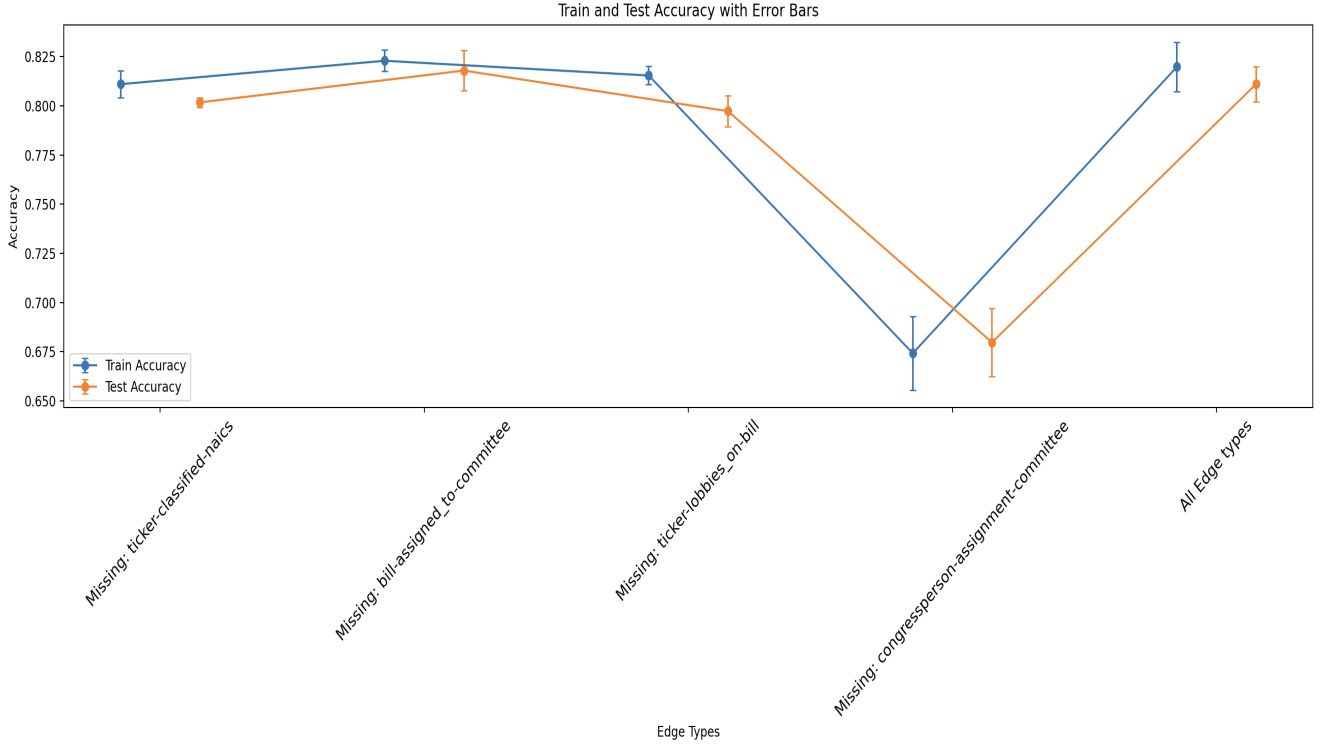


Figure 11: **Accuracy drop for different edge types.** The figure shows the accuracy of the model with all edge types included and with each edge type removed one at a time. With all edge types included, the model achieved an accuracy of approximately 81%. The most significant drop in accuracy, to 67%, was observed when the edge type ('congressperson', 'assignment', 'committee') was removed. This suggests that the ('congressperson', 'assignment', 'committee') edge type carries significant information for predicting a congressperson's stock transactions.

is measured by prediction accuracy or AUC-ROC.

The Shapley values were computed over all  $16(= 2^4)$  possible combinations of the four different edge types, with each combination evaluated through a five-fold cross-validation process. The uncertainty associated with each Shapley value, as reflected in the standard deviation across the five folds, provides an indication of the stability of the Shapley value estimates. The results of this analysis are shown in Figure 13.

These findings indeed underline the crucial role of the edge representing the committee assignment of the congressperson in predicting congresspersons' stock transactions. Nevertheless, it's worth noting that these results are contextually bound to our graph design. The importance of committee assignments in predicting stock transactions might be somewhat overemphasized due to the absence of other possible edges connecting congresspersons to stocks, such as bill co-sponsorship, business relations, or previous occupation. Including additional edges reflecting other aspects of congressional activities in future work might provide a more accurate view of the importance of committee assignments and other factors in

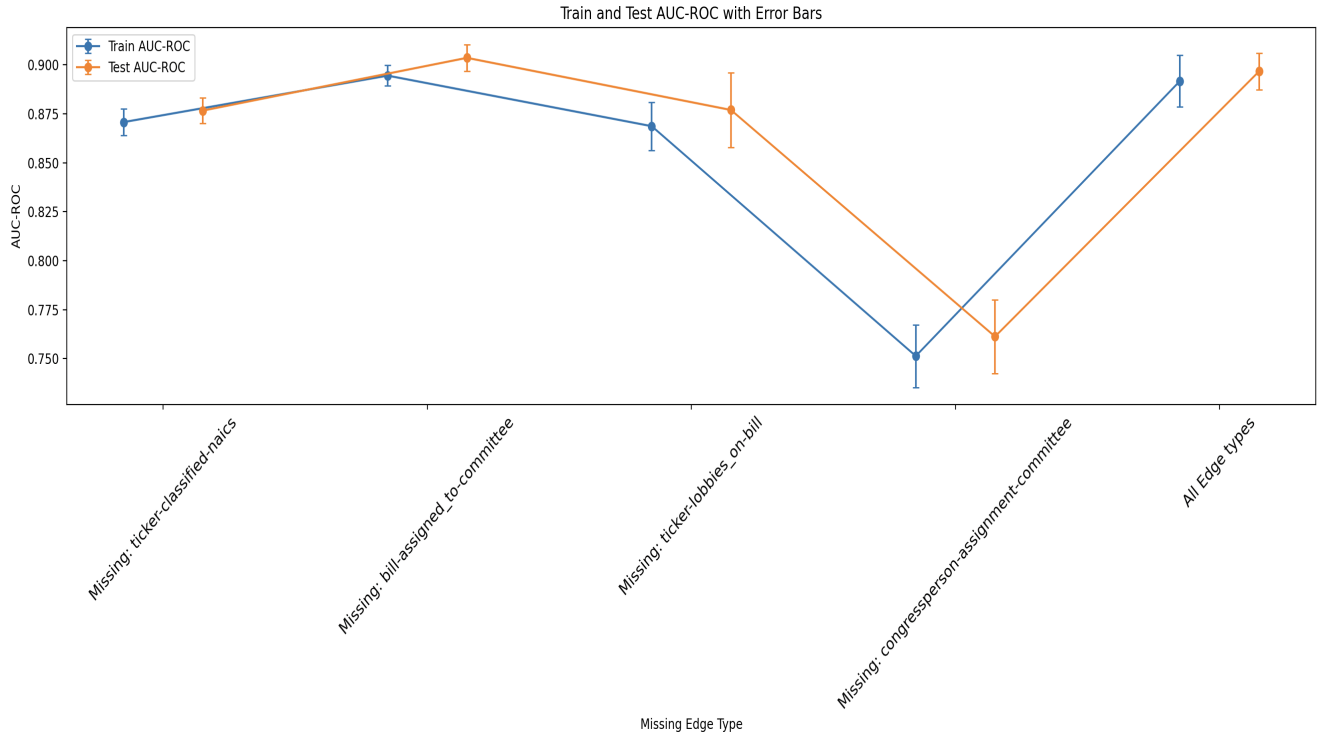


Figure 12: **AUC-ROC drop for different edge types.** The figure shows the AUC-ROC of the model with all edge types included and with each edge type removed one at a time. With all edge types included, the model achieved an AUC-ROC of approximately 0.89. The most significant drop in AUC-ROC, to 0.76, was observed when the edge type ('congressperson', 'assignment', 'committee') was removed. This suggests that the ('congressperson', 'assignment', 'committee') edge type carries significant information for predicting a congressperson's stock transactions.

predicting congresspersons' stock trading behavior.

In the Shapley value analysis, I observed that the edge type ('bill', 'assigned\_to', 'committee') had a Shapley value of zero or even negative. This suggests that this type of edge does not contribute to increasing the performance of the model. In fact, it appears to harm the performance when included. The reason for the zero or negative contribution of the edge type ('bill', 'assigned\_to', 'committee') is not immediately clear and warrants further investigation. One possible explanation could be that bills can be assigned to different committees, making this information more complex and potentially harder for the model to utilize effectively. In contrast, the firm-level lobbying information and industry-level classification of firms provided by the edge types ('ticker', 'classified\_as', 'naics') and ('ticker', 'lobbies\_on', 'bill') are more straightforward. These edge types may allow the model to more easily discern patterns in company behavior and use this information to make accurate predictions.

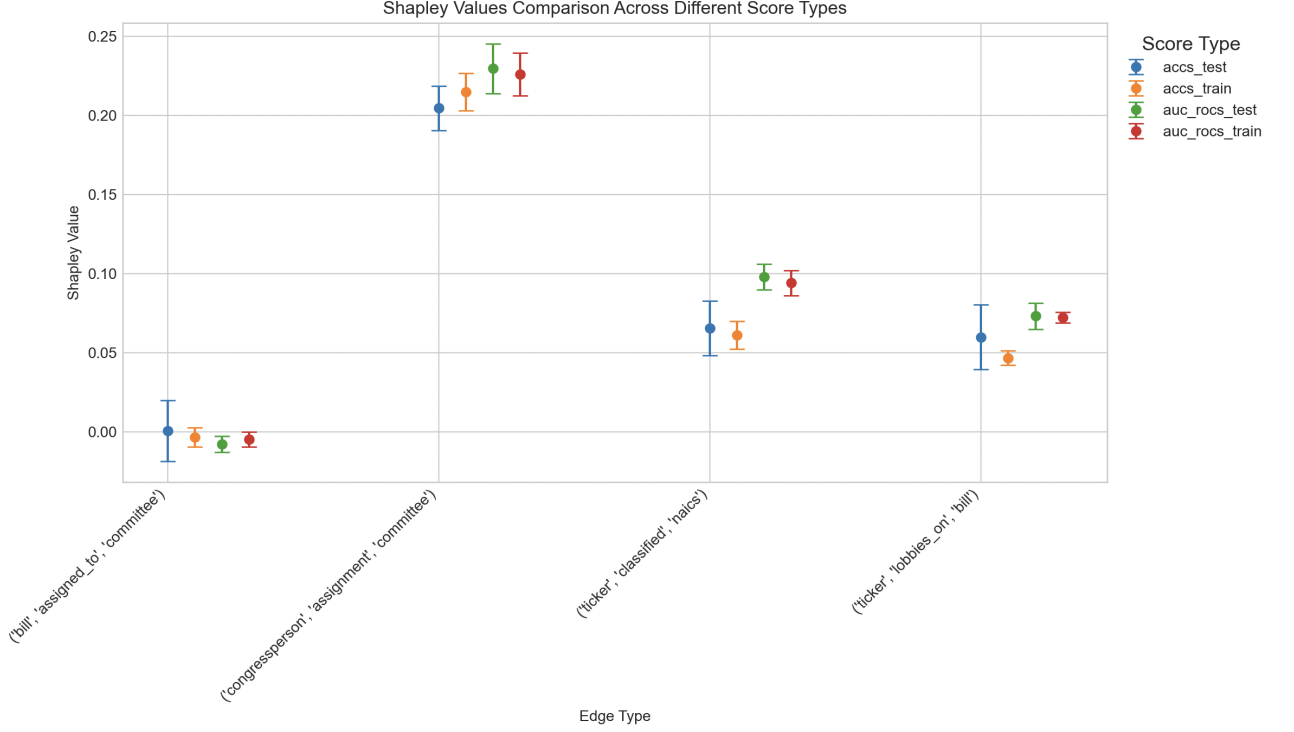


Figure 13: **Shapley values for different edge types.** The figure shows the Shapley values for each edge type, computed over all  $16(= 2^4)$  possible combinations of the four different edge types. The Shapley value for an edge type represents its average marginal contribution to the performance of the model, considering all possible combinations of edge types. The most important feature, according to the Shapley value analysis, was ('congressperson', 'assignment', 'committee'), followed by ('ticker', 'classified\_as', 'naics') and ('ticker', 'lobbies\_on', 'bill'). This further reinforces the conclusion that the ('congressperson', 'assignment', 'committee') edge type plays a crucial role in predicting congressperson's stock transactions.

## 5.4 Interpreting Predictions with GNNExplainer<sup>11</sup>

To further explain which nodes and edges the trained model focuses on to output such predictions, I used GNNExplainer (Ying et al., 2019), which trains soft node and edge masks that can be applied to the original graph to extract the subgraph most relevant to the prediction.

The detailed implementation of GNNExplainer involves modifying the update rule for the node representations in the GNN. The original update rule is:

$$\mathbf{h}_i^{(l+1)} = \Theta^{(l)} \mathbf{h}_i^{(l)} + \sum_{j \in \mathcal{N}(i)} \text{MLP}_k^{(l)} \left( \mathbf{e}_{i,j}^{(k)} \right) \cdot \mathbf{h}_j^{(l)}$$

In this modified update rule, we introduce soft node and edge masks, denoted by  $m_i$  and  $m_{i,j}$  respectively, which are element-wise multiplied with the node and edge representations:

<sup>11</sup>Reproducible code for this subsection is available at [https://github.com/syyunn/gnnex/blob/main/hetero/explain\\_edge.py](https://github.com/syyunn/gnnex/blob/main/hetero/explain_edge.py)

$$\mathbf{h}_i^{(l+1)} = m_i \cdot \Theta^{(l)} \mathbf{h}_i^{(l)} + \sum_{j \in \mathcal{N}(i)} m_{i,j} \cdot \text{MLP}_k^{(l)} \left( \mathbf{e}_{i,j}^{(k)} \right) \cdot \mathbf{h}_j^{(l)}$$

The soft masks are continuous values between 0 and 1, as opposed to hard masks which are either 0 or 1. This allows us to optimize the masks using stochastic gradient descent (SGD).

The objective of the optimization is to minimize the L2 loss between the predictions of the original graph and the masked graph, denoted by  $y_{\text{original}}$  and  $y_{\text{masked}}$  respectively:

$$\mathcal{L} = \|y_{\text{original}} - y_{\text{masked}}\|^2 + \lambda \cdot (\|m_i\|_1 + \|m_{i,j}\|_1)$$

Here,  $\lambda$  is a regularization parameter that controls the complexity of the subgraph by encouraging sparsity in the masks. For this study, I used a value of 0.01 for  $\lambda$ . The masks are trained separately for each prediction, which makes the method less scalable but provides insights into which nodes and edges are important for mimicking the original model’s prediction. After training the node and edge masks, I can generate a subgraph by applying the masks to the original graph. The complexity of the subgraph can be controlled by setting a cutoff level for the mask values, or by adjusting the regularization parameter  $\lambda$ . Figures 14 and 15 provide examples of the output from GNNExplainer for specific stock transactions.

Figure 14 focuses on Senator Ron Wyden’s transaction of Applied Materials Inc. (AMAT)’s stock. To generate the subgraph from the entire graph, I applied the node and edge masks trained by the GNNExplainer. This process involved selecting the nodes and edges with the highest scores from the masks. For instance, among all 56 bills that AMAT lobbied on, I selected the two bills that had the highest scores in the edge mask. Similarly, among the two NAICS code classifications of AMAT, I selected the NAICS code 333242 (Semiconductor Machinery Manufacturing), which had the highest score in the edge mask.

The GNNExplainer successfully identified the most relevant bill for this transaction, S.3933-116, the CHIPS Act, which subsidizes the US semiconductor industry. Interestingly, the GNNExplainer also highlighted H.R.7617-116, a more general appropriations act, titled “Defense, Commerce, Justice, Science, Energy and Water Development, Financial Services and General Government, Labor, Health and Human Services, Education, Transportation, Housing, and Urban Development Appropriations Act, 2021 ”. While this bill may not be directly related to subsidization of the semiconductor industry,



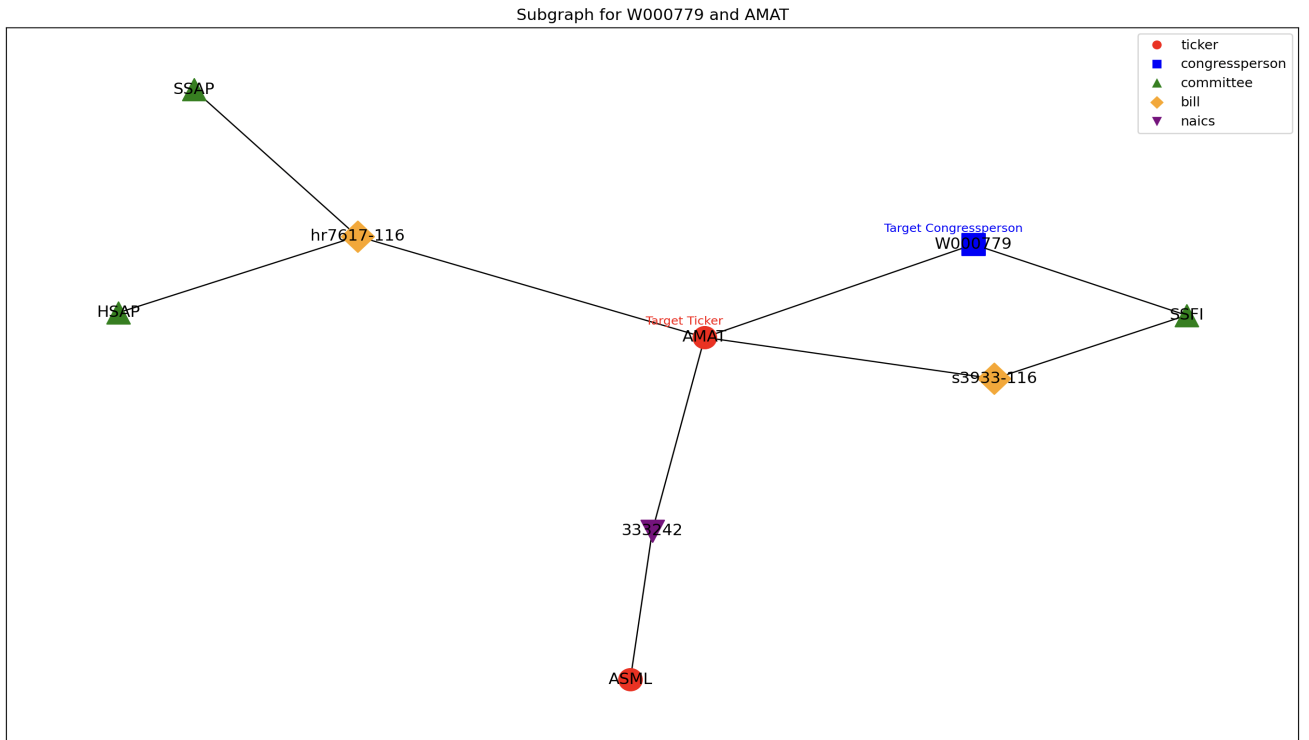


Figure 14: **GNNExplainer output for Senator Ron Wyden’s transaction of AMAT’s stock.** The figure shows the subgraph extracted from the entire graph using the node and edge masks trained by the GNNExplainer. The GNNExplainer identified S3933-116 (the CHIPS Act) and HR7617-116th among the bills that AMAT lobbied on, NAICS code 333242 (Semiconductor Machinery Manufacturing) among the NAICS code classifications of AMAT, and ASML, among the firms classified as 333242 as the most influential factors for this transaction.

it is indicative of the broader legislative environment. It’s worth noting that the National Defense Authorization Act (NDAA) for Fiscal Year 2021, which is often associated with appropriations for the semiconductor industry, was also part of the data. However, the GNNExplainer did not identify it as a highest score node for this particular transaction. This could be due to a variety of reasons, such as the complexity of the appropriations process or the indirect relationship between the NDAA and H.R.7617-116.

In addition to identifying relevant bills, the GNNExplainer also provided insights into the industry context of Senator Wyden’s transaction of AMAT’s stock. The NAICS code 333242, which corresponds to Semiconductor Machinery Manufacturing, includes four different companies: Applied Materials, ASML LLC (ASML), Azenta Inc (AZTA), and Tokyo Electron America Inc (TOELY). The GNNExplainer ranked ASML as the most relevant node for this transaction. This makes sense given the industry dynamics. While Azenta is a semiconductor company, it primarily focuses on bio-related semiconductor products, which may not be as directly relevant to AMAT’s business. Tokyo Electron, on the other hand,

is a much smaller firm compared to ASML or AMAT. Most importantly, ASML and AMAT are known to have a competitive relationship, with AMAT consistently striving to capture market share from ASML. Therefore, the GNNExplainer’s identification of ASML as the most relevant node for this transaction is consistent with the industry context.

In addition, as the chair of the Senate Finance Committee (SSFI), Senator Wyden has been a key player in initiatives related to the semiconductor industry. Despite the unexpected selection of H.R.7617-116 by the GNNExplainer, the overall results still highlight the relevance of Senator Wyden’s legislative activities and industry context to his stock transactions.

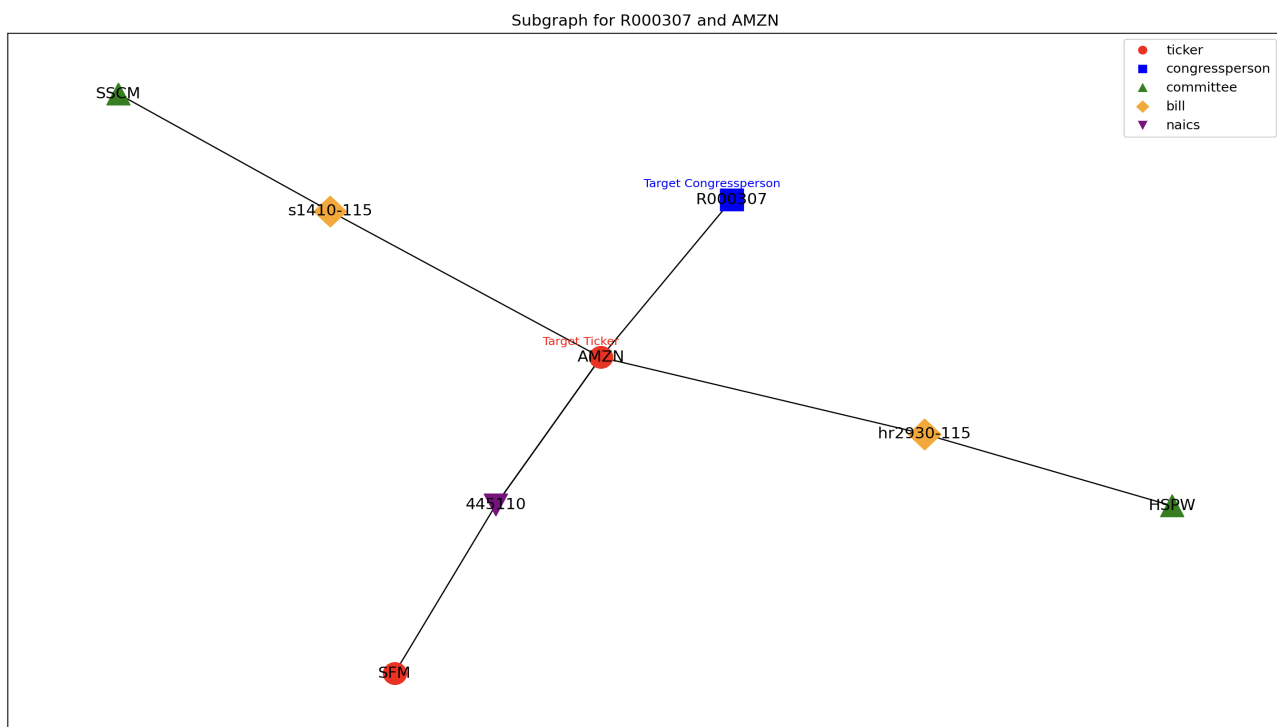


Figure 15: **GNNExplainer output for Senator Pat Roberts’s transaction of Amazon’s stock.** The figure shows the subgraph extracted from the entire graph using the node and edge masks trained by the GNNExplainer. The nodes and edges in the subgraph were selected based on their high scores in the masks, indicating their relevance to the model’s prediction for this specific transaction. The GNNExplainer identified H.R.2930-115 (Drone Innovation Act of 2017) and S.1410 (Safe DRONE Act of 2017) among the bills that Amazon lobbied on as the most influential factors for this transaction. This is particularly interesting given Senator Roberts’s known legislative activities related to drone technology.

Figure 15 presents another example, focusing on Senator Pat Roberts’s transaction of Amazon (AMZN)’s stock. Among the 119 bills that Amazon lobbied on, the GNNExplainer identified H.R.2930-115 (Drone Innovation Act of 2017) and S.1410 (Safe DRONE Act of 2017) as the most relevant bills. This is particularly interesting because Senator Roberts co-sponsored the bill S.2730-116, which establishes a

Drone Advisory Committee. Furthermore, 2017 was the year when Amazon started to publicize their plans for drone delivery<sup>12</sup>. Therefore, we can interpret that Senator Roberts’s transaction of Amazon’s stock was likely influenced by his legislative activities related to drone technology.

Indeed, Figures 16 and 17 reinforce the main story that committee membership matters. These examples showcase how the GNNExplainer can elucidate the decision-making process of the graph neural network for individual predictions. These figures demonstrate the ability of GNNExplainer to generate a plausible explanation for each specific prediction case, thereby providing a mechanism to interpret the predictions made by the neural network. The “black-box” nature of neural networks is a common criticism (Shrikumar et al., 2016; Joshi et al., 2021; Castelvechi, 2016), owing to their complex, high-dimensional, and non-linear predictive process, which is often challenging to interpret and understand. This makes it difficult to trust their predictions, especially in sensitive contexts where understanding the reasoning behind predictions is essential. GNNExplainer addresses this criticism by providing a tool to interpret the predictions of the GNN, thereby making it more transparent and trustworthy.

These examples demonstrate the ability of the GNNExplainer to highlight relevant legislative and industry context for specific stock transactions. It’s noteworthy that the GNNExplainer was able to identify these relationships even without explicit information about bill titles, text, or sponsorship relationships. This suggests that the GNNExplainer is effectively capturing the underlying patterns in the data that are relevant for the prediction task.

## 6 Identifying Informative Predictors in Congressional Stock Trading Using an LLM Agent

This section delves into the design of a new approach to solve the same link prediction task from Section 5 using an LLM Agent. I employ an emerging tool - a Large Language Model (LLM) agent (Park et al., 2022; Shinn et al., 2023; Wang et al., 2023) - notable for its ability to reason and act autonomously. By tasking this LLM agent with performing link prediction tasks across various legislator-ticker pairs, I anticipate that the LLM Agent will formulate its own generalizable theories applicable to various instances of predictive tasks, thereby deducing underlying decision rules that can be readily interpreted by humans through direct examination of the generated text.

---

<sup>12</sup><https://www.businessinsider.com/amazon-takes-critical-step-toward-drone-delivery-2017-5>

## 6.1 High-Dimensional Complexity and Theorization in Heterograph Data

In this section, I justify the use of a Large Language Model (LLM) agent in our study. The primary motivation for employing an LLM agent is to identify important features that can explain congressional stock trading behaviors in an interpretable manner. This task becomes particularly challenging with high-dimensional data, a common characteristic of graph-structured data. Unlike traditional tabular data, graph-structured data often involves large adjacency matrices that are highly dimensional, scaling as  $N^2$  with  $N$  being the number of nodes (Ward et al., 2011). This high dimensionality poses distinct challenges in data analysis and theorization, particularly when analyzing networks that capture societal phenomena (Tang and Liu, 2012; Thi and Nguyen-Hoang, 2013).

For instance, Eggers and Hainmueller (2014) adopt a manual binary indicator as a covariate rather than directly utilizing the “connections” of lobbying activities of firms to committees and congresspersons. This oversimplification is somewhat an unwilling preference among scholars when dealing with network data, largely because of the limitations in human capacity to design effective measurements. This challenge often leads to a reliance on machine learning, as demonstrated in Section 5, to learn optimal feature representations that effectively utilize such connections.

In sum, the field of political science currently lacks a widely agreed-upon approach to manage the high dimensionality inherent in graph-structured data (Lazer, 2011; Ward et al., 2011). The challenge of managing high dimensionality in graph-structured data is further intensified when using heterographs as predictors. Heterographs, characterized by their diverse node and edge types, offer a comprehensive data representation format for social phenomena. Despite this, they are infrequently employed in social network analysis. This underutilization is notable given their potential as a general framework for capturing the complexities of social interactions.

To address the challenges associated with the high dimensionality of graph-structured data, particularly when using heterographs as predictors in political science, this study introduces a novel method that leverages a Large Language Model (LLM) agent. The primary motivation behind employing an LLM agent is to effectively identify key features within the network—such as specific types of connections and combinations thereof—that can explain congressional stock trading behaviors in an interpretable manner.

The LLM agent’s role in this context is to sift through the complex network data, pinpointing potential

predictors that might otherwise be overlooked due to the data’s high dimensionality. For example, it could be hypothesized that if a legislator assigned to a committee handles a significant number of bills targeted at a particular industry, this could be a strong predictor of whether that legislator engages in transactions of certain stock tickers related to the same industry. The following sections will delve into a contextual explanation of what Large Language Models (LLMs) and LLM agents are, and the specific design of the theorizing agent being proposed.

## 6.2 Large Language Models (LLMs) and LLM Agents

Large Language Models (LLMs) are advanced AI systems capable of understanding, generating, and engaging with human language (Zhao et al., 2023). These models are trained on vast amounts of text data, allowing them to grasp a wide array of language patterns, nuances, and contexts. LLMs like OpenAI’s GPT-3.5 and GPT-4 exemplify this technology (Radford and Narasimhan, 2018), demonstrating near-human-like reasoning abilities. These models can make decisions based on a complex understanding of language, context, and, to some extent, the logic and knowledge embedded in their training data (Lampinen et al., 2022).

One key characteristic of LLMs is their ability to generate coherent and contextually relevant text, making them particularly useful for a range of applications, from content creation to data analysis. In the realm of predictive modeling, LLMs stand out for their ability to process and interpret complex data structures in a way that mimics human cognitive processes (Kojima et al., 2022; Hayashi et al., 2019). This capability makes them highly valuable in fields like political science (Luitse and Denkena, 2021), where interpreting intricate relationships within data is crucial.

A notable trend in their application is the shift from using them as mere one-time question-answering systems to leveraging them in a more dynamic and interconnected manner. Expanding on this concept, the proposed approach seeks to enhance the problem-solving capabilities of LLMs by enabling them to dissect complex problems into a sequence of interconnected steps. This methodology facilitates a more detailed and systematic exploration of solutions, where each step builds on the insights gained from the previous one. By breaking down problems in this manner, LLMs can provide a more accurate solution of the problem (Wei et al., 2022; Yao et al., 2023a).

In addition to the aforementioned step-by-step approach, there is an emerging trend of augmenting Large Language Models (LLMs) with a variety of tools, thereby enhancing their problem-solving capa-

bilities. These augmented LLMs, or so called “LLM Agents”, are empowered to utilize any given tools based on their own discretion and reasoning, choosing the most appropriate tool for a specific situation (Yao et al., 2023b). For instance, an LLM augmented with web search tool can effectively invoke this tool as a part of its problem-solving process whenever it deems necessary. This self-awareness enables them to identify situations where their existing knowledge base may be insufficient. In such scenarios, an LLM Agent can decide to refer to external sources of data, such as the web or apis, to acquire more current and accurate information (Qin et al., 2023; Patil et al., 2023).

Then, how can we leverage these LLM Agents to theorize and explain complex phenomena such as congressional stock trading, especially by navigating through graph-structured data? This question heralds a new realm of possibilities in the application of LLM Agents. In the following section, I propose a novel architecture that enables LLM Agents to tackle the link prediction task addressed in Section 5 using GNN. This architecture is designed to not only enable LLM Agents to analyze and interpret intricate data structures like graphs but also to autonomously generate explanations and hypotheses that address our key question: What drives congressional stock trading behaviors?

### 6.3 Designing Self-Learning LLM Agents

The academic community and industry have shown considerable interest in exploring new applications of Large Language Models (LLMs), encountering unique challenges along the way. One significant issue is that LLMs, due to their immense parameter size and often proprietary nature, are difficult to fine-tune.

Consequently, the focus has shifted towards designing learning experiences for LLMs that do not require traditional weight updates or parameter adjustments. This approach, known as “no-derivative learning”, has rapidly evolved into a significant field of study (Yang et al., 2023; Zhu et al., 2023; Wang et al., 2023). It involves crafting system architectures that enable LLMs or LLM Agents to learn and achieve specific goals without altering their underlying LLM model weights.

A prime example of this method is the “LLM as optimizer” (OPRO) concept (Yang et al., 2023), which proposes no-derivative learning by creating a meta-prompt that is iteratively updated. This method demonstrates that prompts can evolve through self-evaluation or self-reflection, effectively allowing the LLM to function as an optimizer. This approach has been successfully applied to solve linear regression problems, showcasing the LLM’s ability to operate in an optimizer-like capacity.

Furthermore, from the perspective of optimizing actions, Shinn et al. (2023) introduced a verbally

reflecting agent that generates task feedbacks. These feedbacks are then maintained in an episodic memory buffer, aiding the agent in making better decisions in subsequent trials. This concept of self-evaluation or self-reflection, through designing a meta-prompt that allows the agent to assess its own performance and store these reflections for future reference, has become another cornerstone of no-derivative learning.

Additionally, the Wang et al. (2023) presented a novel approach where embeddings of these reflections are stored in a vector database. This approach enables the use of similarity searches over vector space, enhancing the agent’s ability to effectively recall the most relevant past experiences. For example, Wang et al. (2023) trained LLM Agents to play Minecraft by storing information about the environment as embeddings, with corresponding skills as key-value pairs. These environment-skill pairs are then retrieved based on the current environment, allowing the LLM Agent to augment its reasoning process by applying skill sets that were effective in similar past scenarios.

## 6.4 Predictive Theorization with Self-Learning LLM Agents

As discussed, self-learning Large Language Model (LLM) Agents have emerged as a promising solution for accumulating memory from past experiences and for generating theories or hypotheses about underlying patterns in data. They achieve this by iteratively solving multiple instances of problems within the same class. By engaging these agents in iterative predictive tasks focused on the existence of links between legislator-ticker pairs, they can self-evaluate their performance. This process involves the agent assessing which observations within the dense graph data and which specific aspects of its topology or structure were pivotal in their problem-solving approach.

In this context, I propose an innovative methodology where the traditional task of link prediction is approached through a self-learning LLM Agent. This method entails having the Agent predict the presence or absence of a link for a given legislator-ticker pair. Following each prediction, the agent engages in self-reflection or evaluation, analyzing the reasons behind its success or failure in comparison to the ground truth. This approach introduces a novel dimension of self-assessment and learning. It allows the LLM Agent to progressively refine its understanding and approach to the data, potentially uncovering significant insights about the relationship between congressional activities and stock trading behaviors.

In the subsequent subsection, I will delve into the detailed design of this agent, elaborating on the

specific tasks it was assigned and the methodology employed in its development. This will include a thorough explanation of how the agent operates, the nature of its self-learning and self-evaluating mechanisms, and the rationale behind how the agent organizes its memory in a manner that aids in formulating appropriate theories or hypotheses, which in turn facilitates its achievement of high accuracy in the link prediction task.

#### 6.4.1 Dual-Tool for the LLM Agent: GraphDB and MemoryDB

In order to undertake the link prediction task within the heterograph data from Section 3, I developed an LLM Agent endowed with two distinctive tools: Graph Database (GraphDB) Access and Memory Database (MemoryDB) Access as follows:

**GraphDB Access:** This tool allows the agent to directly interact with the heterograph data from Section 3 stored in a GraphDB. The agent has the freedom to connect to this database and execute queries based on its own logic and requirements. For instance, it can formulate a query like “MATCH (l:Legislator)-[:Buy\_Sell] → (t:Ticker)-[:BELONGS\_TO] → (n:NAICS) RETURN t.ticker”, which could retrieve a list of all stock tickers that a legislator has transacted in, provided these tickers belong to the same NAICS code industry classification as the Ticker of interest. This tool empowers the agent with the ability to dynamically explore and analyze the relational data stored in the graph database.

**MemoryDB Access:** The second key component is the MemoryDB Access, which allows the agent to connect with a Vector Database. This database serves as a repository for the agent’s historical analysis and decision-making processes. Following each iteration of the link prediction task, the agent is programmed to self-generate feedback on its performance. This introspective process is informed by the agent’s own logs and reasoning paths used during prediction. From this self-reflection, the agent develops key-value pairs that encapsulate specific conditions and corresponding actions. The intention behind this mechanism is to create a set of principles that are not only applicable to the current scenario but also generalizable to future tasks. These principles are then converted into text-based dictionary formats, each paired with an OpenAI embedding that represents the textual description of the conditions. These embeddings, along with their associated condition-action pairs, are stored in the Vector Database.

This approach enables the agent to query the Vector Database for relevant decision-making criteria when faced with similar conditions in the future where the similarity is measured by the cosine similarity. Importantly, this system is designed to abstract away from specific legislator-ticker details, instead fo-



cusing on broader principles and strategies. By doing so, the agent is not just recalling past decisions but is actively learning and adapting its approach for improved predictive performance in future link prediction tasks.

#### 6.4.2 LLM Agent’s Prediction and Reflection Pipeline

This section elucidates the operational mechanics of the algorithm outlined in Algorithm 1, which governs the LLM Agent’s functionality in the context of the link prediction task. The pipeline leverages GraphDB and MemoryDB tools, enabling the agent to iteratively perform predictions, evaluations, and self-reflections.

---

**Algorithm 1:** LLM Agent Prediction and Reflection Pipeline

---

```

1 LLM  $\leftarrow$  initialize LLM with model specification;
2 graphDB, memoryDB  $\leftarrow$  initialize database connectors;
3 agent  $\leftarrow$  initialize agent with (LLM, graphDB, memoryDB);
4 foreach (legislator, ticker, label) in TrainDataset do
5     (pred, log)  $\leftarrow$  make_pred(legislator, ticker, agent);
6     eval  $\leftarrow$  evaluate(pred, label);
7     reflection  $\leftarrow$  generate_reflection(log, eval, agent);
8     insert_memory(reflection, memoryDB);
9 end
```

---

**1. LLM Initialization with GPT-4-1106-preview (GPT4-Turbo):** The agent utilizes the LLM model specified as ‘gpt-4-1106-preview’, commercially known as GPT4-Turbo. This model is chosen for its superior token-length limit, 128K, allowing the agent to process the largest possible context length among currently available GPT models by OpenAI. This extensive context handling capability is crucial for analyzing complex, high-dimensional data.

**2. Prediction Process:** The ‘make\_pred’ function is executed by the agent using a prediction template. This template explicitly defines the task of link prediction, instructing the agent to determine the existence or non-existence of a link. The agent is guided to utilize specific data and tools. The template is available at Appendix C.

**3. Evaluation Function:** The ‘evaluate’ function returns a binary outcome – 1 for a successful

prediction (correct link existence) and 0 for an unsuccessful one (incorrect prediction).

**4. Generation of Reflections:** Post-evaluation, the agent generates reflections in the form of a list of dictionaries, each containing ‘condition’ and ‘action’ keys with their respective values. These reflections encapsulate the agent’s reasoning and decision-making process in each prediction instance. The reflection is executed by the agent using a reflection prompt template. The template is available at Appendix C.

**5. Memory Insertion:** The ‘insert\_memory’ function processes each ‘condition’ in the reflection list, creating OpenAI embeddings for them. These embeddings, along with their associated actions, are then stored in the MemoryDB. This process facilitates the agent’s future recall and application of learned strategies and decisions.

An illustrative example demonstrating a complete cycle of this pipeline, specifically for the case of Ron Wyden’s interaction with Applied Materials (AMAT), is provided in the Appendix C. This example offers a practical insight into how the agent navigates through a single iteration of the prediction and reflection process.

### 6.4.3 Assessment of the LLM Agent’s Predictive Performance on Selected Cases

I tested the LLM Agent’s capabilities on a carefully chosen subset of 20 legislator-ticker pairs, focusing on its ability to predict transactional relationships. Given the expansive nature of the GPT-4-turbo API, a full-scale analysis was limited by time and cost constraints. This sample selection not only addressed these resource limitations — costing approximately \$31 in total, or an average of \$1.55 per inference — but also fit within a feasible time frame. The entire process took around 30 minutes, averaging approximately 1.5 minutes per case. This approach struck a balance between existent and non-existent transactional edges, allowing for a practical yet insightful analysis.

The top 10 Legislator-Ticker pairs, characterized by their unusually high excess returns as depicted in Figure 6, were chosen. To maintain diversity among the legislators and to simulate a realistic scenario, only one pair per unique legislator was included. Additionally, for each legislator, a ticker with which they had no prior transactions was randomly selected to create a hypothetical ‘negative’ edge. This approach not only facilitated a balanced examination of both positive and negative aspects of the transactional relationships but also allowed for a more fair understanding of the agent’s predictive accuracy. The selected cases are detailed in Table 4, providing an overview of the pairs analyzed and their respective

classifications in terms of transaction existence. Following the detailed presentation of cases in Table 4, the performance of the LLM Agent’s predictions are outlined in Table 5. This table presents the counts of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN), along with the calculated values for key performance indicators: Accuracy, Precision, Recall, and F1 Score.

Table 4: **Predictions on Legislator-Ticker Pairs using LLM Agent**

Legislator	Ticker	Ticker Name	Pred	Label	Eval
Ron Wyden	AMAT	Applied Materials	EXIST	EXIST	Accurate
Ron Wyden	BKNG	Booking Holdings	EXIST	NON-EXIST	Inaccurate
Angus King	QQQ	NASDAQ-100	EXIST	EXIST	Accurate
Angus King	CZR	Caesars Entertainment Inc	NON-EXIST	NON-EXIST	Accurate
Sheldon Whitehouse	IHI	iShares US Medical Devices ETF	EXIST	EXIST	Accurate
Sheldon Whitehouse	AVGO	Broadcom Inc.	NON-EXIST	NON-EXIST	Accurate
Timothy Kaine	VB	Vanguard Small-Cap Index Fund	NON-EXIST	EXIST	Inaccurate
Timothy Kaine	IJH	iShares Core S&P Mid-Cap ETF	EXIST	NON-EXIST	Inaccurate
David Perdue	DISCA	Warner Bros Discovery Inc	NON-EXIST	EXIST	Inaccurate
David Perdue	XBI	SPDR S&P Biotech ETF	EXIST	NON-EXIST	Inaccurate
Mike Rounds	VCR	Vanguard Consumer Discretionary ETF	EXIST	EXIST	Accurate
Mike Rounds	GLW	Corning Incorporated	EXIST	NON-EXIST	Inaccurate
Jerry Moran	FB	Facebook	NON-EXIST	EXIST	Inaccurate
Jerry Moran	RF	Regions Financial Corp	EXIST	NON-EXIST	Inaccurate
Benjamin Cardin	XLC	Communication Services Select Sector	EXIST	EXIST	Accurate
Benjamin Cardin	WMT	Walmart	EXIST	NON-EXIST	Inaccurate
Shelley Capito	MSFT	Microsoft Corporation	EXIST	EXIST	Accurate
Shelley Capito	SNE	Snap-On Inc	NON-EXIST	NON-EXIST	Accurate
Pat Roberts	WMT	Walmart	EXIST	EXIST	Accurate
Pat Roberts	CB	Chubb Limited	NON-EXIST	NON-EXIST	Accurate

Table 5: **Performance Metrics of the LLM Agent’s Predictions**

Metric	Value
True Positives (TP)	7
True Negatives (TN)	4
False Positives (FP)	6
False Negatives (FN)	3
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN} = \frac{7+4}{7+4+6+3} = 0.55$
Precision	$\frac{TP}{TP+FP} = \frac{7}{7+6} = 0.5385$
Recall	$\frac{TP}{TP+FN} = \frac{7}{7+3} = 0.7$
F1 Score	$2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \cdot \frac{0.5385 \times 0.7}{0.5385 + 0.7} = 0.6087$

The performance metrics of the LLM Agent, as detailed in Table 5, clearly indicate that its current level of predictive accuracy and precision is suboptimal. The low accuracy and precision suggest that the agent is not yet sufficiently reliable, particularly due to its tendency to produce a significant number of false positives. This performance shortfall underscores the need for further refinement and training of the agent, especially considering the limited size of the dataset used for its evaluation. It’s important to recognize, however, that this outcome is largely influenced by the nature and scale of the dataset used for the learning task. The agent’s training and evaluation were conducted on a relatively small set of

data, comprising just 20 cases.

However, it is crucial to highlight the LLM Agent’s innate ability to develop its own hypotheses and theorization through reflection. This capability is demonstrated by its continual generation of conditions, directing its focus on specific features by utilizing its memory module. This aspect of self-reflection and self-directed learning shows promise in the agent’s development, indicating a capabilities of reasoning and adaptability. In the following subsection, I will delve deeper into the contents of the memory module, exploring and investigating the hypotheses and theories formulated by the agent during its learning process. This examination will shed light on the agent’s internal reasoning mechanisms and its potential for growth and improved performance with further training and exposure to more extensive datasets.

#### 6.4.4 Exploring the LLM Agent’s Memory: Unraveling the Hypotheses and Theorization Capabilities of the LLM Agent

During the training involving 20 distinct pairs of legislator and ticker data, the LLM Agent successfully generated and stored 27 unique condition-action pairs within its memory module. This accumulation of pairs indicates the agent’s ability to recognize and adapt to a variety of scenarios. Below is one such example that illustrates how the agent operates:

##### Example of Condition-Action Pairs (Generated by LLM Agent)

**Condition:** When a publicly traded company has lobbied on bills that are assigned to the committees a legislator is a member of.

**Action:** Investigate the bills lobbied by the ticker and their assignment to the legislator’s committees using the following Cypher query: `MATCH (t:Ticker)-[:LOBBY_ON]->(b:Bill) MATCH (b)-[:ASSIGNED_TO]->(c:Committee) WHERE c.id = '<committee_id>' RETURN b.official_title, b.id`, and then predict 'EXIST' based on the relevance of the bills to the legislator’s committee assignments.

**Interpretation of the Cypher Query (By the Author):** This query extracts data from a graph database, focusing on the interaction between legislative committees and lobbying efforts by companies. Specifically, it identifies bills that a company (ticker) has lobbied for and which have been assigned to specific committees that a legislator is part of. By specifying a committee ID, the query returns the titles and IDs of relevant bills. This information is used to assess the potential

influence of the company’s lobbying efforts on the legislator, based on their committee involvement. If there’s significant relevance found between the lobbying activities and the legislator’s committee work, the agent predicts an ‘EXIST’ relationship, suggesting a possible transactional or influential connection.

For a clearer understanding of the LLM Agent’s decision-making, I used GPT-4 to summarize the key points from all 27 unique condition-action pairs identified in the study. This summary makes it easier to grasp the agent’s thought process and theories, providing insights in a more concise and digestible format:

## Summary of Memory Pairs

### Category: Legislative Committee and Ticker Lobbying Interactions

**Condition:** When a ticker (company) lobbies on bills related to a legislator’s committee assignments.

**Actions:**

- Investigate lobbyist activities and committee assignments.
- Predict ‘EXIST’ for a ‘BUY\_SELL’ relationship based on bill-ticker-legislator committee overlaps.

**Cypher Query:**

```
MATCH (t:Ticker)-[:LOBBY_ON]->(b:Bill)-[:ASSIGNED_TO]->(c:Committee),
      (l:Legislator)-[:COMMITTEE_ASSIGNMENT]->(c)
WHERE t.ticker = '<Ticker>' RETURN b.official_title, b.id
```

### Category: ETF Sector and Legislative Committee Alignment

**Condition:** When an ETF’s sector representation aligns with a legislator’s committee memberships.

**Actions:**

- Predict 'EXIST' for 'BUY\_SELL' relationships considering sector overlap.

**Cypher Query:**

```
MATCH (t:Ticker {ticker: '<ETF_Ticker>'})-[:BELONGS_TO]->(n:NAICS)

RETURN n.desc
```

**Category: Broad Indirect Connections**

**Condition:** Absence of direct 'BUY\_SELL' relationship data and lack of explicit NAICS code overlaps.

**Actions:**

- Explore broader indirect connections like historical trading patterns, personal interests, or market trends.

**Category: Industry-Specific Legislative Influence**

**Condition:** Legislators with transactions in specific NAICS categories and companies lobbying on related bills.

**Actions:**

- Investigate committee assignments relevant to the industry of a ticker.
- Predict 'EXIST' based on industry interests and legislative influence.

**Cypher Query:**

```
MATCH (l:Legislator)-[:COMMITTEE_ASSIGNMENT]->(c:Committee),

      (t:Ticker)-[:LOBBY_ON]->(b:Bill)

WHERE l.bioguide = '<bioguide>' AND t.ticker = '<Ticker>'

RETURN b.official_title, b.id
```

As demonstrated in the summary, the Agent shows an ability to autonomously create decision-making frameworks. It identifies specific conditions where it gathers pertinent information using Cypher queries

and makes decisions based on these criteria. Indeed, the Agent skillfully utilizes indirect connections found in the heterograph data, employing industry-level information and the links between firms’ lobbying efforts on bills and their connections to committees. Notably, the Agent also theorizes about ETFs, considering their industry-level connections in relation to the industry specializations of committees. This approach underlines the Agent’s capacity to navigate and interpret complex network relationships and leverage this understanding in its decision-making processes. Moreover, these theories provide support for the findings from Section 5, where the use of GNN and GNNExplainer highlighted industry-level information and committee assignments as key predictors, further reinforcing the significance of these factors in congressional stock trading analysis.

An additional advantage of using an LLM Agent over a GNN is its inherent understanding of specific entities, such as ETFs. For instance, the Agent already recognizes that the ticker “IHI” represents the “iShares U.S. Medical Devices ETF” and can utilize this knowledge directly, which wasn’t manually input by me. In contrast, in Section 5, the inclusion of detailed explanations for each ticker was impractical due to manual processing constraints. There, the GNN model relied solely on the ticker’s information in relation to its network connections, without any textual description of each ticker. However, with the LLM Agent, such meticulous pre-processing of data is unnecessary. This represents a significant benefit, as it reduces the need for extensive manual data preparation and allows the Agent to apply its pre-existing knowledge effectively in the analysis.

These capabilities mentioned in this section is not just a testament to the Agent’s advanced analytical abilities but also opens up new possibilities in the realm of theory building. By allowing the LLM to stack memories and reflect upon them, researchers can derive nuanced insights and form hypotheses based on the model’s internal decision-making patterns. This method showcases a novel way of leveraging Large Language Models for more than just data analysis, extending into the domain of theoretical development and hypothesis generation.

## 7 Conclusion and Future Directions

In this study, I delved into the dynamics of congressional stock investment, exploring what exactly influences these investment choices.

The first analysis aligned with the traditional financial literature’s approach of excess return, which

provided a direct estimation of possible excess return in congressperson-ticker-timing level. The estimation successfully addressed the range-censored limitation of the financial disclosure at the specific congress-ticker level, considering the life-cycle of transactions. This reconfirmed the findings of Eggers and Hainmueller (2013), which argued there was no widespread excess return among congressional investments. However, my findings indicate that such excess returns do exist at least abnormally and asymmetrically, more pronounced in the positive skewness compared to the negative returns. This suggests that some privileged information may drive such asymmetry in their excess return overall.

Secondly, I addressed a puzzling contradiction arising from the conclusions of Eggers and Hainmueller (2014), who found no substantial evidence that congresspersons disproportionately invested in stocks linked to their lobbying efforts and committee assignments. This outcome was somewhat counterintuitive, diverging from extensive research on the influence of committee assignments and congresspersons' specialization in areas under their committees' jurisdiction. To explore this further, I developed a novel graph-structured dataset, employing a heterograph to represent the multifaceted interactions between various entities involved in congressional activities. In this analysis, a critical insight emerged: the importance of "industry-level" information, which is garnered through committee involvement. This aspect was notably absent in Eggers and Hainmueller (2014), leading to their conclusion that committee assignments were not significant in congressional stock trading. However, my findings indicate that both committee assignments and industry-level information are crucial and should be considered together. By utilizing cross-entropy as a novel measure, I demonstrated that a congressperson's stock portfolio significantly mirrors, in terms of industry-code distribution, the stocks related to their assigned committees, as opposed to those they are not assigned to. This emphasizes the intertwined significance of committee roles and industry-specific knowledge in shaping congressional stock trading decisions.

Thirdly, I expanded on the work of Eggers and Hainmueller (2014) by employing a Graph Neural Network (GNN) to explore how factors like congressional activities, as captured in graph data, influence congresspersons' stock transactions. This approach led to a key discovery: committee assignments of congresspersons and lobbying activities of firms, contrary to the findings of Eggers and Hainmueller (2014), are significant predictors of their stock selections. The use of GNN in this context is crucial because it allows the model to learn feature representations by understanding how to aggregate information flowing through the network. This methodology is a substantial improvement over traditional approaches that rely on naive and oversimplified measurements, transforming complex network data into



more manageable but arbitrary forms for regression analysis. Additionally, to tackle the black-box nature of neural network predictions, I implemented GNNExplainer, an explainability tool that enhances the interpretability of the GNN model. This tool not only complements the evaluation metric but also provides semantically rich explanations for the decisions made by the model. It helps in understanding why a particular congressperson might choose certain stocks, based on the aggregated information from their committee assignments and the lobbying activities they are exposed to. This comprehensive approach not only refutes the findings of Eggers and Hainmueller (2014) but also demonstrates the necessity of using advanced methods like GNN to accurately interpret and predict congressional stock trading behavior.

Finally, I have implemented a Large Language Model (LLM) Agent for automated theorization in congressional stock trading analysis, introducing a pioneering approach to synthesizing theoretical hypotheses from intricate, high-dimensional datasets. The LLM Agent autonomously constructs decision-making frameworks by identifying specific conditions and gathering relevant information, effectively using the complex web of connections in the heterograph data. This includes skillfully leveraging industry-level information and the links between firms' lobbying on bills and committee assignments. Notably, the Agent's approach in theorizing about ETFs and committee specializations further reinforces the findings from other sections, particularly emphasizing the significance of industry-level information generated through committee assignments as a crucial factor in predicting and understanding congressional stock trading.

Adding onto these findings, it's worth examining the broader implications of our results. On a macro level, this research invites us to reassess our understanding of the motivations underpinning congressional service. The common perception is that congresspersons are primarily motivated by the goal of reelection, with their actions driven by a desire to serve their constituents and deliver policy outcomes that align with their promises and their party's platform. However, our findings suggest that the picture may be more complex, with financial considerations also playing a significant role. The evidence that some congresspersons are able to achieve outsized returns on their stock investments points to the potential for personal financial gain to be a motivating factor in their decisions and actions. This raises profound questions about the alignment of incentives in our political system and the possibility of conflicts of interest.

On a micro level, the analysis reveals fascinating variations in the behavior of congresspersons when it comes to stock trading. While some closely mirror the stock transactions related to their committee

assignments, others diverge considerably. This cautious behavior may be an attempt to avoid the appearance or reality of insider trading, a legal and ethical boundary that all congresspersons must navigate. Unraveling the sources of these behavioral differences presents an intriguing avenue for further research. Are these variations simply a reflection of individual personalities, strategic thinking, and risk tolerance? Or are they indicative of deeper systemic factors within our political and financial systems that are yet to be fully understood? Could the differences in trading patterns, for instance, be linked to the disparities in the level of scrutiny that different congresspersons face, their connections within the industry, or the financial literacy they possess? Do congresspersons with certain committee assignments have more access to non-public market-moving information? Uncovering these underpinnings would offer a more nuanced understanding of the complex interplay between politics and finance and could inform policy decisions to improve transparency and fairness in our political system.

## References

- Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *CoRR*, abs/1803.08375.
- Amari, S. (1993). Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4):185–196.
- Annaert, J., Van Osselaer, S., and Verstraete, B. (2009). Performance evaluation of portfolio insurance strategies using stochastic dominance criteria. *Journal of Banking & Finance*, 33(2):272–280.
- Asher, H. B. (1974a). Committees and the norm of specialization. *The Annals of the American Academy of Political and Social Science*, 411:63 – 74.
- Asher, H. B. (1974b). Committees and the norm of specialization. *The Annals of the American Academy of Political and Social Science*, 411:63–74.
- Bainbridge, S. (2010). Insider trading inside the beltway. Law-Econ Research Paper 10-08, UCLA School of Law.
- Barber, B. M. and Odean, T. (2000). Trading is hazardous to your wealth: The common stock investment performance of individual investors. *The Journal of Finance*, 55(2):773–806.
- Barberis, N. and Thaler, R. (2003). A survey of behavioral finance. In *Handbook of the Economics of Finance*, volume 1, pages 1053–1128. Elsevier.

- Bauer, M. and Rudebusch, G. D. (2014). The signaling channel for federal reserve bond purchases. *International Journal of Central Banking*.
- Białkowski, J., Darolles, S., and Le Fol, G. (2008). Improving vwap strategies: A dynamic volume approach. *Journal of Banking & Finance*, 32(9):1709–1722.
- Boros, H. S. and Fenno, R. F. (1968). *Administrative Law Review*, 20(2):335–337.
- Boxer, B., Isakson, J., Pryor, M., Roberts, P., Brown, S., Risch, J. E., Sassaman, J. C., and Gillis, A. (2012). Restrictions on insider trading under securities laws and ethics rules. United States Senate, Select Committee on Ethics.
- Buchanan, J. M. and Tollison, R. D., editors (1984). *The Theory of Public Choice - II*. University of Michigan.
- Buhrmester, V., Münch, D., and Arens, M. (2021). Analysis of explainers of black box deep neural networks for computer vision: A survey. *Machine Learning and Knowledge Extraction*, 3(4):966–989.
- Castelvecchi, D. (2016). Can we open the black box of ai? *Nature News*, 538(7623):20.
- Curry, J. M. (2019). Knowledge, expertise, and committee power in the contemporary congress. *Legislative Studies Quarterly*, 44(2):203–237.
- Cziraki, P., Lyandres, E., and Michaely, R. (2021). What do insiders know? evidence from insider trading around share repurchases and seos. *Journal of Corporate Finance*, 66:101544.
- Das, L., Sivaram, A., and Venkatasubramanian, V. (2020). Hidden representations in deep neural networks: Part 2. regression problems. *Computers & Chemical Engineering*, 139:106895.
- Dayhoff, J. E. and DeLeo, J. M. (2001). Artificial neural networks: opening the black box. *Cancer: Interdisciplinary International Journal of the American Cancer Society*, 91(S8):1615–1635.
- Duffie, D. and Dworczak, P. (2021). Robust benchmark design. *Journal of Financial Economics*, 142(2):775–802.
- Eggers, A. and Hainmueller, J. (2013). Capitol losses: The mediocre performance of congressional stock portfolios, 2004-2008. *Journal of Politics*, 75.

- Eggers, A. C. and Hainmueller, J. (2014). Political capital: Corporate connections and stock investments in the u.s. congress, 2004-2008. *ERN: Models of Political Processes: Rent-Seeking*.
- Fenno, Richard F., J. (1977). U.s. house members in their constituencies: An exploration. *The American Political Science Review*, 71(3):883–917.
- Féraud, R. and Clérot, F. (2002). A methodology to explain neural network classification. *Neural networks*, 15(2):237–246.
- Gardner, M. W. and Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric Environment*, 32:2627–2636.
- Gilligan, T. W. and Krehbiel, K. (1989). Asymmetric information and legislative rules with a heterogeneous committee. *American Journal of Political Science*, 33(2):459–490.
- Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. *CoRR*, abs/1704.01212.
- Hart, S. (1989). *Shapley value*. Springer.
- Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.
- Hayashi, H., Hu, Z., Xiong, C., and Neubig, G. (2019). Latent relation language models. *ArXiv*, abs/1908.07690.
- Henaff, M., Bruna, J., and LeCun, Y. (2015). Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*.
- Hoechle, D. and Zimmermann, H. (2007). A generalization of the calendar time portfolio approach and the performance of private investors. *Faculty of Business and Economics - University of Basel, Working papers*.
- Ivković, Z. and Weisbenner, S. (2005). Local does as local is: Information content of the geography of individual investors’ common stock investments. *The Journal of Finance*, 60(1):267–306.
- Jeng, L. A., Metrick, A., and Zeckhauser, R. (2003). Estimating the returns to insider trading: A performance-evaluation perspective. *Review of Economics and Statistics*, 85(2):453–471.

- Jerke, B. W. (2010). Cashing in on capitol hill: Insider trading and the use of political intelligence for profit. *University of Pennsylvania Law Review*, 158:1451–1523.
- Joshi, G., Walambe, R., and Kotecha, K. (2021). A review on explainability in multimodal deep neural nets. *IEEE Access*, 9:59800–59821.
- Kaushik, R., Shenoy, P., Bohannon, P., and Gudes, E. (2002). Exploiting local similarity for indexing paths in graph-structured data. In *Proceedings 18th International Conference on Data Engineering*, pages 129–140. IEEE.
- Kiewiet, D. R. and McCubbins, M. D. (1991). *The Logic of Delegation*. American Politics and Political Economy Series. University of Chicago Press, Chicago.
- King, D. C. (1994). The nature of congressional committee jurisdictions. *The American Political Science Review*, 88(1):48–62.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *ArXiv*, abs/2205.11916.
- Krehbiel, K. (1992). *Information and Legislative Organization*. Michigan Studies in Political Analysis. University of Michigan Press, Ann Arbor.
- Lampinen, A. K., Dasgupta, I., Chan, S. C. Y., Matthewson, K., Tessler, M. H., Creswell, A., McClelland, J. L., Wang, J. X., and Hill, F. (2022). Can language models learn from explanations in context? In *Conference on Empirical Methods in Natural Language Processing*.
- Lazer, D. M. J. (2011). Networks in political science: Back to the future. *PS: Political Science & Politics*, 44:61 – 68.
- Lenz, G. and Lim, K. (2009). Getting rich(er) in office? corruption and wealth accumulation in congress.
- Littlechild, S. C. and Owen, G. (1973). A simple expression for the shapley value in a special case. *Management Science*, 20(3):370–372.
- Lu, Y. and Lu, J. (2020). A universal approximation theorem of deep neural networks for expressing distributions. *CoRR*, abs/2004.08867.

- Luitse, D. and Denkena, W. (2021). The great transformer: Examining the role of large language models in the political economy of ai. *Big Data & Society*, 8.
- Madhavan, A. N. (2002). Vwap strategies. *Trading*, 2002(1):32–39.
- Mao, J., Yao, D., and Wang, C. (2013). A novel cross-entropy and entropy measures of ifss and their applications. *Knowledge-Based Systems*, 48:37–45.
- Mayhew, D. R. (1975). Congress: The electoral connection.
- Myers, B. W. (2009). Firms, politicians, and capital structure. Working Paper.
- Olden, J. D. and Jackson, D. A. (2002). Illuminating the “black box”: a randomization approach for understanding variable contributions in artificial neural networks. *Ecological modelling*, 154(1-2):135–150.
- Park, J. S., Popowski, L., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. (2022). Social simulacra: Creating populated prototypes for social computing systems.
- Patil, S. G., Zhang, T., Wang, X., and Gonzalez, J. E. (2023). Gorilla: Large language model connected with massive apis. *arXiv preprint arXiv:2305.15334*.
- Patterson, S. C. (1970). The professional staffs of congressional committees. *Administrative Science Quarterly*, 15(1):22–37.
- Price, D. E. (1978). Policy making in congressional committees: The impact of “environmental” factors. *American Political Science Review*, 72(2):548–574.
- Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y.-T., Lin, Y., Cong, X., Tang, X., Qian, B., Zhao, S., Tian, R., Xie, R., Zhou, J., Gerstein, M. H., Li, D., Liu, Z., and Sun, M. (2023). Toolllm: Facilitating large language models to master 16000+ real-world apis. *ArXiv*, abs/2307.16789.
- Radford, A. and Narasimhan, K. (2018). Improving language understanding by generative pre-training.
- Rauber, P. E., Fadel, S. G., Falcao, A. X., and Telea, A. C. (2016). Visualizing the hidden activity of artificial neural networks. *IEEE transactions on visualization and computer graphics*, 23(1):101–110.
- Román, J. H., Hulin, K. J., Collins, L. M., and Powell, J. E. (2012). Entity disambiguation using semantic networks. *Journal of the American Society for Information Science and Technology*, 63(10):2087–2099.

- Ruby, U. and Yendapalli, V. (2020). Binary cross entropy with deep learning technique for image classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9.
- Sarno, L. and Thornton, D. L. (2003). The dynamic relationship between the federal funds rate and the treasury bill rate: An empirical investigation. *Journal of Banking & Finance*, 27(6):1079–1110.
- Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2008). The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80.
- Schweizer, P. (2011). *Throw Them All Out: How Politicians and Their Friends Get Rich Off Insider Stock Tips, Land Deals, and Cronyism That Would Send the Rest of Us to Prison*. Houghton Mifflin Harcourt (HMH).
- Shanken, J. (1996). 23 statistical methods in tests of portfolio efficiency: A synthesis. *Handbook of statistics*, 14:693–711.
- Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., and Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning.
- Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.
- Simonovsky, M. and Komodakis, N. (2017). Dynamic edge-conditioned filters in convolutional neural networks on graphs. *CoRR*, abs/1704.02901.
- Sivakumar, K. and Waymire, G. (1994). Insider trading following material news events: Evidence from earnings. *Financial Management*, pages 23–32.
- Tahoun, A. (2014). The role of stock ownership by us members of congress on the market for political favors. *Journal of Financial Economics*, 111(1).
- Tang, J., Deng, C., and Huang, G.-B. (2016). Extreme learning machine for multilayer perceptron. *IEEE Transactions on Neural Networks and Learning Systems*, 27(4):809–821.
- Tang, J. and Liu, H. (2012). Feature selection with linked data in social media. In *SDM*.
- Thi, D. B. and Nguyen-Hoang, T.-A. (2013). Features extraction for link prediction in social networks. *2013 13th International Conference on Computational Science and Its Applications*, pages 192–195.

- Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., and Anandkumar, A. (2023). Voyager: An open-ended embodied agent with large language models.
- Ward, M. D., Stovel, K., and Sacks, A. (2011). Network analysis and political science. *Annual Review of Political Science*, 14:245–264.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E. H., Le, Q., and Zhou, D. (2022). Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903.
- Weiss, C. H. (1989). Congressional committees as users of analysis. *Journal of Policy Analysis and Management*, 8(3):411–431.
- Winter, E. (2002). The shapley value. *Handbook of game theory with economic applications*, 3:2025–2054.
- Wu, H., Yuan, Y., Wei, L., and Pei, L. (2018). On entropy, similarity measure and cross-entropy of single-valued neutrosophic sets and their application in multi-attribute decision making. *Soft Computing*, 22:7367–7376.
- Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. (2020). A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24.
- Yang, C., Wang, X., Lu, Y., Liu, H., Le, Q. V., Zhou, D., and Chen, X. (2023). Large language models as optimizers.
- Yang, Z., Ding, M., Zhou, C., Yang, H., Zhou, J., and Tang, J. (2020). Understanding negative sampling in graph representation learning. *CoRR*, abs/2005.09863.
- Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. (2023a). Tree of thoughts: Deliberate problem solving with large language models.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. (2023b). React: Synergizing reasoning and acting in language models.
- Ying, Z., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. (2019). Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32.
- Zhang, C., Song, D., Huang, C., Swami, A., and Chawla, N. V. (2019). Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 793–803.



- Zhang, M. and Chen, Y. (2018). Link prediction based on graph neural networks. *CoRR*, abs/1802.09691.
- Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J., and rong Wen, J. (2023). A survey of large language models. *ArXiv*, abs/2303.18223.
- Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI open*, 1:57–81.
- Zhu, X., Chen, Y., Tian, H., Tao, C., Su, W., Yang, C., Huang, G., Li, B., Lu, L., Wang, X., Qiao, Y., Zhang, Z., and Dai, J. (2023). Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory.
- Ziobrowski, A. J., Boyd, J. W., Cheng, P., and Ziobrowski, B. J. (2011). Abnormal returns from the common stock investments of members of the u.s. house of representatives. *Business and Politics*, 13(1):1–22.
- Ziobrowski, A. J., Cheng, P., Boyd, J. W., and Ziobrowski, B. J. (2004). Abnormal returns from the common stock investments of the u.s. senate. *The Journal of Financial and Quantitative Analysis*, 39(4):661–676.

## A Data Merging and Entity Disambiguation

One of the key challenges to make this graph-structured data is the effective disambiguation of entities, as the data is collected from multiple sources, including LobbyView, Senate/House Financial Disclosures, and naics.com. In this graph-structured dataset, entities such as Congresspersons and firms may appear under different names or expressions. For example, “Ron Wyden” may also be referred to as “Ron L. Wyden”, and “Apple” may appear as “Apple Inc.”. To accurately disambiguate these differing text representations of entities, it is essential to establish a unique identifier for each entity, regardless of the variations in their names.

Theoretically, matching entities based on text similarity between two datasets with  $n$  and  $m$  rows has a computational complexity of  $O(nm)$  (Román et al., 2012). Therefore, as the datasets grow larger, this complexity becomes prohibitively expensive. For instance, matching 70,000 firm names from LobbyView

to 4,000 firm names appearing in the ticker table would require 280,000,000 times of computations for text similarity. To address this challenge, I developed a novel approach that leverages URLs as unique identifiers for entities.

The approach involves acquiring the corresponding URL for each entity through Google searches, such as [https://en.wikipedia.org/wiki/Ron\\_Wyden](https://en.wikipedia.org/wiki/Ron_Wyden) for Ron Wyden and <https://www.apple.com/> for Apple, Inc. A key advantage of using URLs as unique identifiers is that they facilitate effective entity disambiguation. For example, if two different expressions, “Ron Wyden” and “Ron L. Wyden” are both assigned the same URL [https://en.wikipedia.org/wiki/Ron\\_Wyden](https://en.wikipedia.org/wiki/Ron_Wyden), we can confidently recognize that these two expressions refer to the same entity. This approach allows us to accurately consolidate information about entities that may be represented in various ways across different data sources. Additionally, this method reduces the computational complexity to  $O(n+m)$ , as only one query is required for each row of data. To further scale up this process, I parallelized the URL acquisition process by batching queries and distributing them across multiple servers available through commercial cloud services like AWS.

## **B Effective Parsing Technique for Financial Disclosures**

Financial Disclosures from the House are provided as encrypted PDF files. While text can be extracted from these files, the encryption results in irregular patterns, particularly in the tables that contain information about Congresspersons’ stock buying and selling activities. These irregular patterns make it challenging to parse the data using manually coded patterns, as the deviations are difficult to anticipate and account for. To address this challenge, I utilized OpenAI’s APIs, specifically the GPT-3.5 Turbo language model, to parse the PDFs into a CSV format that includes information such as when and who bought or sold which ticker, and how much.

The process involves querying the Large Language Model (LLM) with the extracted text from the PDFs and instructing the model to convert the irregularly formatted tables into structured CSV data which includes columns such as the date of the transaction, the name of the Congressperson, the ticker symbol of the stock, the type of transaction (buy or sell), and the amount of the transaction.

By leveraging the capabilities of the GPT-3.5 Turbo language model, I was able to effectively parse information contained in PDF files that would normally require manual human labor. This approach

significantly streamlines the data extraction process and ensures the accuracy and consistency of the parsed data.

In summary, this innovative approach to entity disambiguation through URL acquisition and parallelization enables efficient data merging from diverse sources, ensuring the accuracy and scalability of the analysis.

## C Appendix: Task & Reflection Prompt and Detailed Log from LLM Agent Performing Prediction and Reflection

### Task Prompt for the Agent to Perform ‘Make\_Prediction’

```
You are granted access to a graph database that encompasses interactions and
↪ activities related to the U.S. Congress, its Legislators, and publicly traded
↪ companies.

Your mission is to deduce the probability of a financial transaction between a
↪ designated U.S. Legislator, {Legis}, and a specific publicly traded company or
↪ ETF, {Company}, whose ticker is {Ticker} and name is {Company}.

Available Nodes:

- Legislator: Represents members of the U.S. Congress. Essential attributes include
↪ their 'name' and an identifier called 'bioguide'.

- Ticker: Symbolizes publicly traded companies or ETF on stock exchanges. Each
↪ company or ETF is uniquely identified by its 'ticker' symbol, a shorthand used on
↪ stock trading platforms, and it has additional property 'name', the official
↪ corporate designation or name of the ETF.

- Committee: Stands for the specialized committees within the U.S. Congress, each of
↪ which focuses on specific domains like finance, defense, health, or education.
↪ Crucial properties include 'name' and 'id'.
```

- NAICS: Corresponds to the North American Industry Classification System, which
  - ↪ categorizes businesses and industries. Pertinent attributes are 'ticker' and
  - ↪ 'desc', which describe the sector or industry where

a publicly traded company belongs.

- Bill: Denotes legislative bills processed within the U.S. Congress. Essential
  - ↪ attributes include 'summary', 'id', 'official\_title', and 'short\_title'.

#### Available Relationships:

- BUY\_SELL: Indicates a financial transaction or stock trading activity between a
  - ↪ U.S. Legislator and a publicly traded company, with date specifics. (Legislator
  - ↪ -> BUY\_SELL -> Ticker)
- COMMITTEE\_ASSIGNMENT: Relays the assignment of a U.S. Legislator to a specific
  - ↪ Committee within Congress for a set tenure. (Legislator -> COMMITTEE\_ASSIGNMENT
  - ↪ -> Committee)
- LOBBY\_ON: Chronicles the endeavors where a publicly traded company, via its Ticker,
  - ↪ lobbied a particular Bill. (Ticker -> LOBBY\_ON -> Bill). However, ETF does not
  - ↪ lobby on a bill like QQQ or SPY.
- BELONGS\_TO: Designates the industry or sector, as per the NAICS system, where a
  - ↪ publicly traded company operates. (Ticker -> BELONGS\_TO -> NAICS)
- ASSIGNED\_TO: Signals the allocation of a Bill to a distinct Committee within
  - ↪ Congress. (Bill -> Assigned\_to -> Committee)

#### [TASK]

Using the information, scrutinize the nodes and relationships to estimate the

- ↪ likelihood of a "BUY\_SELL" link between {Legis} and {Ticker}. You should give
- ↪ your guess (prediction) about a EXIST or NOT-EXIST of the transaction between
- ↪ {Legis} and {Ticker} as a final answer.

#### [Note]

1. For the purposes of this task, any existing direct "BUY\_SELL" relationships
  - ↪ between {Legis} and {Ticker} have been intentionally deleted from graph database.
  - ↪ Your goal is to infer the likelihood of such a relationship based on indirect
  - ↪ data points.

Therefore, make sure that

- 1. You don't need to check whether there exist a direct "BUY\_SELL" relationship
  - ↪ between {Legis} and {Ticker} because it's already and intentionally deleted
  - ↪ if exists.
- 2. You should "guess" a EXIST or NOT-EXIST of the transaction between {Legis}
  - ↪ and {Ticker} as a final answer.

2. When you fetch the bill node, do not fetch the `summary` attribute of the node
  - ↪ because it's too long and this will deplete your Token limits to conduct the task
  - ↪ successfully. Instead, you should fetch the `official\_title` attribute of the
  - ↪ node to understand the bill's content.

#### [Harnessing the Memory Vector Database]

You have a Memory Vector Database that encapsulates key strategies derived from your

- ↪ history of analyzing legislator-company stock transactions. This tool operates on
- ↪ a principle of condition-action pairs:

Condition: A specific scenario or set of circumstances from past tasks.

Action: The strategic response that was effective in that scenario.

When faced with a new task, the database can be leveraged as follows:

Describe the Condition: Input a description of the current task's conditions into the

- ↪ database. Then the database uses vector similarity to find past conditions that
- ↪ closely match your description.

It then presents you with a list of condition-action pairs that were effective in

- ↪ similar past scenarios. This process helps you to quickly identify the most
- ↪ relevant strategies for the current task, drawing on successful experiences to
- ↪ inform your decision-making.

The following is a list of condition-action pairs that you may be able to find in the

- ↪ database:

[Example 1]

Condition: Scenarios where a legislator has multiple overlapping NAICS codes

- ↪ with the tickers they've engaged in transactions with.

Action: In such scenarios, confidently assert the existence of a

- ↪ relationship.

[Example 2]

Condition: Situations where there are no overlapping NAICS codes between a

- ↪ ticker and the legislator's transactions.

Action: In these instances, scrutinize any bill titles lobbied by the ticker

- ↪ that are pertinent to the committees the legislator belongs to.

In addition, make sure you start your chain-of-action by using this "Memory" tool by

- ↪ describing your task so that you can more faster and more accurately solve the
- ↪ task.

## Detailed Log of the Agent's Make\_Prediction Process: Ron Wyden - Applied Materials (AMAT)

> Entering new AgentExecutor chain...

To begin with, I need to assess the indirect data points that could suggest a

- ↪ financial transaction between Ron Wyden and AMAT. Since direct "BUY\_SELL"
- ↪ relationships have been intentionally deleted, I should look for overlapping
- ↪ interests, such as committee assignments, bills lobbied by AMAT, and NAICS codes
- ↪ that might indicate a connection between the legislator and the company.

Action: MemoryDB Query

Action Input: I am trying to predict whether a "BUY\_SELL" relationship exists between

- ↪ Ron Wyden and AMAT based on indirect data points such as committee assignments,
- ↪ bills lobbied by AMAT, and overlapping NAICS codes, despite the direct
- ↪ relationship being intentionally deleted from the database.

Observation: [{'condition': "Predicting a 'BUY\_SELL' relationship between a

- ↪ legislator and an ETF based on indirect data points, where the ETF targets
- ↪ multiple companies in the same industry or theme.", 'action': "Consider the
- ↪ legislator's committee assignments, the bills lobbied by the ETF, and the sectors
- ↪ or industries where the ETF operates. Check for any overlap between the
- ↪ committees that the legislator is assigned to and the bills that the ETF has
- ↪ lobbied on. Also check if the legislator has had any financial transactions with
- ↪ companies in the same industry as the ETF."}, {'condition': "When there is a lack
- ↪ of data or relationships to infer the likelihood of a 'BUY\_SELL' link between a
- ↪ legislator and a company.", 'action': "In these instances, consider other factors
- ↪ such as the legislator's overall trading activity, the company's overall lobbying
- ↪ activity, and the general trends in the legislator's transactions and the
- ↪ company's lobbying activities. Use these factors to make an educated guess about
- ↪ the likelihood of a 'BUY\_SELL' link."}]

Thought:Based on the observations from the Memory tool, I should consider Ron Wyden's

- ↪ committee assignments, the bills lobbied by AMAT, and the NAICS codes related to
- ↪ AMAT. I should check for any overlap between the committees that Ron Wyden is
- ↪ assigned to and the bills that AMAT has lobbied on. Additionally, I should
- ↪ consider if Ron Wyden has had any financial transactions with companies in the
- ↪ same industry as AMAT. Since direct "BUY\_SELL" relationships are intentionally
- ↪ deleted, I need to use these indirect data points to make an educated guess about
- ↪ the likelihood of a "BUY\_SELL" relationship.

Action: GraphDB Query

Action Input:

```
MATCH (l:Legislator {name: 'Ron
↪ Wyden'})-[:COMMITTEE_ASSIGNMENT]->(c:Committee)<-[:ASSIGNED_TO]-(b:Bill)<-[:LOBBY_ON]-(t:Ticker
↪ {ticker: 'AMAT'})

RETURN DISTINCT c.name, b.official_title
```

Observation: [[{'c.name': 'Senate Committee on the Judiciary', 'b.official\_title':

- ↪ 'To amend the Immigration and Nationality Act to eliminate the per-country
- ↪ numerical limitation for employment-based immigrants, to increase the per-country
- ↪ numerical limitation for family-sponsored immigrants, and for other purposes.'}

, {'c.name': 'Senate Committee on Finance', 'b.official\_title': 'A bill to amend the

- ↪ Internal Revenue Code of 1986 to establish the semiconductor manufacturing
- ↪ investment credit.'},

{'c.name': 'House Committee on Energy and Commerce', 'b.official\_title': 'To restore

- ↪ American leadership in semiconductor manufacturing by increasing Federal
- ↪ incentives in order to enable advanced research and development, secure the
- ↪ supply chain, and ensure long-term national security and economic
- ↪ competitiveness.'},



```

{'c.name': 'Senate Committee on Finance', 'b.official_title': 'A bill to restore
↳ American leadership in semiconductor manufacturing by increasing Federal
↳ incentives in order to enable advanced research and development, secure the
↳ supply chain, and ensure long-term national security and economic
↳ competitiveness.'},
{'c.name': 'House Committee on Energy and Commerce', 'b.official_title': 'To
↳ modernize and strengthen the Committee on Foreign Investment in the United States
↳ to more effectively guard against the risk to the national security of the United
↳ States posed by certain types of foreign investment, and for other purposes.'},
{'c.name': 'Senate Committee on Finance', 'b.official_title': 'A bill to amend the
↳ Internal Revenue Code of 1986 to enhance tax benefits for research activities.'},
↳ {'c.name': 'Senate Committee on the Judiciary', 'b.official_title': 'A bill to
↳ amend the Immigration and Nationality Act to eliminate the per-country numerical
↳ limitation for employment-based immigrants, to increase the per-country numerical
↳ limitation for family-sponsored immigrants, and for other purposes.'},
{'c.name': 'House Committee on Energy and Commerce', 'b.official_title': 'To
↳ establish a new Directorate for Technology and Innovation in the National Science
↳ Foundation, to establish a regional technology hub program, to require a strategy
↳ and report on economic security, science, research, innovation, manufacturing,
↳ and job creation, to establish a critical supply chain resiliency program, and
↳ for other purposes.'},
{'c.name': 'Senate Committee on Commerce, Science, and Transportation',
↳ 'b.official_title': 'A bill to establish a new Directorate for Technology and
↳ Innovation in the National Science Foundation, to establish a regional technology
↳ hub program, to require a strategy and report on economic security, science,
↳ research, innovation, manufacturing, and job creation, to establish a critical
↳ supply chain resiliency program, and for other purposes.'},

```

```
{'c.name': 'Senate Committee on the Judiciary', 'b.official_title': 'A bill to amend
↳ the Immigration and Nationality Act to authorize additional visas for
↳ well-educated aliens to live and work in the United States, and for other
↳ purposes.'},
{'c.name': 'House Committee on Energy and Commerce', 'b.official_title': 'To provide
↳ for a coordinated Federal research initiative to ensure continued United States
↳ leadership in engineering biology.'}]}
```

Thought: The query results show that there are bills related to semiconductor

↳ manufacturing and technology innovation that have been lobbied on by AMAT and are

↳ assigned to committees that Ron Wyden is a member of, particularly the Senate

↳ Committee on Finance. This suggests that there is an overlap in the interests of

↳ Ron Wyden and AMAT, which could be an indicator of a potential "BUY\_SELL"

↳ relationship.

Next, I should check if Ron Wyden has had any financial transactions with companies

↳ in the same industry as AMAT to further support the likelihood of a "BUY\_SELL"

↳ relationship.

Action: GraphDB Query

Action Input:

```
MATCH (l:Legislator {name: 'Ron
↳ Wyden'})-[:BUY_SELL]->(t:Ticker)-[:BELONGS_TO]->(n:NAICS)
WHERE EXISTS((:Ticker {ticker: 'AMAT'})-[:BELONGS_TO]->(n))
RETURN DISTINCT t.ticker, n.desc
```

Parsing Cypher queries...

```
Input text: MATCH (l:Legislator {name: 'Ron
↳ Wyden'})-[:BUY_SELL]->(t:Ticker)-[:BELONGS_TO]->(n:NAICS)
WHERE EXISTS((:Ticker {ticker: 'AMAT'})-[:BELONGS_TO]->(n))
RETURN DISTINCT t.ticker, n.desc
```

Observation: [[{'t.ticker': 'AVGO', 'n.desc': 'Semiconductor and Related Device  
↪ Manufacturing'}, {'t.ticker': 'INTC', 'n.desc': 'Semiconductor and Related Device  
↪ Manufacturing'}, {'t.ticker': 'NVDA', 'n.desc': 'Semiconductor and Related Device  
↪ Manufacturing'}]]

Thought: The results indicate that Ron Wyden has engaged in financial transactions  
↪ with other companies in the semiconductor and related device manufacturing  
↪ industry, which is the same industry as AMAT. This further supports the  
↪ likelihood of a "BUY\_SELL" relationship between Ron Wyden and AMAT, as there is a  
↪ pattern of interest in this sector.

Given the overlap in committee interests and the legislator's previous financial  
↪ transactions within the same industry as AMAT, I can make an educated guess about  
↪ the existence of a "BUY\_SELL" relationship.

Final Answer: Based on the indirect data points such as committee assignments related  
↪ to technology and semiconductor manufacturing, bills lobbied by AMAT that are  
↪ assigned to these committees, and Ron Wyden's financial transactions with other  
↪ companies in the same industry, it is likely that a "BUY\_SELL" relationship  
↪ between Ron Wyden and AMAT exists (EXISTS).

> Finished chain.

## Reflection Prompt for the Agent to Perform ‘Generate Reflection’

Previously, I presented you with a link prediction task:

[Previous TASK]-----

{Task Prompt}

-----  
  
Your response was:

[LOG]-----

{Log from Make\_Prediction}  
-----

% Either one of the below two lines based on eval:

Your response was ACCURATE, as there is indeed a "BUY-SELL" relationship between  
↪ "{Legis}" and "{Ticker}".

Your response was INACCURATE, as there exists a "BUY-SELL" relationship between  
↪ "{Legis}" and "{Ticker}".

Moving on, your next objective involves self-reflection:

[TASK]

Building upon your failure with the previous task, take a moment to consider the

↪ key factors that contributed to your failure. Then reflect on how and when

↪ certain strategies can be applied in future scenarios to avoid such failure.

Now that you know the answer of the task, you can try multiple possible queries

↪ using the tool GraphDB Query to see which features of the data could have

↪ helped you to solve the task.

Then, structure your self-assessment in two parts:

Condition: Identify specific conditions under which the strategy you came up with  
↪ reflection employed.

Action: Outline the steps of the strategy to be employed under these conditions.

This analysis will serve as a guide for tackling similar tasks in the future, where

↪ "similarity" is determined by the conditions you delineate. You will be  
↪ provided with a new task later on, along with a tool to detect conditions akin  
↪ to those you've described by assessing vector similarity, enabling you to  
↪ retrieve and apply the successful action from your previous experience.

For example:

#### [Example 1]

Condition: Scenarios where a legislator has multiple overlapping NAICS

↪ codes with the tickers they've engaged in transactions with.

Action: In such scenarios, confidently assert the existence of a

↪ relationship.

#### [Example 2]

Condition: Situations where there are no overlapping NAICS codes between a

↪ ticker and the legislator's transactions.

Action: In these instances, scrutinize any bill titles lobbied by the

↪ ticker that are pertinent to the committees the legislator belongs to.

However, as you reflect on the task you've just completed, distill a novel

- ↪ condition-action pair that captures a new understanding or strategy not
- ↪ previously recorded in your memory. This pair should be informed by unique
- ↪ aspects or outcomes of the current task. If no new insights have emerged or if
- ↪ the task's conditions do not warrant a fresh approach distinct from existing
- ↪ entries, please return an empty list.

#### [Output Format]

We want your answer to be jsonl format with the list of your pair of "Condition"

↪ and "Action" as follows:

[{

"condition": "Instances where a legislator has several overlapping NAICS codes

↪ among the tickers they've transacted with.",

"action": "In such a data landscape, confidently indicate 'EXIST'."

}}, ... ] # store multiple pairs of "Condition" and "Action" in a list if you have

↪ multiple pairs of "Condition" and "Action" to store in your memory.

## Detailed Log of the Agent's 'Generate\_Reflection' Process: Ron Wyden-Applied Materials (AMAT)

> Entering new AgentExecutor chain...

To complete the self-assessment, I need to identify the conditions under which the

↪ strategy I previously employed can be effectively utilized and outline the

↪ steps of the strategy to be employed under these conditions.

Action: Memory

Action Input: I need to reflect on the key factors that contributed to the successful

- ↪ prediction of a "BUY\_SELL" relationship between Ron Wyden and AMAT, and how to
- ↪ apply similar strategies in future scenarios with similar conditions.

Observation: [{ 'condition': "Predicting a 'BUY\_SELL' relationship between a

- ↪ legislator and an ETF based on indirect data points, where the ETF targets
- ↪ multiple companies in the same industry or theme.", 'action': "Consider the
- ↪ legislator's committee assignments, the bills lobbied by the ETF, and the sectors
- ↪ or industries where the ETF operates. Check for any overlap between the
- ↪ committees that the legislator is assigned to and the bills that the ETF has
- ↪ lobbied on. Also check if the legislator has had any financial transactions with
- ↪ companies in the same industry as the ETF."}, { 'condition': "When there is a lack
- ↪ of data or relationships to infer the likelihood of a 'BUY\_SELL' link between a
- ↪ legislator and a company.", 'action': "In these instances, consider other factors
- ↪ such as the legislator's overall trading activity, the company's overall lobbying
- ↪ activity, and the general trends in the legislator's transactions and the
- ↪ company's lobbying activities. Use these factors to make an educated guess about
- ↪ the likelihood of a 'BUY\_SELL' link."}, { 'condition': "When predicting a
- ↪ 'BUY\_SELL' relationship between a legislator and a ticker, and there are no
- ↪ indirect relationships between them in the graph database.", 'action': "In these
- ↪ instances, consider the possibility of a direct 'BUY\_SELL' relationship between
- ↪ the legislator and the ticker. Use a query to check for such a direct
- ↪ relationship. If there is a direct 'BUY\_SELL' relationship, confidently assert
- ↪ the existence of a relationship."}]

Thought: I now have a list of condition-action pairs that were effective in similar

- ↪ past scenarios. I can use these to create a self-assessment of the strategy I
- ↪ employed in the current task.

Final Answer:

```

```jsonl
[
  {
    "condition": "When a legislator's committee assignments overlap with bills
↳ lobbied by a company.",
    "action": "Use a Cypher query to identify any overlap between the committees
↳ a legislator is assigned to and the bills a company has lobbied on.
If overlap exists, this suggests a potential 'BUY_SELL' relationship. Cypher
↳ query example: MATCH
↳ (l:Legislator)-[:COMMITTEE_ASSIGNMENT]->(c:Committee)<-[:ASSIGNED_TO]-(b:Bill)
<-[:LOBBY_ON]-(t:Ticker) RETURN DISTINCT c.name, b.official_title"
  },
  {
    "condition": "When a legislator has engaged in financial transactions with
↳ companies in the same industry as another company.",
    "action": "Use a Cypher query to check if a legislator has had financial
↳ transactions with companies in the same NAICS industry as the company in
↳ question. If transactions exist, this indicates a higher likelihood of a
↳ 'BUY_SELL' relationship. Cypher query example: MATCH
↳ (l:Legislator)-[:BUY_SELL]->(t:Ticker)-[:BELONGS_TO]->(n:NAICS) WHERE
↳ EXISTS((:Ticker)-[:BELONGS_TO]->(n)) RETURN DISTINCT t.ticker, n.desc"
  }
]
```

> Finished chain.

```