Word Count: 1672

Introduction

Investor behaviour has long played a role in asset price formation, with recent research focusing on the measurable effects of investor attention. As digital search activity becomes a critical component of information gathering, tools like Google Trends provide an opportunity to quantify public interest in real time.

This analysis will answer the question "Does Google Trends Search Interest Predict Stock Returns in the Short Term Among Fortune 25 Companies?". Using a fixed-effects panel regression framework, I will analyse the relationship between google search interest and quarterly returns over the period between 2010 and 2025. Through a combination of search and financial data, I aim to test the value of search interest as an indicator of price movement, even among the most mature firms in the economy.

Literature Review

Investor attention has emerged as a non-traditional factor in asset-pricing, particularly the use of online data. Da, Engelberg, and Gao (2011) pioneered this approach by introducing Google Search Volume Index (SVI) as a proxy for investor attention, observing an association between search interest and short-term return predictability. Similarly, Bank, Larch, and Peter (2011) showed that abnormal search volume can forecast excess returns in European markets. These findings provide a crucial background; however, results are often sensitive to sector, time horizon, and sentiment. This analysis will build on past studies through sectoral-level analysis, using fixed-effects regression to isolate firm-specific dynamics.

Methodology

Data Selection & Collection

To analyse the correlation between google trends data and stock prices, I decided to choose the top 25 companies currently on the fortune 500. (Fortune, 2024). As these companies represent the largest proportion of market-share and encompass several sectors.

Google trends data was downloaded directly from their website, encompassing the period from 01/01/2010 to 01/01/2025. (Google Trends, 2025)**.** Google Trends data consists of a 0–100-point scale measuring 'search interest'.

Historical stock market data was sourced from Yahoo Finance, and was taken from the same period. (Yahoo Finance, 2025). It is important to note that not every company on the list has been trading publicly for the entire period.

I then stitched together all the data for each company into a single file using python. (See appendix)

Data Analysis

Analysis was entirely conducted using python libraries including Pandas, NumPy and StatsModels. The stock price and search interest data was loaded and reshaped into long format to facilitate time-series analysis.

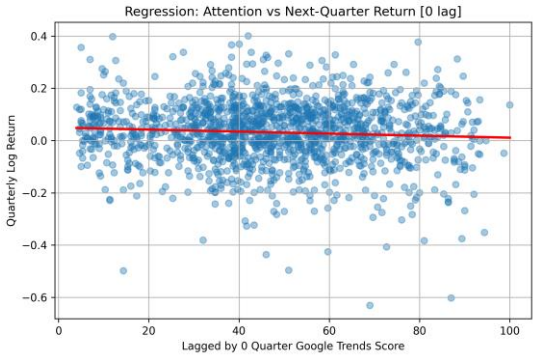
To stabilise variance, reduce skew, and compare performance across stocks with various price scales, I then calculated quarterly log returns by taking the difference in the log of the stock price quarter to quarter within groups defined by their stock ticker.

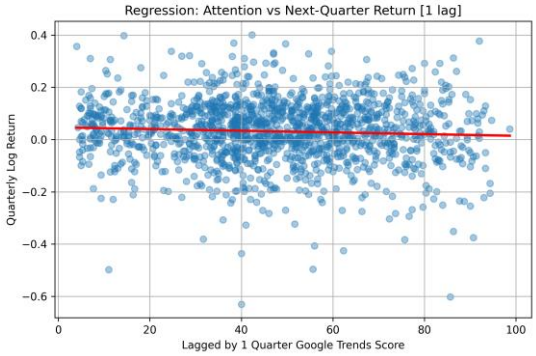
To identify the delayed impacts of search interest on stock returns, analysis was conducted using a lagged attention variable. Essentially comparing log return to search interest with various levels of offset. This was carried out for lag ranging from 0-4 quarters.

I then used the Ordinary Least Squares (OLS) regression model to incorporate stock-specific and time-specific fixed effects, with the aim of controlling for unobserved heterogeneity and temporal trends. Additionally,

clustered standard errors were implemented to correct for autocorrelation within individual stock tickers, thus enhancing the robustness of the statistical inference. (All code can be found in the appendix)

General Analysis with Varied Lag





Above are the plots for quarterly log return against lagged google trends scores for the two most statistically significant lag levels. Refer to the 'Graphs' section of the Appendix for all plots.

lag	coef	pval	r2
0	-0.000598	0.000064	0.2624
1	-0.000399	0.003224	0.2603
2	-0.000358	0.005834	0.2586
3	-0.000199	0.177993	0.2576
4	-0.000359	0.013713	0.2587

Above is a table showing the results of the regression test at each level of lag, there is the strongest correlation at zero lag. Across all levels of lag, it is observable that there is some negative impact of search interest on stock prices.

Observations

From the data, it is clear that there is a general correlation between search interest and stock price changes. The most significant windows are immediate and within 1 quarter. Both of which have a p value below 0.01, indicating

statistical significance at a 1% level. Their R^2 values are similar at roughly 0.26, indicating a moderate impact, as roughly 26% of variance in stock price is explainable by search interest.

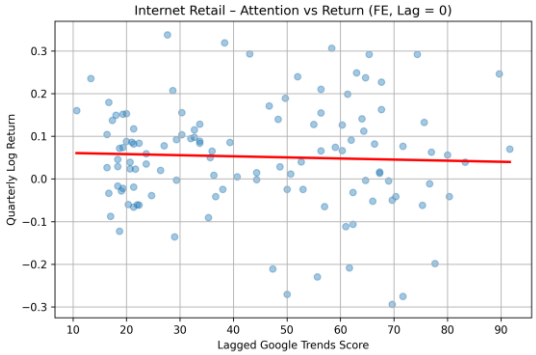
It is interesting that their coefficients are both negative, indicating a negative correlation between search interest and stock price. This may be explained by the fact that search interest is increased by negative coverage of companies more than it is by positive coverage. This is a valuable insight, it shows that consumer/investor interest in a company is not necessarily positive interest.

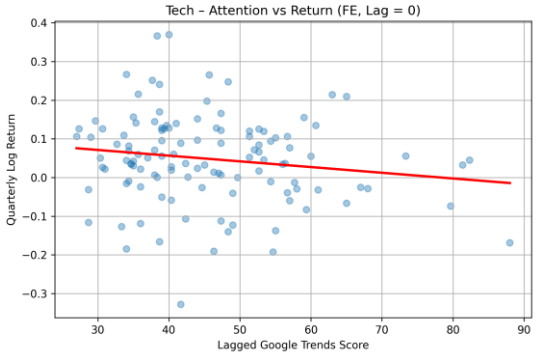
Methodology Continued

After exploratory analysis identified that a lag of 0 provided the strongest and most statistically significant predictive power, I selected this lag for a detailed sector-by-sector evaluation. The goal of a sector specific evaluation was to enhance the specificity and applicability of insights. Stocks were mapped to industry sectors and sectors with fewer than 30 observations or fewer than three unique tickers (companies) were excluded to ensure sufficient statistical validity.

Similar OLS regression modelling to earlier was used, maintaining consistent controls and clustering strategies. The results of sector-by-sector analysis are displayed in the following section. (Code found in appendix)

Sector Analysis with Zero Lag



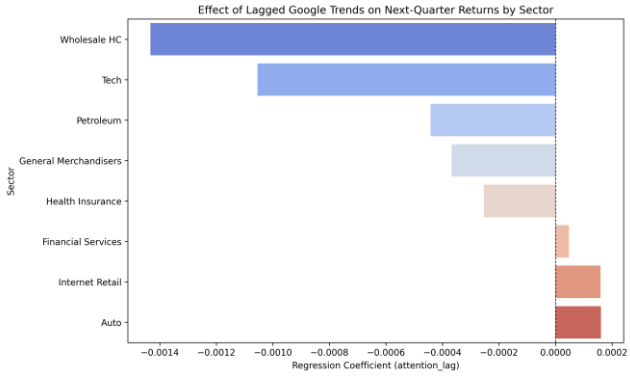


Above are the plots for quarterly log return against zero-lag google trends scores for the two most statistically significant sectors, Technology, and Internet Retail. Plots for other sectors may be found in the appendix.

sector	coef	pval	r2
Tech	-0.00105	~0.000000	0.7567538980017445
Internet Retail	0.00016	~0.000000	0.7749495267751321
Petroleum	-0.000440	0.001609	0.7525893660281782
Health Insurance	-0.000250	0.390977	0.6646704565440116

Financial Services	0.000047	0.521184	0.7523698148483239
General Merchandisers	-0.000370	0.728089	0.3980707428062471

Above is a table with the results of the regression calculations for each sector. Technology and Internet Retail are the sectors with the most significant correlation between trends activity and stock price. While there are other sectors with larger coefficients, Internet Retail has a low p-value and according to its R^2 value, the largest proportion of change in stock price explainable by trends data.



Above is a bar chart showing the coefficients for several sectors, visualising the large difference in search interest impact across sectors. Note that wholesale HC and Auto are included on the bar chart, but were not included in the table of summary statistics as they did not produce a p-value.

Observations

From the data presented, there is visibly a large disparity in impact across sectors. Technology and Internet Retail seem to be the most impacted by search interest in a zero-lag timeframe. Internet Retail is the only sector with a statistically significant positive correlation between search interest and stock price, this may be due to the fact that search interest in an online retailer has a higher chance of being genuine interest in consumption (indicated by its R^2 of 0.775, the highest); search interest in other industries is more likely attributed to news or other factors.

Petroleum is also worth discussing, the sector showed a strong R^2 value and a sub-5% significance level, so while its coefficient was not as high as Technology or Wholesale HC, its variance can be strongly attributed to search interest. Its negative regression coefficient strongly suggests that search interest is a negative for petroleum companies, likely due to climate scrutiny and other scandals.

General Merchandisers and Financial Services each have very high p-values of 0.73 and 0.52 respectively, this indicates that the variable the regression was performed on (search interest) was unlikely to be the cause behind any variance. Observations were likely due to chance or confounding factors not considered in this analysis.

General vs Sector Specific

When compared in the same time-frame of 0 lag, there are clear differences between our general analysis and sector-specific analysis.

Primarily, the disparity in R^2 values. General analysis returned a value of ~0.26 in this time frame, a moderate proportion of variance attributable to search interest. But each sector analysed returned values much higher. This can be attributed to the homogenous nature of each sector, patterns can be more consistently explained by search

interest; In general analysis, it is less easy to explain variation through search interest due to the diverse nature of the dataset.

Additionally, there is a disparity in p values. Apart from Technology and Internet Retail, every other sector analysed has a much higher p value than that of the general analysis. The disparity between the other sectors and the general analysis can be explained by the fact that the dataset in the general analysis is much larger. Technology and Internet Retail have a very high correlation between search interest and stock price changes, it is large, stable, and significant. This means they have extremely low p values, and are likely the main drivers of the low general p value.

There is no clear, large difference between the general regression coefficient and the sector coefficients as a whole; they likely average out to form the general regression coefficient. However, it is notable that Internet Retail is distinct in its positive coefficient (apart from Financial Services, however that sector has a high p value and is thus not statistically significant). This indicates that while in general search interest predicts a decrease in stock price, Internet Retail is unique as a sector. Likely due to Internet Retail being retail on the internet, any consumption is associated with a google search. This naturally brings us onto the limitations of our model.

Conclusion

In this analysis, I have explored the predictive relationship between search interest (through Google Trends data), and short-term stock returns among the Fortune 25. Using a fixed-effects panel regression framework, it was found that search interest is a statistically significant predictor of returns in general, with stronger effects in specific sectors such as Technology and Internet Retail. It is notable that in general analysis, I found that the effect is strongest within one quarter, and that that effect was a negative return.

However, the direction and strength of the correlation varied by sector, and most sectors did not show significant correlation in isolation. This may be due to the limited number of companies in the dataset. But it could also be due to confounding variables, an endogenous relationship between stock price and search interest, or a lack of sentiment analysis. This approach was limited, but still yielded statistically significant insights into the link between search interest and stock returns.

APPENDIX: Code

CSV Stitcher for Trends, Similar used for Stock	Code for General Regression & Plotting

```python
# === Google Trends Data Aggregation Script ===
# This script loads monthly Google Trends CSVs, extracts data from the correct header row,
# cleans and aligns them, averages values by quarter, and combines them into a single DataFrame.

import pandas as pd
import glob
import os

csv_dir = r"C:\Users\#########\Documents\QM Project stuff\trends CSVs"
csv_files = glob.glob(os.path.join(csv_dir, "*.csv"))

trend_data = {}

# Loop through each file in the folder
for file in csv_files:
    # Use filename (minus .csv) as the column name
    trend_name = os.path.splitext(os.path.basename(file))[0]

    # Identify the line that contains the actual header
    header_line_index = None
    with open(file, 'r', encoding='utf-8') as f:
        for i, line in enumerate(f):
            if "Month" in line:
                header_line_index = i
                break
    if header_line_index is None:
        continue  # skip file if no "Month" found

    # Read the CSV, starting from the correct header line
    df = pd.read_csv(file, sep=",", skiprows=header_line_index, engine="python")
    df.columns = df.columns.str.strip()  # clean column names

    if "Month" not in df.columns:
        continue  # skip if structure is unexpected

    # Rename for consistency
    df.rename(columns={"Month": "Date"}, inplace=True)
    df["Date"] = pd.to_datetime(df["Date"], format="%Y-%m", errors="coerce")

    # Identify the search interest column (assumes only one besides Date)
    trend_cols = [col for col in df.columns if col != "Date"]
    if not trend_cols:
        continue

    trend_col = trend_cols[0]
    df[trend_col] = pd.to_numeric(df[trend_col], errors="coerce")
    df.set_index("Date", inplace=True)

    # Resample monthly data into quarterly averages
    quarterly_avg = df.resample("Q").mean()
    quarterly_avg.rename(columns={trend_col: trend_name}, inplace=True)

    # Store cleaned and averaged data
    trend_data[trend_name] = quarterly_avg[trend_name]

# Combine all trends into a single DataFrame, indexed by quarter-end
combined_df = pd.DataFrame(trend_data)
combined_df.index.name = "Quarter_End"

# Export to Excel for use in regression analysis
combined_df.to_excel("combined_trends_quarterly_avg.xlsx")
```

Sector-By-Sector Regression and Plotting 1

```python
# === General Analysis: Google Trends vs Stock Returns ===
# This script runs a fixed-effects regression to test whether lagged Google Trends
# scores predict next-quarter log returns across all Fortune 25 companies.

import pandas as pd
import numpy as np
import statsmodels.formula.api as smf
import seaborn as sns
import matplotlib.pyplot as plt

# Set lag length to use in the regression
lag = 4  # Number of quarters to lag attention data

# Load combined quarterly data
trends = pd.read_excel("combined_trends.xlsx")
prices = pd.read_excel("combined_stocks.xlsx")

# Convert from wide to long format for merging
trends_long = trends.melt(id_vars="Quarter_End", var_name="ticker", value_name="attention")
prices_long = prices.melt(id_vars="Quarter_End", var_name="ticker", value_name="price")

# Merge Trends and Price data
df = pd.merge(trends_long, prices_long, on=["Quarter_End", "ticker"], how="inner")
df["Quarter_End"] = pd.to_datetime(df["Quarter_End"])
df = df.sort_values(["ticker", "Quarter_End"])

# Compute log returns and lagged attention scores
df["log_return"] = df.groupby("ticker")["price"].transform(lambda x: np.log(x) - np.log(x.shift(1)))
df["attention_lag"] = df.groupby("ticker")["attention"].shift(lag)

# Drop rows with missing values (usually early quarters)
df_clean = df.dropna(subset=["log_return", "attention_lag"])

# Create quarter identifiers for fixed effects
df_clean["quarter_str"] = df_clean["Quarter_End"].dt.to_period("Q").astype(str)

# Run fixed-effects regression with firm and time dummies, clustered by ticker
model = smf.ols("log_return ~ attention_lag + C(ticker) + C(quarter_str)", data=df_clean)
results = model.fit(cov_type="cluster", cov_kwds={"groups": df_clean["ticker"]})

# Extract key statistics
coef = results.params.get("attention_lag", np.nan)
stderr = results.bse.get("attention_lag", np.nan)
pval = results.pvalues.get("attention_lag", np.nan)
r2 = results.rsquared

# Format and save regression summary
summary_text = (
    f"=== Regression Summary (Lag = {lag}) ===\n"
    f"Coefficient for attention_lag : {coef:.6f}\n"
    f"Standard Error                : {stderr:.6f}\n"
    f"P-value                       : {pval:.6f}\n"
    f"R-squared                     : {r2:.4f}\n"
)
print("\n" + summary_text)
with open(f"regression_summary_lag_{lag}.txt", "w") as f:
    f.write(summary_text)

# Plot raw scatterplot with regression line (note: this does not reflect fixed effects)
plt.figure(figsize=(8, 5))
sns.regplot(
    x="attention_lag",
    y="log_return",
    data=df_clean,
    scatter_kws={"alpha": 0.4},
    line_kws={"color": "red"},
    ci=None
)
plt.xlabel(f"Lagged by {lag} Quarter Google Trends Score")
plt.ylabel("Quarterly Log Return")
plt.title(f"Raw Scatter: Attention vs Return (Lag = {lag})")
plt.grid(True)
plt.savefig(f"regression_plot_lag_{lag}.png", dpi=300)
plt.show()
```

Sector-By-Sector Regression and Plotting 2

```python
# === Sector-by-Sector Analysis of Attention and Returns ===
# This script runs a fixed-effects regression for each sector to estimate whether
# lagged Google Trends attention predicts next-quarter stock returns.
# It also outputs sector-level plots and a summary bar chart.

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
import statsmodels.formula.api as smf

# ------------------
# CONFIGURATION
# ------------------
lag = 0  # Number of quarters to lag attention scores
min_points = 30  # Minimum data points required to include a sector
dpi = 300 # Resolution for saved plots
output_dir = "sector_plots"  # Folder to save individual sector plots
# ------------------

# 1. Load pre-processed quarterly attention and price data
trends = pd.read_excel("combined_trends.xlsx")
prices = pd.read_excel("combined_stocks.xlsx")

# 2. Reshape to long format for merging
trends_long = trends.melt(id_vars="Quarter_End", var_name="ticker", value_name="attention")
prices_long = prices.melt(id_vars="Quarter_End", var_name="ticker", value_name="price")

# 3. Merge and sort chronologically
df = pd.merge(trends_long, prices_long, on=["Quarter_End", "ticker"], how="inner")
df["Quarter_End"] = pd.to_datetime(df["Quarter_End"])
df = df.sort_values(["ticker", "Quarter_End"])

# 4. Compute log returns and lagged attention values
df["log_return"] = df.groupby("ticker")["price"].transform(lambda x: np.log(x) - np.log(x.shift(1)))
df["attention_lag"] = df.groupby("ticker")["attention"].shift(lag)

# 5. Drop rows with missing values
df_clean = df.dropna(subset=["log_return", "attention_lag"]).copy()

# 6. Map company tickers to defined sectors
sector_map = {
    "WMT": "General Merchandisers",
    "COST": "General Merchandisers",
    "KR": "General Merchandisers",
    "TGT": "General Merchandisers",
    "AMZN": "Internet Retail",
    "GOOG": "Internet Retail",
    "AAPL": "Tech",
    "MSFT": "Tech",
    "UNH": "Health Insurance",
    "CI": "Health Insurance",
    "CNC": "Health Insurance",
    "ELV": "Health Insurance",
    "BRK-B": "Financial Services",
    "JPM": "Financial Services",
    "BAC": "Financial Services",
    "C": "Financial Services",
    "XOM": "Petroleum",
    "CVX": "Petroleum",
    "MPC": "Petroleum",
    "CAH": "Wholesale HC",
    "MCK": "Wholesale HC",
    "CVS": "Pharmacy",
    "F": "Auto",
    "GM": "Auto"
}
df_clean["sector"] = df_clean["ticker"].map(sector_map).fillna("Other")
df_clean["quarter_str"] = df_clean["Quarter_End"].dt.to_period("Q").astype(str)

# 7. Ensure output directory exists
os.makedirs(output_dir, exist_ok=True)

# 8. Run regressions and generate sector plots
sector_results = []

for sector in df_clean["sector"].dropna().unique():
    sector_data = df_clean[df_clean["sector"] == sector]

    if len(sector_data) < min_points:
        print(f"Skipping {sector} (only {len(sector_data)} rows)")
        continue

    unique_tickers = sector_data["ticker"].nunique()
    if unique_tickers < 2:
        print(f"Skipping {sector} (only {unique_tickers} unique ticker)")
        continue

    # Fixed-effects regression with firm and time dummies
    model = smf.ols("log_return ~ attention_lag + C(ticker) + C(quarter_str)", data=sector_data)
    results = model.fit(cov_type="cluster", cov_kwds={"groups": sector_data["ticker"]})

    coef = results.params.get("attention_lag", float("nan"))
    pval = results.pvalues.get("attention_lag", float("nan"))
    r2 = results.rsquared

    print(f"{sector:<22} | coef: {coef:+.5f} | p: {pval:.4f} | R²: {r2:.3f}")
    sector_results.append({"sector": sector, "coef": coef, "pval": pval, "r2": r2})

    # Plot scatter with OLS trendline (not fixed effects)
    plt.figure(figsize=(8, 5))
    sns.regplot(
        x="attention_lag",
        y="log_return",
        data=sector_data,
        scatter_kws={"alpha": 0.4},
        line_kws={"color": "red"},
        ci=None
    )
    plt.title(f"{sector} — Raw Scatter: Attention vs Return (Lag = {lag})")
    plt.xlabel("Lagged Google Trends Score")
    plt.ylabel("Quarterly Log Return")
    plt.grid(True)

    safe_name = sector.replace(" ", "_").replace("/", "_")
    plt.savefig(f"{output_dir}/{safe_name}_regression.png", dpi=dpi)
    plt.close()

# 9. Save summary of sector-level results
summary_df = pd.DataFrame(sector_results)
summary_df = summary_df.sort_values("pval")
summary_df.to_csv("sector_regression_summary.csv", index=False)
print("\nSaved regression summary to 'sector_regression_summary.csv'")

# 10. Generate summary bar chart of coefficients by sector
summary_df = summary_df.sort_values("coef")

plt.figure(figsize=(10, 6))
sns.barplot(
    data=summary_df,
    x="coef",
    y="sector",
    palette="coolwarm",
    orient="h"
)
plt.axvline(0, color="black", linestyle="--", linewidth=0.8)
plt.title("Effect of Lagged Google Trends on Next-Quarter Returns by Sector")
plt.xlabel("Regression Coefficient (attention_lag)")
plt.ylabel("Sector")
plt.tight_layout()
plt.savefig("sector_regression_bar_chart.png", dpi=300)
plt.show()
```

Bibliography

- Bank, M., Larch, M. and Peter, G. (2011) ‘Google Search Volume and Its Influence on Liquidity and Returns of German Stocks’, *Financial Markets and Portfolio Management*, 25(3), pp. 239–264.
- Da, Z., Engelberg, J. and Gao, P. (2011) ‘In Search of Attention’, *The Journal of Finance*, 66(5), pp. 1461–1499.
- Fortune (2024) *Fortune 500*. Available at: https://fortune.com/ranking/fortune500/ (Accessed: 1 May 2025).
- Google Trends (2025) *Google Trends*. Available at: https://trends.google.com/ (Accessed: 3 May 2025).
- Yahoo Finance (2025) *Historical stock data*. Available at: https://finance.yahoo.com/ (Accessed: 29 April 2025).