[[1]](#footnote-1)

Twitter Informed Trading on FOREX

Remy Lagrois, Daniel Lowery, Mike Sylvester

*Abstract*—Currency traders, like many others, have a presence on social media. They tweet about the trades they have made, are going to make, and what their position on different currency pairs are. We collected their tweets about the Euro/US Dollar pairing along with real time price updates from FOREX and loaded them into MongoDB. Code written in python was used to execute trades based on rules which took into account the number of tweets or the movement of the average rate. The pair was also traded using a buy and hold strategy and at random intervals. The results of these trading strategies were analyzed using SAS to determine if Twitter based trades outperformed the other methods and if not which was the optimal strategy.

*Index Terms*— Currency Trading, Dollar, Euro, FOREX, MongoDB, Python, SAS, Social Media, Twitter, USD

# INTRODUCTION

T

HIS will be the introduction

# FOREX and Twitter Data

The Foreign Exchange Market, more widely known as FOREX or FX, is the most traded currency market in the world. Trillions of dollars are traded daily globally by various organizations such as banks and individual investors. Unlike more traditional trading platforms the FOREX market is open 24 hours a day five days a week. This allows the various time zones to openly execute trades against all available currencies. The FOREX market is based on currency exchange rates and traders buy or sell selected currency pairs at a given rate known as the “Ask” price. Conversely, the currency pair can be sold at the “Bid” price. The goal is simple; buy at a lower price and sell at a higher price, or, sell at one price and buy at a lower price known as a short sell. One of the most common currency pairs is the euro against the dollar indicated by the symbol EUR/USD. Trading this pair consists of buying or selling one currency and selling the other, hence the idea of an exchange. While FOREX trading is rather simple there are many details of the process not described. This level of detail is not a fundamental aspect of this project, therefore, a general understanding of the concept is adequate.

Twitter is an avenue for millions of people to express their thoughts, feelings, or simply what is happening around them.  As such it represents a mine of information for the things happening in society and how those events affect it.  Researchers have leveraged this new resource to make quickly reacting predictions of greater social trends.  A well-publicized example is the ability to track the spread of the flu in real time and predict where an outbreak will occur.(citation) Twitter data has also been used on less concrete events as well.  The volume of Tweets about a given Presidential candidate can correlate very strongly with their polling numbers.  One group of researchers found that analyzing the actual sentiment of the Tweets was no more effective than simply looking at the volume.(citation) Given that we will only be taking volume into account.

## FOREX Data Collection

The project relies on two sources of data; FOREX and Twitter. Combined they form the basis for the project, however, they are gathered from different sources using slightly different technologies. Freely distributed application programming interfaces (API’s) are provided in both cases but the software and techniques used to capture and store the data are quite different between the two data sources.

The FOREX data API used integrates well with Microsoft.NET technologies. The particular API utilized is broken into two distinct categories; rate data interface and trading functions. The rate data interface is of primary interest in this project since it provides a direct TCP/IP socket interface to the price publication system. In other words, it provides the real-time FOREX data. The code required to connect and retrieve the data is quite simple but highly dependent on an interface key.



Fig. 1. FOREX API interface key

The example in Figure 1 is written in Visual Basic and illustrates the API interface key process. The variable “key” is session specific using a combination of user credentials. This key value is passed to the rate service as the primary method of authentication. Once connected to the rate service the live data feeds become available.

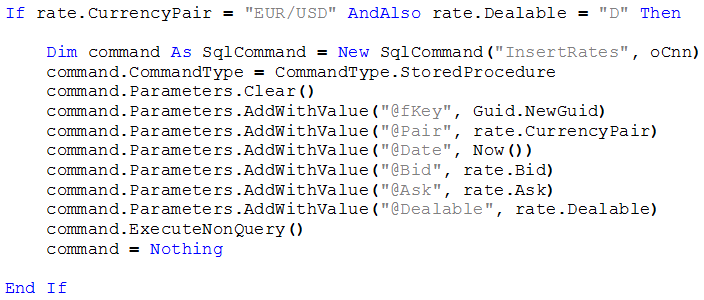


Fig. 2. Capturing FOREX data

The example in Figure 2, again written in Visual Basic, illustrates how the live data is captured and stored in a database. All available currency pairs are available, however, in the above example the EUR/USD is of interest and filtered accordingly. The “if” statement looks at only this pair and checks to ensure the pair is “Dealable”, i.e. currently tradeable. The transactional data is transmitted in simple tuple format with a single primary key value and various attributes such as the current bid and ask. The additional code presented in Figure 2 stores the tuples in a local SQL Server database using a stored procedure.

## Twitter Data Collection

There are several ways to access Twitter data and the streaming method was selected for the project. The Streaming API provides very low latency read only access to the entire global network. The API also relies on interface keys freely accessible to anyone with a Twitter account.

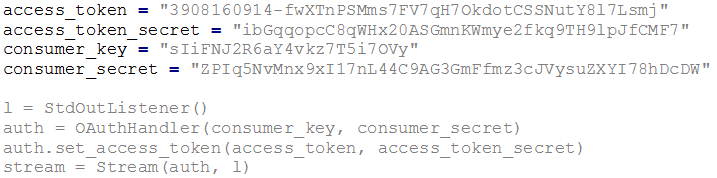


Fig. 3. Twitter API interface key

The example in Figure 3 is written in Python and illustrates the various keys required to connect to the streaming Twitter data. Once connected the data becomes available via various commands. Targeting data in an important aspect of unstructured data and the API allows for stream filtering. This significantly reduces the amount of unrelated data and allows the project to target Tweets related to “EUR/USD”. The filtered data is presented in Java Script Object Notation (JSON) format and while it is in readable format additional parsing of the data is required prior to locally storing in a database.

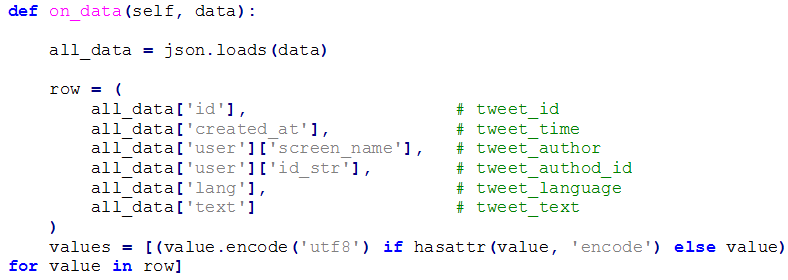


Fig. 4. Capturing Twitter Data

The example in Figure 4, again written in Python, illustrates the streaming and parsing of data. The JSON data is stored in the variable “all\_data” and various attributes/value pairs of interest are extracted such as screen name and Tweet text. The values are stored in a tuple and formatted accordingly for later storage in a local database.

# MongoDB

The data was loaded into MongoDB using the *mongoimport* tool. Data was stored in a database with two collections, a twitter collection for tweet data and a forex collection for forex data. From here the data was pulled into Python to apply the trade rules and determine a profit or loss (P/L).

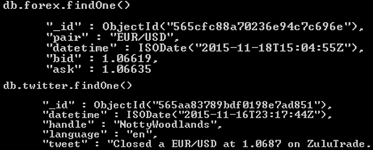


Fig. 5. FOREX and Twitter Schema

## Trading Period

The FOREX data was collected at the tick level. This means every time the price of the currency changed, the new price was stored in the database. These price changes can occur as much as multiple times a second. FOREX traders will normally select a particular time period to trade depending on their strategy. For instance, a short-term trader may decide to trade on the fifteen minute period. This kind of trader would most likely focus on the fifteen minute close of the pair they are trading. The close is the last recorded price (tick) that the currency pair traded for. In this project, it was decided to use an hourly closing price. In the FOREX markets, this trading period could be considered an intermediate-term where typically no decision to trade would be made until the end of the hour.

# Trading Rules

## Buy and Hold

The buy and hold strategy is one a lot of investors use in the stock market. It is commonly thought that if one holds on to an investment the price will eventually go up. While this may be true in many scenarios, the investment will eventually need to be closed out. For instance, if the investment was entered during a market peak and then sold twenty years later during a multi-year recession, money could actually be lost.

Given the time constraints of this project, it was impossible to hold on to a currency pair for an extremely long period. In FOREX, many traders would consider a day as a long time to hold a trade, therefore, for the buy and hold strategy, the EUR/USD was bought at the first hourly close of the day (00:59) and sold at the last hourly close (23:59). A total of four trades were made with a mean P/L of -$0.0012725.

## Moving Average

Using a moving average is a very simple and popular method used to determine whether prices appear to be going up or going down. Trade rules are usually made based on whether the current price of the currency pair is above or below the moving average. For instance, a ten period moving average would look at the average of the last ten closing prices. If the current price of the currency pair is above the moving average, it would suggest that the price is going up because it is literally above average. On the other hand, if the price is below the moving average, price would seem to be down. Once the price is considered up or down compared to the moving average, it is expected that price will continue in that direction for some period of time.

For the moving average strategy, the EUR/USD was bought if the price had risen above the ten period moving average and sold once the price dipped below the ten period moving average. A total of nine trades were made with a mean P/L of -$0.00031

## Random Buy and Sell

In many people’s minds, investing in any market, and especially the FOREX market, is the same as gambling. The thought is that no matter what strategy the investor uses, they could get the same results randomly buying and selling.

This project uses a random buy and sell strategy as a control. A random day and hour were selected for when to buy the currency pair and then a random day and hour after the buy were selected to sell the currency pair. This was done using the “randrange” function from the “random” Python module. Given a range of numbers, a random number will be generated within that range. To generate a random day, the dataset’s range of days was used. To generate a random hour, 0-23 was used. A total of ten trades were made with a mean P/L of $0.001917.

## Trading Based on Tweet Volume

The thought behind this trading strategy is that if a currency pair receives a large number of tweets then a large price change would soon follow. The EUR/USD was bought whenever it was mentioned in over 99 tweets for a given hour. If a buy position had been entered, the position would then be sold at the close of any hour where the EUR/USD was mentioned less than 100 times. A total of nine trades were made with a mean P/L of -$0.0008467.

# Statistical Analysis

## 1

## 2

## 3

## 4

## 5

# Conclusion

CONCLUSION HERE

References

*Basic format for books:*

1. J. K. Author, “Title of chapter in the book,” in *Title of His Published Book, x*th ed. City of Publisher, Country if not
2. USA: Abbrev. of Publisher, year, ch. *x*, sec. *x*, pp. *xxx–xxx.*

*Examples:*

1. G. O. Young, “Synthetic structure of industrial plastics,” in *Plastics,* 2nd ed., vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
2. W.-K. Chen, *Linear Networks and Systems.* Belmont, CA: Wadsworth, 1993, pp. 123–135.

*Basic format for periodicals:*

1. J. K. Author, “Name of paper,” *Abbrev. Title of Periodical*, vol. *x,* no. *x,* pp*. xxx-xxx,* Abbrev. Month, year.

*Examples:*

1. J. U. Duncombe, “Infrared navigation—Part I: An assessment   
   of feasibility,” *IEEE Trans. Electron Devices*, vol. ED-11, no. 1, pp. 34–39, Jan. 1959.
2. E. P. Wigner, “Theory of traveling-wave optical laser,” *Phys. Rev*.,   
   vol. 134, pp. A635–A646, Dec. 1965.
3. E. H. Miller, “A note on reflector arrays,” *IEEE Trans. Antennas Propagat*., to be published.

*Basic format for reports:*

1. J. K. Author, “Title of report,” Abbrev. Name of Co., City of Co., Abbrev. State, Rep. *xxx*, year.

*Examples:*

1. E. E. Reber, R. L. Michell, and C. J. Carter, “Oxygen absorption in the earth’s atmosphere,” Aerospace Corp., Los Angeles, CA, Tech. Rep. TR-0200 (4230-46)-3, Nov. 1988.
2. J. H. Davis and J. R. Cogdell, “Calibration program for the 16-foot antenna,” Elect. Eng. Res. Lab., Univ. Texas, Austin, Tech. Memo. NGL-006-69-3, Nov. 15, 1987.

*Basic format for handbooks:*

1. *Name of Manual/Handbook*, *x* ed., Abbrev. Name of Co., City of Co., Abbrev. State, year, pp. *xxx-xxx.*

*Examples:*

1. *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44–60.
2. *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, 1989.

*Basic format for books (when available online):*

1. Author. (year, month day). *Title.* (edition) [Type of medium]. *volume (issue).* Available: site/path/file

*Example:*

1. J. Jones. (1991, May 10). *Networks.* (2nd ed.) [Online]. Available: [http://www.atm.com](http://www.atm.com/)

*Basic format for journals (when available online):*

1. Author. (year, month). Title. *Journal.* [Type of medium]. *volume (issue),* pages. Available: site/path/file

*Example:*

1. R. J. Vidmar. (1992, Aug.). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3),* pp. 876–880. Available:<http://www.halcyon.com/pub/journals/21ps03-vidmar>

*Basic format for papers presented at conferences (when available online):*

1. Author. (year, month). Title. Presented at Conference title. [Type of Medium]. Available: site/path/file

*Example:*

1. PROCESS Corp., MA. Intranets: Internet technologies deployed behind the firewall for corporate productivity. Presented at   
   INET96 Annual Meeting. [Online]. Available: <http://home.process.com/Intranets/wp2.htp>

*Basic format for reports and handbooks (when available online):*

1. Author. (year, month). Title. Comp an y . C ity, State or Country. [Type of Medium]. Available: site/path/file

*Example:*

1. S. L. Tall een. (1996 , Apr . ). The In t r an et Archi -tecture: M a nagi ng i n f o rm at i on i n t h e ne w paradigm. Amdahl Corp., CA. [Online]. Available:<http://www.amdahl.com/doc/products/bsg/intra/infra/html>

*Basic format for computer programs and electronic documents (when available online):* ISO recommends that capitalization follow the accepted practice for the language or script in which the information is given.

*Example:*

1. A. Harriman. (1993, June). Compendium of genealogical software. *Humanist.* [Online]. Available e-mail: [HUMANIST@NYVM.ORG](mailto:HUMANIST@NYVM.ORG) Message: get GENEALOGY REPORT

*Basic format for patents (when available online):*

1. Name of the invention, by inventor’s name. (year, month day). *Patent Number* [Type of medium]. Available: site/path/file

*Example:*

1. Musical toothbrush with adjustable neck and mirror, by L.M.R. Brooks. (1992, May 19). *Patent D 326 189*

[Online]. Available: NEXIS Library: LEXPAT File: DESIGN

*Basic format for conference proceedings (published):*

1. J. K. Author, “Title of paper,” in *Abbreviated Name of Conf.*, City of Conf., Abbrev. State (if given), year, pp. *xxxxxx.*

*Example:*

1. D. B. Payne and J. R. Stern, “Wavelength-switched pas- sively coupled single-mode optical network,” in *Proc. IOOC-ECOC,* 1985,   
   pp. 585–590.

*Example for papers presented at conferences (unpublished):*

1. D. Ebehard and E. Voges, “Digital single sideband detection for interferometric sensors,” presented at the 2nd Int. Conf. Optical Fiber Sensors, Stuttgart, Germany, Jan. 2-5, 1984.

*Basic format for patents:*

1. J. K. Author, “Title of patent,” U.S. Patent *x xxx xxx*, Abbrev. Month, day, year.

*Example:*

1. G. Brandli and M. Dick, “Alternating current fed power supply,”   
   U.S. Patent 4 084 217, Nov. 4, 1978.

*Basic format**for theses (M.S.) and dissertations (Ph.D.):*

1. J. K. Author, “Title of thesis,” M.S. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.
2. J. K. Author, “Title of dissertation,” Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

*Examples:*

1. J. O. Williams, “Narrow-band analyzer,” Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, 1993.
2. N. Kawasaki, “Parametric study of thermal and chemical nonequilibrium nozzle flow,” M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

*Basic format for the most common types of unpublished references:*

1. J. K. Author, private communication, Abbrev. Month, year.
2. J. K. Author, “Title of paper,” unpublished.
3. J. K. Author, “Title of paper,” to be published.

*Examples:*

1. A. Harrison, private communication, May 1995.
2. B. Smith, “An approach to graphs of linear forms,” unpublished.
3. A. Brahms, “Representation error for real numbers in binary computer arithmetic,” IEEE Computer Group Repository, Paper R-67-85.

*Basic format for standards:*

1. *Title of Standard*, Standard number, date.

*Examples:*

1. IEEE Criteria for Class IE Electric Systems, IEEE Standard 308, 1969.
2. Letter Symbols for Quantities, ANSI Standard Y10.5-1968.

1. This paper was submitted for grading on December 10th, 2015 for the ‘File Organization and Database Management’ course from Southern Methodist University.

   The authors have received no financial support in the research, conduction, or writing of this paper.

   All authors are current students in the Masters of Data Science program of Southern Methodist University, Dallas Texas 75205. [↑](#footnote-ref-1)