Interpreting The Relationship Between Implied And Historical Volatility Through Sentiment Analysis

by

Qinmei Chen

An honors thesis submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Science

Business Honors Program

NYU Shanghai

May 2017

Professor Marti G. Subrahmanyam Professor Jiawei Zhang	Professor Stephen Figlewski
Faculty Advisers	Thesis Adviser

Acknowledgements

To Prof. Jiawei Zhang and Prof. Marti Subrahmanyam

- Thank you for providing this opportunity and resource in this honors program for every individual to pursue their own research interest

To Prof. Stephen Figlewski

- Thank you for your continued guidance throughout this year. Thank you for guiding me in the analysis processing and reminding me of every other possibilities of research direction. I really enjoyed this research throughout this year.

Abstract

The implied volatility indicates market expectation of future volatility. The difference between implied volatility and historical volatility could be interpreted as a risk premium that investors pay for when they invest in options. The risk premium could also be interpreted as market sentiment, which we extracted from Weibo posts, as an overall indication. Weibo is similar to twitter and is the third most widely used social website and the biggest open source social network. Weibo collects countless investors' opinions. These texts can extract more valuable information to forecast the gap between historical volatility and implied volatility when sentiment text analysis technics applied. If the positivism of Weibo texts indicates investors are optimistic about the future market, the higher the investors' optimistic, the lower the gap between implied volatility and historical volatility. Oppositely, if investors are pessimistic about the future market, the gap between implied volatility and historical volatility goes higher. In this research project, we collect the dataset by a crawler software (data pre-processing); we perform machine learning techniques – sentiment text analysis – to extract the sentiment features from texts; we conduct Granger Causality Analysis—to predict the gap between historical volatility and historical volatility. Our project aims to develop sentiment analysis tools and correlate contributed content to predict the gap between historical volatility and implied volatility. Our study demonstrated some objective indications that combining massive new data sources from Weibo posts offer a better understanding on the behavior of the gap between implied volatility and historical volatility in Chinese financial market.

Table of Contents

Introduction

With the prevalence of social networks, investors posted thousands of comments of the stock market on their social networks. For Chinese Investors, they tend to use Weibo, Wechat and other Chatting Rooms. During the booming period, most investors act like experts and posted various seemingly professional advices with positive recommendations. Yet when the crash of stock market happened, investors expressed their pessimistic expectation for the stock market accompanied with their desperate massive dumping of stocks. It seems that investors' sentiments exert a substantial influence on the stock market, which corresponds to behavioral economics theory.

Behavioral economics theory indicates that emotions can profoundly affect individual behavior and decision-making. In the Early 2010, a group of computer scientists discover an astonishing relationship between public emotion and the value of the Dow Jones Industrial Average (DJIA).[1] They analyze the text content of daily Twitter feeds by mood tracking tools that measures mood in several dimensions, including Calm, Alert, Sure, Vital, Kind, and Happy. Through Granger causality analysis they conclude that public mood states, are predictive of changes in DJIA closing values.

Inspired by the astonishing finding, scientists from UC Santa Barbara and Tsinghua University conducted sentiment analysis on posts and comments posted by various authors on the Collaborative Investing Platforms. They extracted sentiments from posts and the corresponding comments then ranked author based on their prediction accuracy. [2]

Although the above researches seem promising in developing a sophisticated trading strategy for making profits, they did not mention other drivers for stock prices. One of the factors

---

[1] Bollen, J., Mao, H. and Zeng, X.-J. 2010. Twitter mood predicts the stock market.

[2] Wang, Gang, Tianyi Wang, Bolun Wang, Divya Sambasivan, Zengbin Zhang, Haitao Zheng, and Ben Y. Zhao. "Crowds on Wall Street."

they ignore is the company performance, which is a fundamental driver for the stock price. One common situation is that when the quarterly or annual reports issues, if the company's performance exceeds analysts' estimates, the stock price goes up; otherwise, the stock price goes down. Such situation did not explained by the above research papers. Hence, to develop a trading strategy purely on sentiment analysis, stock may not be proper product.

Then the question becomes is there a financial instrument who only drives by investors' sentiments? The answer is Yes. Still in 2015, beyond what happens in drastically oscillating stock market, Chinese stock exchange also issued its first option instrument, the 50 ETF options. In financial Mathematics, "the implied volatility of an option contract is that value of the volatility of the underlying instrument which, when input in an option pricing model (such as Black–Scholes) will return a theoretical value equal to the current market price of the option.[3] In other words, the implied volatility implies market expectation of the stock's volatility and its purely depends on investors' expectation of the future volatility of the Index.

Noticeably, no matter index rises or falls, there is always a gap between the implied volatility and historical volatility. Although the historical volatility may influence investors' judgment of future volatility, investors' expectation of future volatility (Implied Volatility) are still distinguished from the historical volatility. They may expect a different trend from the present market, or a big political or social economical event. All of those uncertainties affect their expectation of future volatility, which generates the gap between implied and historical volatility. At the time when Index experienced a significant drop, historical volatility reached one of historical climaxes. Although the implied volatility at the same period demonstrate a similar pattern as historical volatility, there exists a significant gap between those two, which is

[3] Mayhew, Stewart. "Implied Volatility."

interpreted by most economists as the discrepancy between the reality and investor's expectation for the future.

The discrepancy between implied volatility and historical volatility serves as a perfect research entity to conduct sentiment analysis. When the gap (defined as implied volatility – historical volatility) between the implied and historical volatility are small, the investors are highly possible optimistic about the future market. Oppositely, if investors are pessimistic about the future market, the gap between implied volatility and historical volatility goes higher. In this thesis, I crawled the Weibo Posts regarding the stock market from a Weibo Crawler, quantified the sentiments in the Weibo posts through Natural Language Processing Sentiment Analysis Technique, and analysis the causality relationship between the Weibo Sentiments and the volatility gap between the historical volatility and implied volatility. The following sections demonstrate the primary results of this research and provide an objective view on the Investors sentiments in the prediction of gap between historical and implied volatility. Our Research finds that the when market goes up, positive sentiment are statistically significant and the more positivity, the less volatility gap. Such finding matches our research hypothesis.

Data

Weibo Posts – Obtained through a Web Crawler Software

What is a Web Crawler?

Web crawler is an application, which would automatically browses the websites according to the keywords that users specifies, and then downloaded all the search results with the date-stamp, author information and posts contents.

Why posts from Weibo instead of posts from other social networks?

Among all the social networks, including Weibo, Wechat and other online chatting rooms and forums, we choose Weibo posts to conducts the sentiment analysis, for the following reasons: First, compared to Wechat and QQ, all the posts on Weibo is visible to all the web users. Although Wechat and QQ are the most-used Social Media in Chinese society, the posts of someone can only been seen within its friend circle. And the Tencent Company, who invented Wechat and QQ, strictly prohibits the disclosure of users' posts. So Weibo, the third most popular social network in China, provides the most information we can crawl from crawler software. Second, although there exists several stock discussion forums, none of them has much more users than Weibo and the discussion posts on these forums mix up with irrelevant information so we cannot use a Web Crawler to automatically detect those relevant posts with certain keywords in a time manner. Despite Weibo is not a financial-specific posts aggregation, it facilitates us to obtain all the relevant posts with key words related to stocks with corresponding user and date information. Therefore, Weibo is the ideal platform to crawl the posts from.

The basic information of the Weibo Posts:

All the Weibo posts obtained through Web Crawler, with the search keywords "Stock", "Stock Market" and "Options", dating from Jan 20 Till Sep 20 2016. After cleaning the redundant

Weibo posts, the total number of Posts is 50378. So on average, roughly 300 posts concerning the key words “Stock”, “Stock Market”. Among those days, holidays and weekends have less posts. So on trading days, there are much more than 300 posts per day.

Implied Volatility of ETF 50 Options – Obtained from Wind Terminal

What is ETF 50 Options?

ETF 50 options are the only traded options in the Chinese financial market. The ETF 50 index is a blue-chip index in the Shanghai Composite market, which consists of less than 50 stocks. The ETF 50 index reflects overall market but concentrated on large strong firms; implied volatility from ETF 50 Index may not perfectly reflect sentiment about smaller and weaker firms.

Implied volatility of ETF 50 Options – Proxy for Chinese VIX

In global market, VIX type index indicates market sentiment of volatility. However, no VIX-type index exists in the Chinese financial market. The implied volatility of ETF 50 index serves as a proxy for the market sentiment in Chinese Financial market.

Wind Terminal

Like Bloomberg Terminal, Wind Terminal is a professional financial database, which focus on the Chinese financial market.

The Wind terminal provides Implied volatility data of ETF 50 Options.

How the Wind calculates the implied volatility of the options?

First the Wind sets the boundary of implied volatility. Then applies Blacks-Scholes Model to calculate the theoretical price of the options under such boundary and compares such theoretical price with the real option price. With continuous applying the Bisection Methods to limit the implied volatility boundary, the implied volatility with certain accuracy can be determined and provided to its users.

Historical Volatility of ETF 50 Index– Obtained from Wind Terminal

The historical volatility is 90 days annualized historical volatility of ETF 50 Index, also obtained from the Wind Terminal. I have verified the data accuracy by calculating the 90-days annualized historical volatility with the ETF 50 Index data downloaded from Yahoo Finance with the same time period.

Volatility Gap of an option:

Volatility gap = Implied_Volatility of the option – 90_days_ETF50_Index_Historical Volatility

Daily Return of ETF 50 Index

The ETF 50 Index data is downloaded from Wind Terminal.

Daily Return = ETF_50_Index (T) – ETF_50_Index(T-1)

Data Processing – Sentiment Analysis

What is Sentiment Analysis?

> "Sentiment analysis (also called opinion mining) refers to the application of natural language processing, computational linguistics, and text analytics to identify and classify subjective opinions in source materials (e.g., a document or a sentence). **Generally speaking, sentiment analysis aims to determine the attitude of a writer with respect to some topic or the overall contextual polarity of a document.** The attitude may be his or her judgment or evaluation, affective state (that is to say, the emotional state of the author when writing), or the intended emotional communication (that is to say, the emotional effect the author wishes to have on the reader)." [4]

Discussion of Sentiment Analysis Techniques

Since the Weibo posts are all in Chinese, the most intuitive way to conduct sentiment analysis is to utilize a Sentiment Analysis technique developed in a Chinese Natural Language Institution. However, the DURF Research I did in Summer 2015 proved that conducting sentiment analysis in Chinese is not a valid option. During that research, I conducted sentiment analysis on comments of a stock China Petroleum through a sentiment dictionary downloaded from an open-source research database. This sentiment dictionary is not an authoritative sentiment dictionary. No authority implies the sentiment dictionary guarantees no completeness of the sentiment words. Financial Sentiments, like "go up", " go down", "bear market", "booming market" are not included in this sentiment dictionary. Another issue with this sentiment dictionary is that such sentiment dictionary can only support conduct sentiment analysis word by word. The result of a word-by-word sentiment analysis cannot accurately represent paragraph sentiment.

---

[4] Luo, Tiejian, et al. Sentiment Analysis. Trust-based Collective View Prediction.

The alternative is to conduct sentiment analysis in English. My research found a Python Natural Language Processing tool named Sentiment Analysis Vader facilitates the sentiment analysis of a whole passage in English, which is developed by a group of Stanford Scientists, and recognized as one of the most commonly used and authoritative sentiment analysis package. Like the Chinese sentiment dictionary, this sentiment analysis technique is neither a finance-specific sentiment analysis. So this sentiment dictionary still cannot detect financial terminologies. Yet there exist no finance-specific sentiment analysis, so the Natural Language Processing sentiment analysis technique is the most advanced sentiment analysis technique we can adopt.

However, before conducting sentiment analysis, we need to first translate the Weibo posts from Chinese into English. Given the gigantic amount of Weibo Posts, manual translation is not time-efficient. Hence automatic machine translation is the only choice. However, no matter how accurate the machine translation performs, the translation process would unavoidably lost some meaning of the original posts. So we are facing the trade-off between losing information within the translation process or losing information when conducting word-by-word sentiment analysis.

From my understanding, the sacrifice in translation process is negligible comparing with information lost within the word-by-word sentiment analysis process. The machine translator functions as a Chinese person who proficient English. Such Chinese person possible understand another the English written by another Chinese person who is also proficient at English. But if we separate a Chinese passage word by word and arrange those words randomly, like the word-by-word sentiment analysis did, another Chinese person could barely understand those scrambled Chinese words.

Based on the above pros and cons comparison of existing sentiment analysis technique, in this research, we apply the sentiment analysis within the Python Natural Language Processing Package.

Comparison of the machine translation: Google VS Baidu

Google is the most used search engine globally and Baidu dominates the Chinese search engine market in China. They both provide the machine translation service in multi-languages. We need to choose one machine translation mechanism provides the most accurate translation and lost the least information. Let us examine a small example of those two translation results to compare the translation accuracy.

The original Chinese Weibo Posts:

> “大盘涨，小盘跟不上，大盘跌，小盘溜的飞快，散户亏本百分之七八十，国家允许上市这么多股票，股民的钱却在急速的减少，这该如何解释？？富人买飞机，我们吃饭都得计划！如今食品不安全，网络不安全，骗局陷阱无处不在，一切的罪魁祸首就是钱钱钱，没有稳定的生活，社会怎么安定！ ”

The Google Translator Result:

> “The broader market up, small cap can not keep up, the broader market fell, small disk slip fast, the retail loss of seventy percent, the state allowed to list so many stocks, investors are in the rapid reduction in the money, how to explain this? The rich buy the plane, we have to plan to eat! Now food is not safe, the network is not safe, scam trap everywhere, all the culprit is money, no stable life, how social stability! ”

The Baidu Translator Result:

> "The market rose, the small cap can not keep up, the market fell, the small slip of the fast, retail loss of seventy or eighty per cent, the state allows the listing of so many stocks,

investors in the rapid reduction of money, how to explain this? Rich people buy planes, we have plans to eat! Today, food safety, network security, fraud traps everywhere, all the arch-criminal is money, without a stable life, how social stability!"

The translation difference is highlighted with red. By comparison of the translation, Google translation is easier to understand in English and closer to the origin Chinese meaning. Since Google translator outperforms Baidu Translator, in this research, we adopted the Google Translator to conduct all the translation work.

How the Machine Translation Works – Python Programming

Basically what this program does is to imitate a Google translator user who inputs the Chinese posts which needed to be translated into the input box on the left and then click the translation bottom below. After the Google translator finished up its translation, this program would copy and store the translated result into a database. In this way, the translation process of a single post is finished. With repeating such process for all the posts, we successfully managed to translate all the Weibo posts from Chinese to English. Refer to the code in Appendix 1.

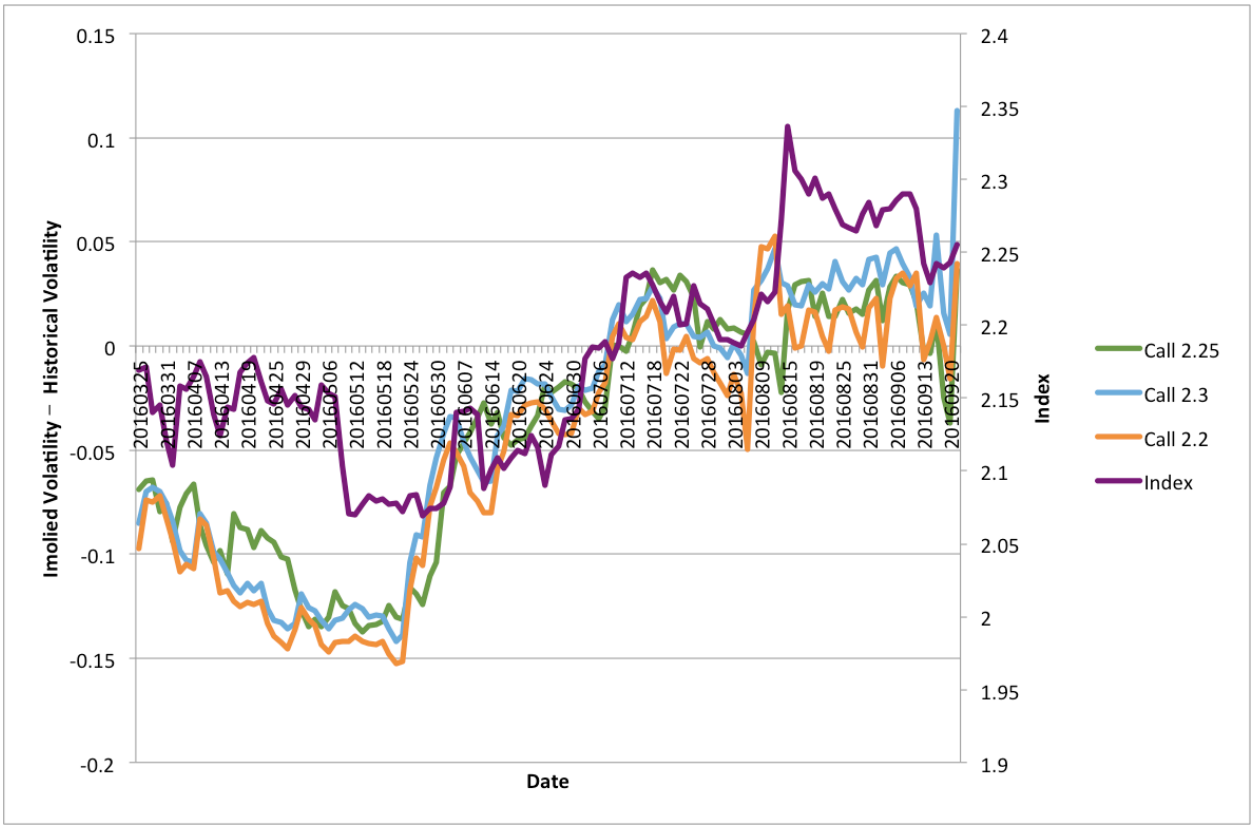
Conduct Sentiment Analysis through Sentiment Analysis Vader – Python Programming with Natural Language Processing Package

Corresponding to the daily implied volatility and historical volatility data, I aggregate the translated posts within the same day as one input file. After inputting the daily aggregated posts into the sentiment analysis Vader, the program automatically responds the sentiment score of the daily posts with three attributes: Positive_Score, Negative_Score, and Neutral Score.

Refer to the code in Appendix 2.

Data Analysis

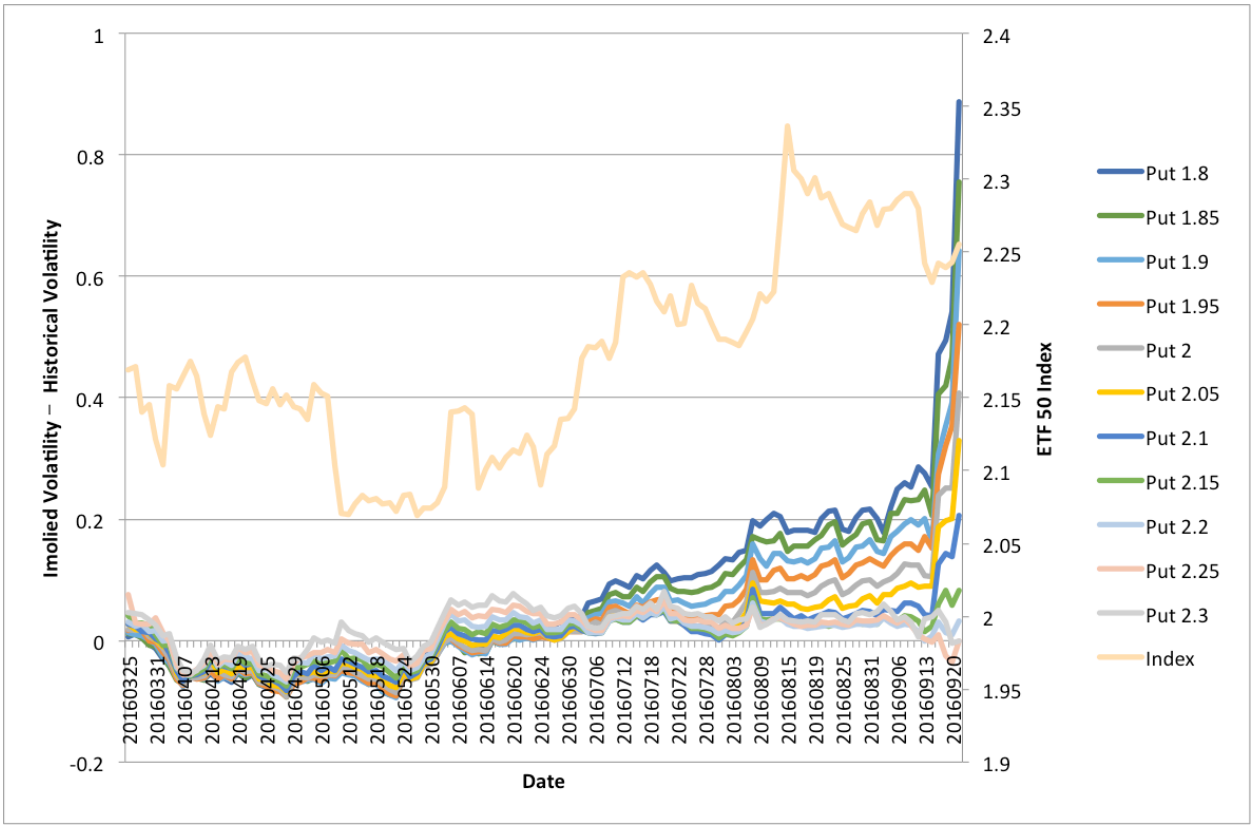
Volatility Gaps of Call Options matured at Sept. 2016



Note: label are formatted as Call + Strike Price

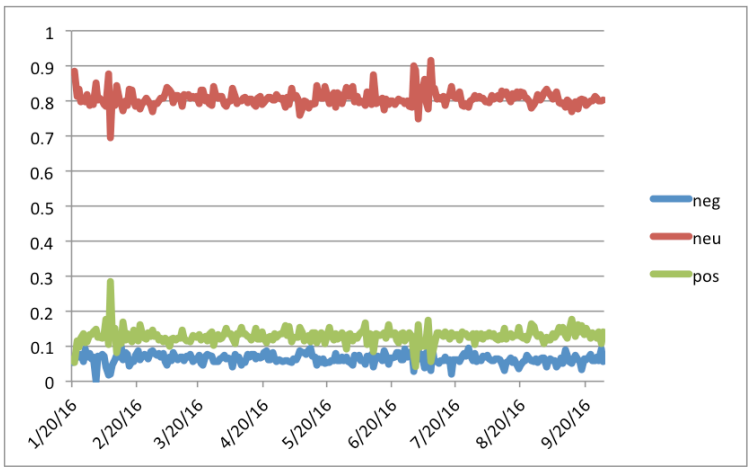
We observe the volatility smile pattern for both out of the money call options. Before July, three options are all out of the money. After July, high strike call “call 2.3” is still out of the money. From March to May, the Index is less volatile. So the volatility gap (implied volatility – historical volatility) moves negative. However, in May, the stock market behaves worse than previous month, so the volatility gap moves towards positive. From July to Sept, the index volatilely goes up. Given the high volatility of index, the volatility gap also went up. In this period, the higher the strike price, the higher the volatility gap (implied volatility – historical volatility). Since they all share the same historical volatility, the higher the strike price, the higher implied volatility.

Volatility Gaps of Put Options matured at Sept. 2016



We also observe volatility smile for low strike puts which are out of the money. The low strike puts are sensitive in their volatility gap (implied volatility – historical volatility) when the index behaves volatile. From March to June, the index behaved less volatile compared to July to September. So from March to June, all puts behaves similar. However, from July to Sept in 2016, a big difference of volatility gap appeared according to the puts various strike prices: the higher the put option strike price, the lower the volatility gap. Because the historical volatility is the same for all put options, the higher strike price indicated the lower implied volatility. This pattern is typical for implied volatilities—out of the money puts, with low strike prices, have higher implied volatilities than options with higher strikes--and it is found all over the world.

Weibo Sentiment Score



Pos: Positive sentiment score of Weibo post, range [0,1], the more optimistic of Weibo posts, the higher the positive sentiment score.

Neg: Negative sentiment score of Weibo post, range [0,1], the more pessimistic of Weibo posts, the higher the negative sentiment score.

Neu: Neutral sentiment score of Weibo post, range [0, 1], refers to the measurement of objectivity of Weibo posts. The more objective of Weibo posts, the higher the neutral sentiment score.

Most posts are neutral, probably because the sentiment analysis does include financial-specific sentiments.

The positive sentiment score is always greater then the negative sentiment score. The results match the industry practice where analyst always put more buy recommendation rather than sell recommendations.

The great fluctuation in the late June may caused by the UK voted to leave the EU.

Causality Regression Analysis and Results

Individual Options Regression Definition

We apply the granger causality test to test whether sentiments have improved the prediction of the gap between the implied and historical volatility. For each of the option, the causality regression tests are set in the following:

Dependent Variable: gap = implied volatility – historical volatility

Regressions:

1. Dependent Variables: **gaplag_1**: yesterdays' gap
2. Independent Variable: **gaplag_1**: yesterdays' gap;

   gaplag_2: the day before yesterday's gap

   neg: yesterday's negative sentiment

   pos: yesterday's positive sentiment

   neglag_1: the day before yesterday's negative sentiment

   poslag_1: the day before yesterday's positive sentiment

   neglag_2: two days before yesterday's negative sentiment

   poslag_2: two days before yesterday's positive sentiment

   Index: Index return of ETF 50 Index

We separate the data to two parts according to whether the ETF 50 Index goes up or goes down. For each of the option, we collected the regression coefficient and its corresponding p-value and the adjusted-R Square for analysis.

Individual Options Regression Results

Table 1 shows the regression result of put option 10000570 with the strike price 1.8, as an example of all the option regression result. Let's take a close look at the volatility gap lag

coefficients and sentiment score coefficients. We can see that the gap_lag1 are statistically significant at 5% level. Gap_lag2 and pos_lag1 are statistically significant at 10% level. This result indicates besides yesterday's and the day before yesterday's volatility gap, the yesterday's positive sentiment score also has some predictive value in forecasting today's volatility gap. However, no negative sentiment appears to be statistically significant in the prediction of volatility gap.

Through analysis all the option regression results, we found that the index variable seemed to be almost never significant. So we dropped those index variable and aggregate the regression outputs into the following Table 2-4:

- Table 2 aggregates the regression results for all the options with different strike prices in the whole period, regardless of market performance. Taking a close look at the gap lags and pos, neg coefficients, we found that for low strike puts, which are out of the money, most gap_lag1 and gap_lag2 are statistically significant. The gap lag coefficients indicate yesterday's gap tends to increase today (gaplag_1 is positive) but to decrease for higher strike puts and for calls. This may be even stronger for the second lag. Also, at least one coefficient from wither pos-lag1 or neg_lag2 are statistically significant.
- Table 3 demonstrates the regression results when market goes up. Still looking at gap_lags coefficients and pos, neg coefficients. Compared to gap_lags coefficients in table 2, for the OTM puts, when index goes up, only the lowest two strike puts are statistically significant. For positive and negative sentiment coefficients, the lowest strike put has a neg coefficient statistically significant and two high strike puts has pos coefficients at 10% significant level.

- Table 4 demonstrates the regression result when market goes down. Still looking at gap_lags coefficients and pos, neg coefficients. For the OTM puts, the coefficients of gap_lag2 and pos_lag 1 are statistically significant at 5% level. Most pos_lag2 coefficients are also statistically significant, mix up with 10% level significance and 5% level significance. Some neg_lag1 coefficients are also statistically significant. The strongly positive gap_lag coefficients indicate the gaps are getting bigger as Implied volatilities tend to go up in down market. This suggests that having had positive sentiment in recent days increase this effect.
- Note: Given the limited number of observations, we cannot expect statistically significance at 1% or 5% level, so we set the significance level at 10%. All the results described above all based on 10% significance level, except from those with specifications.

Individual Options Regression Conclusions:

- There is an interesting result since the connection between index returns and implied volatility is so strong in China, especially in a down market
- Coefficient on negative and positive sentiments are nowhere near significant overall, but negative coefficients are mostly negative on up days and positive on down days, while it is the reverse for positive coefficients. This may possibly indicate the volatility smile getting flatter on up days and higher on down days.
- One of the strongest effects seems to be from lagged positive sentiment when market is down. Implied volatilities tend to go up in down market (the gaps are getting bigger as a consequence, with coefficients on lagged gaps strongly positive). This suggests that having had positive sentiment in recent days increase this effect. For lagged negative

sentiment on up days, we have the same negative coefficients, but they are not as large and not significant.

Regressions on Average Put Volatility Gap

Table 5-7 demonstrated the regression results after we take the average the put volatility gaps.

- Table 5 shows the overall results, regardless of market goes up or down. Looking at the coefficients, the pos coefficients are statistically significant. The strongly negative pos coefficients suggest that the more positive investors feels, the less volatile they expect market to be, so that the volatility gap goes down and implied volatility goes down. This results matches our hypothesis.
- Table 6 shows the regression results of Average Put Volatility Gap when market goes up. Similar to table5, the pos coefficients are statistically significant. The pos coefficients are also strongly negative. This results matches the results in Table 5. And in the up market, we can see an improvement in the adjusted- R square. The pos statistical significance and improvement of adjusted R square strengthened the results that the more positive investors feels, the less volatile they expect market to be, so that the volatility gap goes down and implied volatility goes down.
- Table 7 shows the regression results of Average Put Volatility Gap when market goes down. None of the coefficients are statistically significant when market goes down.

Further Improvement

Improvement of the Sentiment Analysis Tools

Throughout the sentiment dictionary, some financial related sentiments are not included, like, bear, bull and other financial terminologies. The Lack of financial sentiment would possibly result an inaccurate measurement of Weibo Sentiments.

Another improvement for the sentiment dictionary would be to fractionize the positive and negative sentiments into something like super-positive, positive, and less-positive and super-negative, negative and less-negative. Then for each sub category, calculate its sentiment score.

Avoid Information Lost in the translation process

The information loss is unavoidable in any of the translation process. The perfect solution would be conducting sentiment analysis with an authoritative Chinese sentiment analysis tool which includes the financial terminologies.

Filter out Authoritative Weibo Posts

The Weibo posts aggregate comments from every possible individual. But not all the posts are as financially authoritative as those finance VIPs. If we could filter out Weibo posts from those financial VIPs and conduct sentiment analysis only on those posts. The sentiment analysis results may be more accurate and less misleading.

Conclusion

The research of the thesis only scratches the surface of the power of sentiments in the interpretation of the gap between historical and implied volatility. While the causality analysis result– when market goes up, positive sentiment are statistically significant and the more positivity, the less volatility gap –matches our hypothesis, further work on the improvement of sentiment analysis must be done, in order to strengthen this research implication. Research and data on weibo comments is rather incomplete: Not all the Weibo comments are authoritative in financial sense and the existing research is not able to differentiate the authoritative posts from the not authoritative ones. The sentiments analysis would become promising financial analysis techniques with the development of natural language processing techniques: while for the moment we do not have a perfect sentiment analysis tool to incorporate all the fluctuations of the volatility gap, this primary research sheds lights on the effectiveness of such technique in the prediction improvement of the volatility gap.

Table 1 -- Regression result of put option with the strike price 1.8.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
gaplag_1	0.234**	0.200**	0.204**	0.209**	0.199**	0.207**	0.210**	0.220**	0.210**
	(0.011)	(0.030)	(0.028)	(0.023)	(0.034)	(0.027)	(0.025)	(0.019)	(0.028)
gaplag_2		0.166*	0.166*	0.166*	0.170*	0.166*	0.166*	0.166*	0.170*
		(0.073)	(0.075)	(0.073)	(0.070)	(0.073)	(0.075)	(0.073)	(0.070)
neg			-0.0638	-0.0947	-0.101		-0.0625	-0.0935	-0.0990
			(0.431)	(0.298)	(0.287)		(0.443)	(0.305)	(0.299)
pos			0.0300	-0.0450	-0.0568		0.0296	-0.0499	-0.0606
			(0.607)	(0.523)	(0.447)		(0.613)	(0.481)	(0.420)
neglag_1				-0.0319	-0.0533			-0.0315	-0.0502
				(0.723)	(0.615)			(0.726)	(0.636)
poslag_1				-0.132*	-0.167*			-0.140*	-0.171*
				(0.063)	(0.064)			(0.052)	(0.058)
neglag_2					-0.0261				-0.0221
					(0.786)				(0.819)
poslag_2					-0.0480				-0.0441
					(0.525)				(0.561)
index						-0.0461	-0.0434	-0.0658	-0.0614
						(0.581)	(0.605)	(0.434)	(0.470)
_cons	0.00211	0.00184	0.00182	0.00184	0.00187	0.00183	0.00181	0.00182	0.00185
	(0.129)	(0.184)	(0.190)	(0.184)	(0.179)	(0.189)	(0.195)	(0.190)	(0.186)
N	127	127	127	127	127	127	127	127	127
adj. R-sq	0.044	0.061	0.052	0.064	0.052	0.055	0.047	0.061	0.048

* p <0.1, ** p < 0.05, *** p<0.01, **** p <0.001

Table 2 -- Regression results for individual options

All	10000570	10000560	10000561	10000562	10000563	10000564	10000572	10000574	10000596	10000600	10000591	10000595	10000599
	1.8	1.85	1.9	1.95	2	2.05	2.1	2.15	2.25	2.3	2.2	2.25	2.3
	Put	Put	Put	Put	Put	Put	Put	Put	Put	Put	Call	Call	Call
gaplag_1	0.199**	0.217**	0.0925	0.106	0.225**	0.0626	0.0137	-0.160*	-0.196**	-0.104	-0.494****	-0.155*	-0.111
	(0.034)	(0.015)	(0.313)	(0.240)	(0.017)	(0.492)	(0.883)	(0.094)	(0.039)	(0.284)	(0.000)	(0.095)	(0.232)
gaplag_2	0.170*	0.199**	0.184*	0.270***	0.0718	0.153*	0.0772	0.0383	0.0383	0.0185	-0.155	-0.0291	-0.0236
	(0.070)	(0.021)	(0.055)	(0.005)	(0.442)	(0.097)	(0.414)	(0.690)	(0.689)	(0.845)	(0.233)	(0.761)	(0.811)
neg	-0.101	-0.0651	0.0377	-0.0103	-0.0252	0.0197	0.0521	0.0769	0.0256	0.0707	-0.0602	0.0735	-0.0197
	(0.287)	(0.439)	(0.664)	(0.895)	(0.721)	(0.777)	(0.461)	(0.270)	(0.748)	(0.394)	(0.607)	(0.339)	(0.746)
pos	-0.0568	0.0210	-0.0538	-0.0369	-0.0502	-0.0104	-0.0227	-0.0258	0.0288	0.0387	0.00223	0.158***	-0.00210
	(0.447)	(0.752)	(0.430)	(0.547)	(0.362)	(0.847)	(0.681)	(0.634)	(0.641)	(0.545)	(0.981)	(0.004)	(0.965)
neglag_1	-0.0533	-0.0646	-0.00777	-0.102	-0.0309	0.00610	0.0149	0.0679	0.0491	0.0851	0.0746	-0.0790	-0.0530
	(0.615)	(0.494)	(0.936)	(0.240)	(0.688)	(0.936)	(0.848)	(0.378)	(0.578)	(0.358)	(0.567)	(0.348)	(0.433)
poslag_1	-0.167*	-0.121	-0.139*	-0.110	-0.131**	-0.0560	-0.0898	-0.0313	-0.00406	0.0437	0.0196	0.0900	0.0271
	(0.064)	(0.133)	(0.090)	(0.134)	(0.046)	(0.383)	(0.169)	(0.626)	(0.956)	(0.567)	(0.858)	(0.143)	(0.635)
neglag_2	-0.0261	-0.126	-0.105	-0.162**	-0.0737	-0.131*	-0.0708	-0.0424	-0.0993	-0.117	0.0890	-0.0601	0.000995
	(0.786)	(0.135)	(0.233)	(0.041)	(0.293)	(0.061)	(0.315)	(0.546)	(0.220)	(0.170)	(0.451)	(0.447)	(0.987)
poslag_2	-0.0480	-0.0544	-0.0934	-0.0540	-0.0318	-0.00998	-0.0411	0.000243	-0.00278	0.0547	0.0944	0.0381	0.0354
	(0.525)	(0.418)	(0.173)	(0.380)	(0.568)	(0.853)	(0.454)	(0.996)	(0.964)	(0.392)	(0.305)	(0.489)	(0.459)
_cons	0.00187	0.000741	0.00124	0.000872	0.000162	-0.000189	-0.000684	-0.00164	-0.00269**	-0.00233*	-0.00234	0.00000781	-0.00114
	(0.179)	(0.550)	(0.322)	(0.438)	(0.870)	(0.847)	(0.494)	(0.104)	(0.022)	(0.053)	(0.169)	(0.995)	(0.200)
N	127	123	127	127	127	127	127	127	127	127	127	130	127
adj. R-sq	0.052	0.094	0.021	0.065	0.048	0.014	-0.022	-0.010	0.018	0.004	0.152	0.052	-0.039

* p <0.1, ** p < 0.05, *** p<0.01, **** p <0.001

Table 3 -- When market goes up, regression results for individual options

UP	10000570	10000560	10000561	10000562	10000563	10000564	10000572	10000574	10000596	10000600	10000591	10000595	10000599
	1.8	1.85	1.9	1.95	2	2.05	2.1	2.15	2.25	2.3	2.2	2.25	2.3
	Put	Put	Put	Put	Put	Put	Put	Put	Put	Put	Call	Call	Call
gaplag_1	0.299***	0.235**	0.067	0.127	0.215	0.024	-0.0542	-0.176	-0.28	-0.146	-0.316	-0.391**	-0.0921
	(0.007)	(0.039)	(0.610)	(0.350)	(0.151)	(0.856)	(0.724)	(0.241)	(0.117)	(0.303)	(0.148)	(0.012)	(0.335)
gaplag_2	0.327*	0.112	0.0945	0.156	0.155	0.168	0.143	0.0372	-0.0618	-0.142	-0.373	0.203	-0.008
	(0.073)	(0.391)	(0.511)	(0.328)	(0.365)	(0.334)	(0.442)	(0.828)	(0.712)	(0.306)	(0.174)	(0.327)	(0.937)
neg	-0.374***	-0.112	-0.091	-0.124	-0.15	-0.106	-0.053	-0.0245	-0.0581	-0.00139	-0.0375	0.111	-0.0178
	(0.009)	(0.373)	(0.490)	(0.307)	(0.178)	(0.343)	(0.656)	(0.828)	(0.656)	(0.991)	(0.867)	(0.373)	(0.770)
pos	0.0683	0.0453	0.0123	0.000308	0.0659	0.109	0.0778	0.101	0.193*	0.169*	0.0111	0.291****	-0.00491
	(0.521)	(0.634)	(0.903)	(0.997)	(0.449)	(0.213)	(0.409)	(0.263)	(0.065)	(0.071)	(0.949)	0.000	(0.919)
neglag_1	-0.244	-0.0479	-0.172	-0.226	-0.142	-0.0639	-0.0923	-0.0147	0.00467	0.103	0.0692	-0.105	-0.0513
	(0.164)	(0.760)	(0.298)	(0.142)	(0.318)	(0.651)	(0.544)	(0.919)	(0.978)	(0.494)	(0.805)	(0.438)	(0.449)
poslag_1	0.000911	-0.0242	-0.0396	-0.0463	-0.0225	0.0785	0.0116	0.0983	0.186	0.179	0.1	0.239**	0.0239
	(0.994)	(0.828)	(0.739)	(0.674)	(0.824)	(0.439)	(0.915)	(0.345)	(0.119)	(0.100)	(0.616)	(0.016)	(0.677)
neglag_2	-0.189	-0.134	-0.154	-0.183	-0.149	-0.19*	-0.15	-0.139	-0.186	-0.168	0.125	-0.225	0.00388
	(0.148)	(0.252)	(0.214)	(0.114)	(0.167)	(0.079)	(0.194)	(0.206)	(0.145)	(0.148)	(0.560)	(0.138)	(0.950)
poslag_2	0.0153	0.00274	-0.0753	-0.0462	-0.0176	0.0187	-0.0273	0.0377	0.0901	0.138*	0.0702	0.0873	0.0381
	(0.866)	(0.973)	(0.380)	(0.566)	(0.817)	(0.799)	(0.736)	(0.618)	(0.301)	(0.083)	(0.630)	(0.403)	(0.428)
_cons	0.00176	0.00189	0.00142	0.000132	-0.000325	0.000501	0.0000705	-0.00085	-0.00175	-0.00338*	-0.00308	-0.00152	-0.00111
	(0.388)	(0.304)	(0.466)	(0.942)	(0.851)	(0.763)	(0.969)	(0.621)	(0.371)	(0.062)	(0.334)	(0.431)	(0.212)
N	60	60	60	60	60	60	60	60	60	60	60	64	60
adj. R-sq	0.172	0.027	-0.074	-0.029	0.029	-0.029	-0.087	-0.061	0.03	0.049	-0.042	0.19	-0.008

* p <0.1, ** p < 0.05, *** p<0.01, **** p <0.001

Table 4 -- When market goes down, regression results for individual options

Down	10000570	10000560	10000561	10000562	10000563	10000564	10000572	10000574	10000596	10000600	10000591	10000595	10000599
	1.8	1.85	1.9	1.95	2	2.05	2.1	2.15	2.25	2.3	2.2	2.25	2.3
	Put	Put	Put	Put	Put	Put	Put	Put	Put	Put	Call	Call	Call
gaplag_1	0.0188	0.0816	0.299**	0.154	0.128	0.0579	-0.0488	-0.327**	-0.103	0.0771	-0.563****	0.0538	0.22*
	(0.908)	(0.648)	(0.032)	(0.182)	(0.317)	(0.646)	(0.685)	(0.020)	(0.312)	(0.617)	0.000	(0.662)	(0.097)
gaplag_2	0.276***	0.378***	0.269**	0.294***	0.162	0.169*	0.11	0.0812	0.144	0.124	0.0295	-0.0963	0.112
	(0.004)	(0.005)	(0.042)	(0.010)	(0.110)	(0.061)	(0.215)	(0.453)	(0.168)	(0.408)	(0.828)	(0.369)	(0.333)
neg	0.193*	0.094	0.186	0.158	0.151*	0.172**	0.178**	0.235***	0.107	0.149	0.0669	-0.0088	0.0544
	(0.096)	(0.500)	(0.145)	(0.135)	(0.099)	(0.040)	(0.030)	(0.009)	(0.261)	(0.238)	(0.612)	(0.928)	(0.486)
pos	0.0467	0.0501	-0.0117	0.0251	-0.0847	-0.0427	-0.0712	-0.0843	-0.102	-0.0817	0.0655	-0.0463	0.0267
	(0.597)	(0.634)	(0.900)	(0.745)	(0.203)	(0.491)	(0.236)	(0.195)	(0.162)	(0.392)	(0.520)	(0.603)	(0.660)
neglag_1	0.046	-0.0851	-0.0248	-0.0909	0.00237	-0.00583	0.0489	0.122	0.0344	-0.00433	0.0483	0.00457	-0.0951
	(0.675)	(0.517)	(0.831)	(0.350)	(0.977)	(0.941)	(0.525)	(0.156)	(0.715)	(0.974)	(0.706)	(0.968)	(0.193)
poslag_1	-0.344***	-0.364**	-0.396***	-0.319***	-0.312***	-0.270***	-0.265***	-0.234***	-0.316***	-0.2	-0.0787	0.0565	-0.103
	(0.005)	(0.012)	(0.002)	(0.003)	(0.001)	(0.002)	(0.001)	(0.008)	(0.002)	(0.122)	(0.564)	(0.563)	(0.195)
neglag_2	0.0359	-0.224	-0.224*	-0.258**	-0.0859	-0.187**	-0.0797	-0.0137	-0.195*	-0.241*	0.0675	0.0366	-0.0591
	(0.752)	(0.104)	(0.068)	(0.013)	(0.319)	(0.027)	(0.313)	(0.875)	(0.050)	(0.077)	(0.607)	(0.696)	(0.433)
poslag_2	-0.241**	-0.22	-0.284**	-0.211**	-0.166*	-0.162**	-0.185**	-0.157*	-0.297***	-0.236*	0.0728	0.0756	-0.041
	(0.045)	(0.122)	(0.024)	(0.042)	(0.066)	(0.048)	(0.020)	(0.066)	(0.003)	(0.066)	(0.589)	(0.256)	(0.594)
_cons	0.00124	0.000104	0.00143	0.00166	0.000391	-0.000698	-0.00161	-0.00292**	-0.00327**	-0.00106	-0.00156	-0.000282	-0.000732
	(0.434)	(0.956)	(0.385)	(0.227)	(0.748)	(0.547)	(0.156)	(0.025)	(0.022)	(0.570)	(0.401)	(0.855)	(0.648)
N	60	60	60	60	60	60	60	60	60	60	60	63	60
adj. R-sq	0.224	0.158	0.213	0.229	0.157	0.177	0.129	0.168	0.209	0.023	0.435	-0.101	0.003

* p <0.1, ** p < 0.05, *** p<0.01, **** p <0.001

Table 5 – Regression Results on Average Put Volatility Gap

	(1) gap	(2) gap	(3) gap	(4) gap	(5) gap
gap1	−0.0157	−0.0161	0.0177	0.00716	0.0106
	(0.863)	(0.860)	(0.843)	(0.938)	(0.909)
gap2		−0.0204	−0.0254	−0.0195	−0.0200
		(0.823)	(0.776)	(0.830)	(0.831)
neg			0.331	0.0434	0.257
			(0.701)	(0.965)	(0.802)
pos			−1.739**	−2.044**	−2.104**
			(0.006)	(0.008)	(0.010)
neg1				−0.490	−0.0148
				(0.615)	(0.990)
pos1				−0.544	−0.478
				(0.486)	(0.624)
neg2					0.819
					(0.429)
pos2					0.0904
					(0.912)
_cons	0.0210	0.0215	0.0213	0.0216	0.0217
	(0.161)	(0.157)	(0.152)	(0.148)	(0.148)
N	124	124	124	124	124
adj. R-sq	−0.008	−0.016	0.033	0.022	0.010

p-values in parentheses
* p<0.05, ** p<0.01, *** p<0.001

Table 6 – When market goes up, Regression Results on Average Put Volatility Gap

	(1) gap	(2) gap	(3) gap	(4) gap	(5) gap
gap1	−0.0327	−0.0337	0.0484	0.0229	0.0321
	(0.780)	(0.776)	(0.674)	(0.847)	(0.793)
gap2		−0.0304	−0.0623	−0.0313	−0.0243
		(0.800)	(0.586)	(0.792)	(0.843)
neg			0.673	−0.0237	0.189
			(0.575)	(0.987)	(0.902)
pos			−2.494**	−3.243**	−3.171*
			(0.005)	(0.004)	(0.012)
neg1				−1.131	−0.639
				(0.494)	(0.741)
pos1				−1.239	−0.863
				(0.275)	(0.556)
neg2					0.436
					(0.770)
pos2					0.488
					(0.638)
_cons	0.0326	0.0333	0.0292	0.0297	0.0288
	(0.163)	(0.160)	(0.199)	(0.194)	(0.218)
N	59	59	59	59	59
adj. R-sq	−0.016	−0.033	0.079	0.069	0.039

p-values in parentheses
* p<0.05, ** p<0.01, *** p<0.001

Table 7—When market goes down, Regression Results on Average Put Volatility Gap

	(1) gap	(2) gap	(3) gap	(4) gap	(5) gap
gap1	−0.0105	−0.0105	−0.0166	−0.0164	0.00476
	(0.949)	(0.950)	(0.921)	(0.925)	(0.979)
gap2		−0.00774	0.00803	0.00847	−0.00357
		(0.961)	(0.960)	(0.959)	(0.984)
neg			0.0488	0.0565	0.530
			(0.975)	(0.974)	(0.767)
pos			−1.027	−1.020	−1.244
			(0.398)	(0.433)	(0.363)
neg1				0.0248	0.713
				(0.986)	(0.692)
pos1				0.0241	−0.387
				(0.986)	(0.838)
neg2					1.342
					(0.457)
pos2					−0.906
					(0.631)
_cons	0.0119	0.0120	0.0113	0.0113	0.0129
	(0.596)	(0.598)	(0.633)	(0.645)	(0.603)
N	58	58	58	58	58
adj. R-sq	−0.018	−0.036	−0.059	−0.101	−0.118

p-values in parentheses
* p<0.05, ** p<0.01, *** p<0.001

.
.

Appendix 1

Machine Translation Code

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import xlrd
import xlwt
data = xlrd.open_workbook("01_2016.xlsx")
sh = data.sheet_by_name("sheet1")
nrows = sh.nrows

for i in range(1186,nrows):
    content = sh.cell_value(i,5)
    date = sh.cell_value(i,7)
    outputFileName= date + '.txt'
    outputFile = open(outputFileName,'w')

    #driver = webdriver.Chrome()
    driver = webdriver.PhantomJS()
    driver.get("http://translate.google.com")

    # Input the Chinese words for translation
    driver.find_element_by_id("source").send_keys(content)
    driver.find_element_by_id("gt-submit").click()

    # Translation Process involves redirection
    # Apply handler to switch to the current working page
    handle=driver.current_window_handle
    driver.switch_to.window(handle)

    #Wait until the translation process finish to get the result
    signal= WebDriverWait(driver, 30, 0.5).until (
        EC.presence_of_element_located((By.CSS_SELECTOR,"span#result_box>span"))
    )
    elements = driver.find_elements_by_css_selector("span#result_box>span");
    for element in elements:
         outputFile.write(element.text)

    outputFile.close()
    #Quit the Chrome
    driver.quit()
```

Appendix 2

Sentiment Analysis Code

```
from nltk.sentiment.vader import SentimentIntensityAnalyzer
import os
import xlwt

os.chdir("/Users/Jessica/Desktop/Weibo Data/translated")
dir_list = os.listdir(".")

#open a Excel Spreadsheet to store the sentimental analysis result
book = xlwt.Workbook()
sheet = book.add_sheet('sheet1',cell_overwrite_ok=True)
sheet.write(0,0,'date')
sheet.write(0,1,'compound')
sheet.write(0,2,'neg')
sheet.write(0,3,'neu')
sheet.write(0,4,'pos')
row = 1

for file_name in dir_list:
    if len(file_name)>12:
        #open a translated weibo text file
        f =open(file_name,'r')
        paragraph = f.read()
        f.close()
        #note the date, which is the first 11 characters of the file name
        sheet.write(row,0,file_name[:11])

        #sentimental analysis
        sid = SentimentIntensityAnalyzer()
        ss = sid.polarity_scores(paragraph)

        #write the sentimental results into an excel spreadsheet
        column = 1
        for k in sorted(ss):
            sheet.write(row,column,ss[k])
            column +=1
        row +=1
        column = 1

book.save('senti_score.xls')
```

Work Cited

Bollen, J., Mao, H. and Zeng, X.-J. 2010. Twitter mood predicts the stock market. Journal of Computational Science 2(1): 1-8

Wang, Gang, Tianyi Wang, Bolun Wang, Divya Sambasivan, Zengbin Zhang, Haitao Zheng, and Ben Y. Zhao. "Crowds on Wall Street." Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing - CSCW '15 (2015): n. pag. Web.

Mayhew, Stewart. "Implied Volatility." Financial Analysts Journal 51.4 (n.d.) Web.

Luo, Tiejian, et al. Sentiment Analysis. Trust-based Collective View Prediction. Springer New York, 2013:53-68.