주제관련 정보 정리

대분류

기능 사항 2가지

https://www.mk.co.kr/news/stock/view/2020/03/295515/

1. 문서 분석 자연어 처리 엔진 (증권사 리포트 분석 및 뉴스 데이터 분석)
   ⇒ Market insight 도출
2. 시계열 데이터 예측 엔진 (주식 거래량 예측 서비스)

자연어 처리 기능.

1. 데이터 종류 (사람들에게 영향을 주는 것들을 생각)
   - 네이버 증권 종목 토론실 / 리포트 정리
   - 유튜브
2. 데이터 크롤링
3. 자연어 처리 분석 방법

자연어 처리 분석 방법

1. 감성분석의 주요 기법 3가지

https://junpyopark.github.io/social_stock/

표 1. 센티멘트 분석 주요 기법

분석기법	설명
Machine Learning Approach	- 사전에 긍정/부정으로 분류된 학습데이터로 텍스트의 긍정/부정 의견을 분류하는 방식 (SVMs(Support Vector Machines)이 주로 쓰임).
Lexicon-based Approach	- 사전에 정의된 긍정/부정 단어를 이용하여 텍스트에 포함된 긍정/부정 단어의 출현 빈도로 긍정과 부정을 판별하는 방식.
Linguistic Approach	- 텍스트의 문법적인 구조를 파악하여 극성을 판별하는 방식.

여러 가지 감성 분석 기법들

또 영문 번역을 실시하여 감성 분석을 진행하기 때문에 오역 문제를 발생시킬 가능성이 크게 됩니다. 따라서 이 논문에서는 문맥에 따라 극성을 판단하는 Linguistic 방법이 아닌 **긍정과 부정 단어의 빈도를 판별하는 Lexicon-based 방법을 사용합니다.**

이제 이런 분석 과정을 거치게 되면 게시물 별 극성 값이 계산되어 나오게 됩니다. **정성적 데이터를 정량적 수치 데이터로 변환해 준 것입니다.** 극성 값이 0보다 크면 긍정, 0보다 작은 경우 부정으로 설정합니다. 즉, 감성 분석을 통해 해당 게시물의 긍정 단어 개수가 부정 단어 개수보다 많으면 긍정, 부정 단어 개수가 긍정 단어 개수보다 많으면 부정으로 정의합니다. 이렇게 나온 긍정과 부정에 각각 상승과 하락을 매치하여 주가 예측을 실행합니다.

감성사전에 단어별로 **극성의 가중치를 두어 계산**하면 더욱 유의미한 분석 결과를 얻을 수 있지만 한글을 영문으로 **번역하는 절차에서** 오역 문제를 보완하고자 단순히 긍정과 부정으로 나뉜 감성사전을 활용하여 어휘의 **빈도를 통한 감성 분석을 진행하였습니다.**

렉시콘을 이용한 단어 정의를 통해서 부정과 긍정으로 분류하여 상승과 하락을 매치하여 주가 예측을 실행한다.

시계열 데이터 엔진

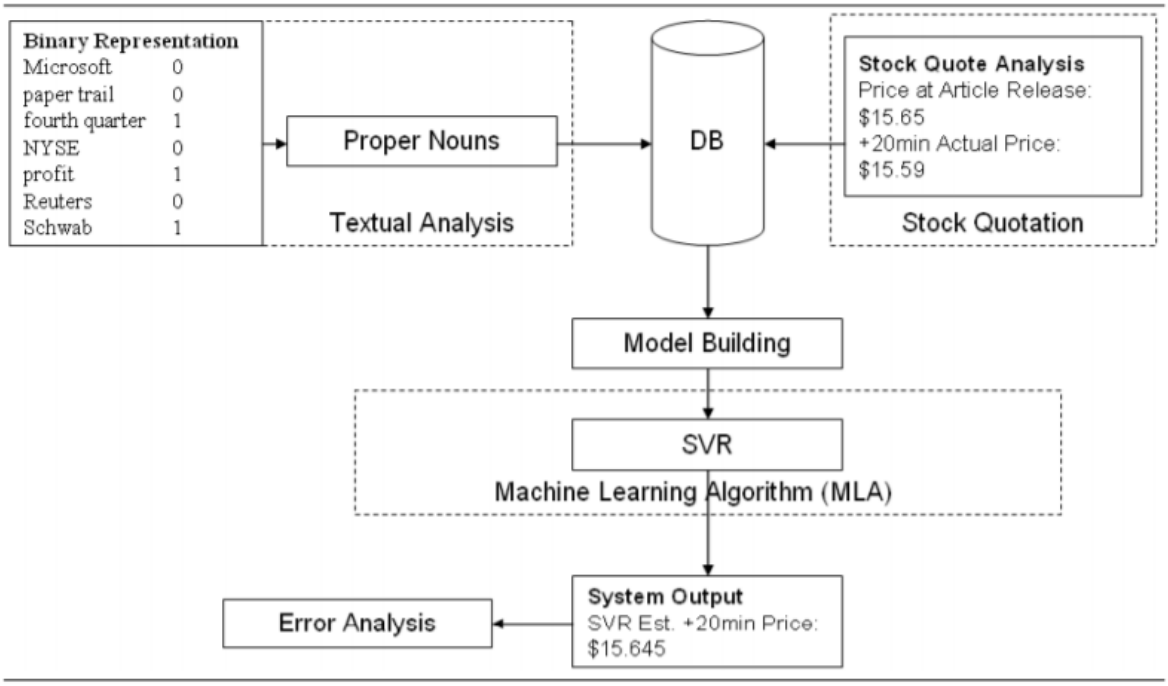

Figure 3. Example AZFinText representation

여기서는 SVM 대신에 SVR을 사용하였는데 왜 그런지를 알아보아야 한다.

<원문 발췌> # 이용하다가 막히면 살펴 볼것

Figure 3 shows an example usage of the AZFinText system. The first step is to extract all
article terms from every article in the corpora. Second, terms are identified by their parts of
speech. Third, the entire set of Proper Nouns are represented in binary as either present or not in
each individual article. Fourth, the price of the stock at the time the article was released, is then
appended to each news article at the model building stage. For this particular article, the price of
Schwab stock was $15.65 at the time of article release and the +20 minute stock price was
$15.59. We selected the +20 minute interval to make our predictions because of its prior
representation as a small window of opportunity in textual financial research [8, 9]. Once the

model is built, machine learning takes place via the SVR algorithm. The SVR is fed a matrix of
11
proper nouns, coded in binary as present or not in the article, as well as the price of the stock at
the time the article was released. This is done for each textual financial news article and the
SVR component makes a discrete prediction of what the +20 minute stock should be. In this
instance, the output price is $15.645.
After training we analyze the data using a Simulated Trading Engine that invests $1,000
per trade and will buy/short the stock if the predicted +20 minute stock price is greater than or
equal to 1% movement from the stock price at the time the article was released [4, 8, 9]. Any
bought/shorted stocks are then sold after 20 minutes.
<A Discrete Stock Price Prediction Engine Based on Financial News>