Using Multi-View Learning to Improve Detection of Investor Sentiments on Twitter

Zvi Ben-Ami[1], Ronen Feldman[2], and Binyamin Rosenfeld[2]

[1] The Hebrew University, School of Business Administration, Jerusalem, Israel

[2] Digital Trowel, New York, USA

{zvi.benami, ronen.feldman}@mail.huji.ac.il, grurgrur@gmail.com

Abstract. Stock-related messages on social media have several interesting properties regarding the sentiment analysis (SA) task. On the one hand, the analysis is particularly challenging, because of frequent typos, bad grammar, and idiosyncratic expressions specific to the domain and media. On the other hand, stock-related messages primarily refer to the state of specific entities – companies and their stocks, at specific times (of sending). This state is an objective property and even has a measurable numeric characteristic, namely, the stock price. Given a large dataset of twitter messages, we can create two separate "views" on the dataset by analyzing text of messages and external properties separately. With this, we can expand the coverage of generic SA tools and learn new sentiment expressions. In this paper, we experiment with this learning method, comparing several types of general SA tools and sets of external properties. The method is shown to produce significant improvement in accuracy.

Keywords. Sentiment analysis, sentiment expression mining, unsupervised learning, multi-view learning, investors' sentiments, social media.

1 Introduction

When choosing their investments on the stock market, investors rely, to a large extent, on company news, press releases, publications, analysts' recommendations and other qualitative information, which eventually affects stock prices [1, 11, 25]. Online investment message boards such as Yahoo! Finance[1], Raging Bull[2] and StockTwits[3], which since the formation of Web 2.0 have become increasingly popular [3], allow investors to share their trading ideas, advice and various opinions on public companies, securities' indices and other financial instruments of their interest. These investment message boards are considered by many investors as a highly valuable source for making their trading decisions [14] and can open an atypical window to typical investors' explicit thoughts and doubts as well as other opinions on the underlining securities. As a result, potential applications of accurately analyzing sentiments in such online messages can be numerous and may benefit investors, public companies as well as various financial portals and microblogs.

Having said the above, correctly identifying the sentiments in such securities-related messages is not a trivial matter. Traditional sentiment classification algorithms tend to perform poorly when applied to social financial messages. One of the main reasons is an abundance of domain-specific sentiment-implying expressions (other reasons are described in the next section). Thus, the task of automatically learning such expressions has great importance in this domain.

This task, learning stock-related sentiments, has special properties compared to learning sentiments in other domains. The difference lies in the fact that sentiments are related to the state of entities (companies and their stocks) at specific moments in time. This is unlike, for example, the states of consumer products, books and movies,

[1] http://finance.yahoo.com

[2] http://www.ragingbull.com

[3] http://stocktwits.com

ISSN 1405-5546

DOI: 10.13053/CyS-18-3-2019

etc., which always stay static, whereas the state of stocks constantly changes. Moreover, these states have an objective numerical measure – the current stock price, which is known for all messages in the past.

In this paper, we investigate a "multi-view" classification-based method of using extra-textual information about messages to facilitate unsupervised learning of domain-specific sentiment expressions. In principle, the method is applicable to any domain which is rich in such additional information (external to the message text). The stock sentiments are particularly suitable, due to the specific properties described above.

The proposed approach is applicable to any Sentiment Analysis (SA) architecture that is able to directly utilize individual sentiment expressions. We apply the method to an SA system based on CARE-II platform for building relation extraction systems [18].

The structure of this paper is as follows: the following section presents the complexity and challenges associated with analyzing finance-related messages. Section 3 summarizes the related work. Section 4 describes in detail our methodological approach for multi-view learning. Section 5 describes the specific SA system that we use for our experiments. Section 6 presents the experimental design and results. Section 7 concludes the work and proposes future studies.

2 The Complexity of Messages on Financial Securities

Determining the sentiment in a given text on financial securities requires far more than merely identifying whether the words used in the text have generally positive or negative connotations. As presented in Figure 1, many different elements should be considered before one classifies a text as being positive or negative. First, the author of a given text should be identified, as the position of the author may have a critical effect on determining the sentiment orientation. A positive sentiment for an investor holding a short position in a given company is a negative sentiment for the company and vice versa. Questions can also be asked about the expertise and know-how of the author, as it was investigated in [3], and higher weight can be assigned to sentiments from authors with more expertise. Further, authors with diverse interests can assign different sentiments for similar events.

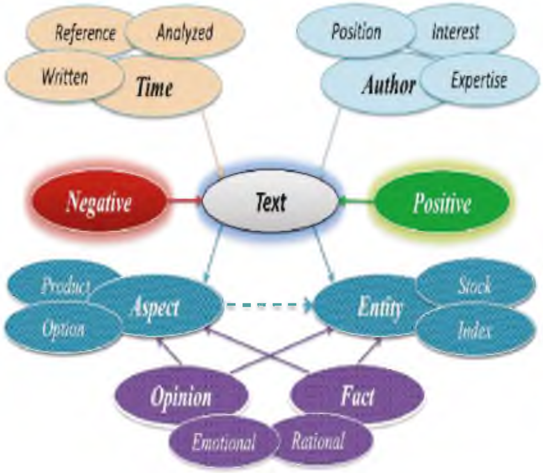

Fig. 1. The complexity of identifying sentiment in financial messages

Time also plays an important role in several dimensions. Sentiments on financial securities are very much time-dependent and it is important to know when a message was written and when it was analyzed in order to determine if the expressed sentiment still holds. For example, any option on a stock has an expiry date and sentiments embedded in messages on such options are restricted in time, for example: "*@chrisbeanie1 is $fb your play tonight? I'm short the July 27, 28 puts at 1.55. Happy to own it at 26.50*".

In addition, sentiment pertaining to stocks can refer to future expectations from the stocks or to the stocks' past performances. For example, the message *"$IMUC had a recent pull-back, will gain momentum in the next few weeks, get in on the way up!*" is negative with regard to the past and positive with regard to the future. Future expectations are much more important pieces of information for two reasons. The first is technical, as past performances of a stock can easily be learned from its historical prices. The second is practical: investors are typically much more interested in what will happen than in what already did.

An additional element to consider pertains to the fact that a text can talk about a specific financial entity, such as a stock or an index, but it can also refer to an aspect of that entity, for instance a call or put options on the stock or a product which is related to the corresponding company. Similar words may have a reverse sentiment when referring to different aspects, e.g., a profitable short option indicates a negative sentiment on the company, but a profitable IPO is positive for the company.

A text may state some facts on a given entity or any of its aspects, or outline the author's opinion of them. Finally, both the factual information and the opinionated information can be classified as being rational or emotional. Emotional facts, such as "*The Stock 'XYZ' rose only by 0.5%*", can bear an important sentiment that should not be ignored.

Only after we have such a comprehensive view on a written text, are we in a position to determine the text sentimental polarity. In this paper, we disregard many of those points, focusing on learning local sentiment expressions. We always assume that a message sentiment can always be identified from the message text (and nothing else) by simply reading and understanding the message, even though in reality it may be not so. Even so, the task is still challenging for general-purpose SA systems, because of the abundance of domain-specific expressions which indicate sentiments only within the financial domain, or even specifically within the domain of Twitter stock-related messages.

It is very time-consuming to identify such expressions manually. Thus, unsupervised or semi-supervised learning approaches for discovering such expressions are very useful.

2.1 Challenges of Analyzing Financial Messages

Analyzing financial messages on social media brings about the need to deal with some specific challenges that are associated with financial text, as well as other challenges associated with analyzing messages on social media in general. The latter pertains to the unique style of language used on social media which involves using short sentences, often omitting conjunctions, pronouns, and so on. Sarcasm and cynicism, which are much more common in social media, are extremely difficult to analyze correctly [12]. For example, the message: *"It's rumored that Facebook will announce a new kind of free stock dividend tonight, namely CrashVille. $FB $ZNGA #SoCalledEarnings"* will be tagged as positive by any tagger failing to understand the deep semantic meaning of the text. Further, textual structures on social media messages are more diverse and complex than textual structures found in news articles, as people using financial social media often have different writing skills and styles.

Challenges that are unique to financial social media messages concern, for instance, the fact that sentiment on financial securities is price-dependent and has a range. It is common to find messages that specify the support and resistance levels, i.e., the price levels that the authors believe the stock price will not go above or below. The following message, for example, depends on the up-to-date price of the security: *"$SINA watch $57.50 which now is a resistance, $59.21 that will open the door to $62.48 before earning, if not we will break $50*". If the price at the time of the analysis is 59.21, the sentiment is very positive, as the author believes it can reach 62.48, otherwise, it is negative as the author believes it can drop to $50.

As in any study on sentiment analysis, careful attention should be given to sentiment modifiers (e.g., "highly"), emphasis modifiers (e.g., "mostly"), opposite modifiers (e.g., "far from") and sentiment shifters (e.g., "not"). Anaphora resolution is also a challenge when conducting sentiment analysis, although, short messages on social media often bear little anaphora.

3 Related Work

The task of analyzing sentiments buried in digital text in general and in social media in particular has been studied widely in the past decade [6, 12, 13, 16, 23]. There are two common approaches to text classification: supervised and unsupervised learning. The underlying assumptions of the supervised approach is that there is a finite set of classes into which the text can be classified and that training data is available for each class. A simple sentiment classification is positive or negative. More sophisticated classifications can also include a neutral class or have some discrete numeric scale in which the strength of the sentiment in the text can be classified. In the supervised approach, given the training data, the system learns a classification model by using one of the common classification algorithms such as SVM, Naïve Bayes, Logistic Regression or KNN. When a numeric value (in some finite range) is to be assigned to the classified text (e.g., the five star system used by Amazon), a regression analysis can be applied.

In [17], the authors have shown that good accuracy can be achieved even when the text is represented as a simple ‘bag of words’. More advanced representations utilized TF*IDF [20], POS information, sentiment lexicons and parse structures. The main disadvantage of using a supervised approach for sentiment analysis is that the method requires a large amount of classified data for each domain and for each possible class within the domain, which is an expensive exercise to generate.

Unsupervised approaches to sentiment analysis are based on determining the semantic orientation of specific phrases within the text. If the average semantic orientation of these phrases is above some predefined threshold, the text is classified as positive. Otherwise, it is deemed negative. A selection of the phrases can be done using a set of predefined POS patterns (such as in [23]) or by using a lexicon of sentiment words and phrases (such as in [22]). One of the classic methods used to determine the semantic orientation of a given word or phrase was presented by the authors of [23], who proposed to calculate the statistical dependence between the PMI (Pointwise Mutual Information) of a given phrase with two sentiment words (excellent, poor), over a corpus or over the web (by utilizing web search queries). Subsequent unsupervised approaches (e.g., [26]) used a modified log-likelihood ratio instead of PMI, while others [7, 22] used a dictionary of sentiment words and phrases with their associated orientations and strength, together with incorporated intensification and negation to compute a sentiment score.

Both the supervised and unsupervised approaches may work well in some scenarios when either the whole document or each individual sentence within it refers to a single entity. In many cases, however, people talk about entities that have many aspects (sometimes called attributes or features) and they have a different opinion about each of them. This often happens in reviews about products or in discussion forums dedicated to specific product categories.

Simple models for sentiment analysis suggest a document level analysis, assuming that a document contains an opinion on one main object expressed by the author of the document (e.g., [17]). More sophisticated models that acknowledged the downsides in such a lenient assumption proposed to look at the sentence level of analysis [26]. Since both the document-level and sentence-level analysis approaches do not discover exactly what people like and don’t like, recently developed models are proposing a finer-grained aspect-level analysis [12].

A unified solution to the problem of sentiment analysis is simply impossible since similar terms and phrases may have different meanings in different domains [13, 24], and within a domain, the same term or phrase may have different sentimental polarity on different aspects or attributes of the same sentiments associated with the entity of interest [12].

Sentiments in financial text drew the interest of researchers from the field of Computer Science [3, 4, 6, 7, 15, 19] as well as from the fields of Economics and Business Management [2, 9, 14]. Studies on finance text in social media have chosen to take simplified approaches to deal with the complex problem of such sentiment analysis. Researchers from the fields of Economics and Business Management tended to simplify the problem with regard to the task of textual analysis and used lenient approaches to text classification such as ‘a bag of words’ and ‘Naïve Bayes’ [2, 14]. Researchers from the field of Computer

Science tended to simplify the problem with regard to the semantic financial meaning of the text and looked at general public moods [4, 10, 27] or only at some specific features of messages, such as the number of retweets and hashtags [19].

The authors of [27] used mood words such as “fear”, “worry”, and “hope” in tweets in general to determine the collective emotion on twitter, regardless to whether or not relevant tickers or companies were mentioned in those tweets. The researchers in [4] investigated whether measurements of collective mood states derived from OpinionFinder and Google-profile of Mood State are correlated to the value of the DJIA over time. The authors of [19] measured the correlation of the activity on Twitter with the changes in the stock prices and trading volume. They did look only at relevant messages by using several filtering methods, however, disregarded the sentimental content on the tweets and paid attention only to different features related to the activity and graphs of tweets in which a given company is mentioned. The authors of [15] used Lexical Scorer and the ‘bag of words’ methods to determine sentiment in financial tweets and tried to associate the sentiments with the stocks’ performances.

The authors of [5] looked at a set of blog posts and corresponding comments on selected firms. They used a SVM regressor on stock market movement and a set of features including the number of posts, the number of comments, the length and response time of comments, strength of comments and various information roles that can be acquired by people. The authors of [5] also did not attempt to study the contextual meaning of these messages.

The authors of [2, 14] classified messages as buy, hold, or sell by using a Naïve Bayes method. The researchers in [9] looked at the impact of a divergence of opinions on the price and volume reactions to earnings announcements by using Maximum Entropy. The shortfall of the above-mentioned approaches is that they analyze the sentiment on the document level and assume that each message contains one sentiment on one entity, an assumption which does not hold true in many cases.

The authors of [3] proposed a general framework for identifying expert investors, and used it as a basis for several models that predict stock rise from stock microblogging messages. This work looked much deeper into the context of financial messages and tried to identify particular message types. They distinguished between facts and opinions and further classified facts into news, chart pattern, trade, and trade outcome. The also classified the opinions into speculation, chart prediction, recommendation, and sentiment. This work is different than the work presented in the current paper in two ways. First, our paper doesn’t attempt to identify expert investors; it rather attempts to identify sentiment in any financial message. Second, our paper does not only look at some particular types of messages, such as only at those indicating an actual transaction, and aims at classifying messages as being positive, negative or neutral with respect to any given financial security.

4 Mining Domain-Specific Sentimental Expressions Using Multiple Views

As shown above, one of the difficulties with sentiment analysis in specific domains lies in the existence of domain-specific sentiment expressions. It is very time-consuming to identify and save them manually. Therefore, unsupervised methods of discovering them are very useful. An unlabeled corpus of domain-related messages is easy to obtain.

Generic domain-independent SA systems are available and can be used for seeding the learning process. Some sentiment expressions are language-wide and applicable to any domain. Systems based only on such expressions are also generally applicable. However, such systems have limited accuracy. Their recall is low because of missing domain-specific expressions. Also, they are often not very precise, because language-wide sentiment expressions frequently change their meaning in specific contexts. One of our goals is to use a large domain-specific corpus together with a general-purpose SA in order to mine for domain-specific sentiment expressions, which can extend the SA to improve its accuracy.

For sentimental analysis in the StockTwits domain, there is another possible source of sentiment information, namely, the objective state of the target stock. By noting the correlations between the movement of stock prices and various expressions occurring in messages generated within its time frame, it should be possible to identify positive-sentiment expressions as correlating with upward movement of the stock prices, and negative-sentiment expressions with downward movement of the stock prices. While using either of the methods is plausible, the best way should be to combine the various methods into a single overall strategy. In this paper, we experiment with this idea.

4.1 Basic Learning Model

Assume first that we have a large corpus $T = \{t_1, t_2, ...\}$ of text messages. Each message t has a true polarity $Pol(t) \in POLS = \{POS, NEG, NEUTRAL\}$, which can always be identified (by people) by reading and understanding the text. We make a further assumption that the polarity of a message is determined by the occurrence of various sentiment expressions within the text of a given message.

The nature of the expressions is not essential at this point. They can be individual words, multi-word sequences, sequences with gaps, syntactic patterns, etc. The important point is that given a message t_i, it must be easy to list all expressions occurring within it. For the purposes of sentiment expression learning, we represent message texts as bags-of-expressions: $t_i = \{w_{i1}, w_{i2}, ...\}$. We will assume that each expression has a true polarity $Pol(w) \in POLS$ and that the polarity of a message is determined by the polarities of expressions within this message. Most of the expressions are neutral, and if a message contains only neutral expressions, the message itself is also neutral. If a message contains a positive or a negative expression, then the message itself is positive or negative, respectively.

For simplicity, and since polar expressions are assumed to be relatively rare, we dismiss the case where more than one polar expression occurs within a message. Note that all these simplifications are used only for the learning model, not for actual sentimental analysis. The anticipation is that, while the simplifications are inaccurate and sometimes broad, they still largely correlate with the true sentiments, and so by investigating a large corpus, any local irregularities will be smoothed over.

If the true polarity $Pol(t)$ is known for each message $t \in T$, there would be a natural way to search for sentimental expressions by simply counting their occurrences. However, in a large unlabeled corpus, the true polarities of messages are unknown.

On the other hand, the polarity of a message is causally-independently influenced by external (relative to the message's text) factors: if a given stock is doing good or bad on a given day, then the polarity of the messages about that stock would tend to be correspondingly positive or negative; messages from the same author about the same stock would tend to have the same polarities, etc.

Thus, we have two parallel views on a set of messages, which allow multi-view learning.

4.2 Using Parallel Views

Given a large corpus T, we first process it with some text-based SA (sentiment analysis) system, which performs as a function $SA: T \rightarrow POLS$, producing a classification $T = T_{SA\text{-}POS} \cup T_{SA\text{-}NEG} \cup T_{SA\text{-}NEUTRAL}$. The SA system is assumed to have a relatively high precision for polarized messages, but insufficient recall. Thus, while generally $T_{SA\text{-}POS}$ and $T_{SA\text{-}NEG}$ contain mostly positive and negative messages, respectively, $T_{SA\text{-}NEUTRAL}$ cannot be assumed to contain only neutral messages. It is also much bigger than the two polarized sets.

Now, using the second "view", we process the corpus T with a feature extractor, which generates a real-valued high-dimensional vector for each message, using any possible properties of it that are conditionally independent from its text content given its polarity. Using this representation, we train a binary SVM classifier with $T_{SA\text{-}POS}$ and $T_{SA\text{-}NEG}$ as the training data. This classifier then produces a score $f(t)$ for each message t in $T_{SA\text{-}NEUTRAL}$.

The properties of $f(t)$ are significant because it is grounded in generic SA and external properties, its sign and magnitude correlates with the true polarity of t, but it is independent from the text patterns within t (conditional on the true polarity of t).

We use f as follows.

Let there be a previously unknown text pattern w, appearing in $T_{SA\text{-}NEUTRAL}$. We are interested in probabilistically estimating the polarity of w, that is, in the value of $P(Pol(w) = \mathrm{A})$, where $\mathrm{A} \in \{\mathrm{POS}, \mathrm{NEG}\}$.

Let $T_w = \{t \in T_{SA\text{-}NEUTRAL}\colon w \in t\}$ be the set of all messages containing w. Then the probability $P(Pol(w) = A)$ can be estimated from $f(t)$ scores of messages in T_w:

$$\begin{aligned} P(Pol(w) = A \mid T_w, f) \\ = \frac{P(f(T_w) \mid Pol(w) = A)}{P(f(T_w))} \\ * P(Pol(w) = A). \end{aligned} \tag{1}$$

The constant prior $P(Pol(w) = A)$ can be ignored, assuming the set T_w is sufficiently large. In the main factor, we can safely assume that the different messages in T_w are independent from each other, so

$$\begin{aligned} P(Pol(w) = A \mid T_w, f) \\ \sim \prod_{t \in T_w} \frac{P(f(t) \mid Pol(w) = A)}{P\big(f(t)\big)}. \end{aligned} \tag{2}$$

Here, the marginal $P(f(t))$ can be estimated directly from the SVM classifier's scores on T. In the other part of the formula above, we only deal with messages that contain w whose polarity is non-neutral. According to our simplifying assumptions, we proceed as if there might be no conflicts, and the polarities of all messages in T_w are equal to the polarity of w. Then, the likelihood $P(f(t) \mid Pol(w) = \mathrm{A})$ can be reduced to

$$P(f(t) \mid Pol(w) = A) \approx \frac{P(f(t) \,\&\, Pol(t) = A)}{P(Pol(t) = A)}. \tag{3}$$

The constant marginal $P(Pol(t) = A)$ can be directly estimated from a manually labeled development test set. And the rest can be estimated using the SA-polarized sets.

Now, because of conditional independence of $SA(t)$ from $f(t)$, given the true polarity of t, we have

$$\begin{aligned} & P(f(t) \& Pol(t) = A) \\ & = \sum_{B \in POLS} P(f(t) \,\&\, Pol(t) = A \,\&\, SA(t) = B) \\ & = \sum_{B \in POLS} P(f(t) \mid SA(t) = B) \\ & \qquad * P(SA(t) = B \mid Pol(t) = A) \\ & \qquad * P(Pol(t) = A) \end{aligned} \tag{4}$$

and $(f(t) \mid SA(t)=B)$ as well as $P(SA(t)=B)$ are estimated directly from the SA results and SVM results. $P(SA(t) = B \mid Pol(t) = A)$ is estimated on a development test set.

4.3 Features for the "External" View

The external view can use message properties that are independent from the message text. For the experiments in this paper, we use the properties which follow.

1. **Stock-price-related features.** These are related to the price of the stock referenced in the messages, within some time frame of the message post time. The numerical values of the stock prices cannot be used directly, because they vary widely from stock to stock. However, we can identify and use 'price change events' – the points in time where the price of a stock significantly changed from one day to the next. There are many different possible adjustable parameters for identifying the useful price changes, and there is no a priori reason to select any particular numbers. We, therefore, choose several different reasonable values, and let an SVM classifier training algorithm choose the best. We use all possible combinations of the following:

    i. SMALL is when the price changed by 2–5%, LARGE is when the change is at least 5%, NOCHANGE if the change is less than 1%.

    ii. YESTERDAY is when the change occurs between yesterday's closing price (relative to the message post time) and today's opening price. TODAY is for changes between today's opening and closing prices, and TOMORROW is for changes between today's closing and tomorrow's opening prices.

iii. PLUS when the price increased, and MINUS when it decreased.

We use all combinations of these properties as binary features, and also all possible intersections of pairs of them.

2. **Seed SA-related features**. These features utilize more directly the connections between messages established by the identity of their authors and/or subject. Again, it is a priori unclear which specific properties are important, so we use all of them and let the SVM decide. We used the properties as follows.

   i. Let POS, NEG, NEUTRAL stand for the binary property of some messages (not the target's ones!) having overall positive, overall negative, and overall neutral sentiment labeling, respectively, according to the seed SA. Also, let HASPOS, HASNEG, and HASANY stand for there being some positive, some negative, and some polarized sentiment expression within the messages. (Thus, for example, POS and HASNEG may both be true for a message, if a negative sentiment expression occurs within it, but the overall sentiment is positive).

   ii. Given some set of messages, let EXIST_X be a binary property, of at least one of the messages satisfying X from (i). Also, let SUM_X, and AVERAGE_X be the real-valued sum and average of X over the messages in the set.

   iii. Given a target message, let DAYBEFORE, DAYAFTER, WITHINDAY, WITHINHOUR, WITHIN10MIN be the sets of messages posted the day before the message, the day after, within the same day, within the same hour, and within 10 minutes, respectively.

   iv. Given a target message, let SAMEAUTHOR and ANYAUTHOR be the sets of messages posted by the same author and by any author, respectively.

We use all possible combinations of these properties as features. At first glance, it may seem that the SA-related features are not independent from the message text, since they are based on the SA which does analyze the text. However, the way we use them show that the dependence is always mediated by the true state of the external entities (authors and stocks), and so the features are conditionally independent from the text given the polarity, which is what we need.

3. **Intersection properties.** These are intersections of the stock-related features and the seed SA-related features. Theoretically, using these is equivalent to using a quadratic kernel for the SVM classifier.

4.4 Pattern-based Filtering

This is a pattern-centric learning method, different from the message-based learning described above. The method uses a different learning model. It is based on the observation that different sentiment expressions occurring within the same message generally tend to be of the same polarity. Thus, given a candidate expression, we can ascertain its polarity by observing all messages that contain the expression which was labeled by the seed SA.

The method cannot be directly incorporated into the above-described learning model, because it directly uses the message text and does not satisfy the independence condition. However, the method can be used to perform an additional filtering step for the learned expressions, significantly improving the precision of the overall learning process.

4.5 Learning Architecture and Data Flow

The data flow of the full learning system is schematically represented in Figure 2.

5 Sentiment Analysis Systems

The basic architecture of the SA system determines the kinds of sentiment expressions that can be used and learned. It is also the seed SA used for starting off the learning process and for calculating the SA-related features.

Since the goal is to seek new polarity expressions, the SA system must be able to use

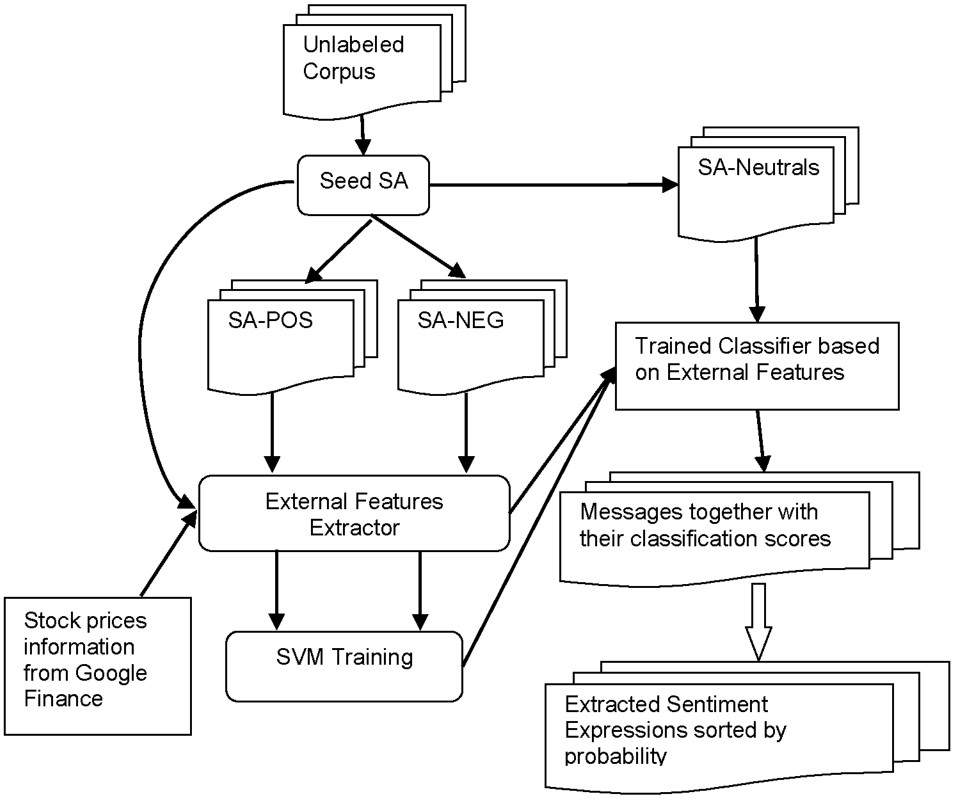
Unlabeled Corpus
Seed SA
SA-Neutrals
SA-POS
SA-NEG
Trained Classifier based on External Features
External Features Extractor
Messages together with their classification scores
Stock prices information from Google Finance
SVM Training
Extracted Sentiment Expressions sorted by probability

Fig. 2. Architecture of the learning system

them directly. This immediately eliminates from consideration some SA architectures, such as bag-of-words classification-based ones (they should also be eliminated on independent grounds, as explained in Section 2). Mainly, we experiment with an SA architecture, which is based on the CARE-II-HPSG parser and a relation extraction system, described in more detail in [18]. The architecture is briefly presented below.

We experiment with two versions of the system: GenericSA, a general-purpose SA system, which contains only language-wide sentiment expressions, and FinancialSA, which is an extension of GenericSA, created by manually adding many financial domain-specific sentiment expressions. For the baseline, we also use DictionarySA, a simple system that classifies a message into positive, negative, or neutral categories according to the number of sentiment expressions that occur within the message. The expressions are the words and multi-word sequences taken from the GenericSA, without the additional syntactic information.

5.1 Architecture of CARE-II-HPSG-based SA Systems

The main SA system used in this paper is based on the CARE-II-HPSG parser and a relation extraction system.

5.1.1. General Description

CARE-II is a domain-independent framework for building Information Extraction systems. The framework includes a grammar description language and the supporting tools. The core of the framework is a parser which is capable of parsing arbitrary weighted typed-feature-structure context-free grammars (WTFSCFGs). These are weighted CFGs, in which every matched symbol, terminal or non-terminal, carries a typed feature structure; the grammar rules have access to the feature structures of their component symbols, building from them the feature structures for their heads by applying the operations of unification, slot extraction, and slot removal.

CARE-II-HPSG is an English grammar written for the CARE-II framework, based on the principles of HPSG grammar theory. The grammar's lexicon is largely underspecified. Only the most frequent and functional words have full definitions, while the open classes of words are defined using generic underspecified lexical entries and tightly-integrated feature-rich sequence classification models for parts-of-speech (POS) and named entities recognition (NER). The models provide weights for different possible typed-feature-structure assignments. Then, for any input sentence, the parser generates a single highest-weight parse – the parse which is the most consistent with both the grammar rules and the NER and POS classifiers.

This architecture results in a relatively fast parser, which generally produces parses of average to low quality. However, when extended with a small set of domain and/or task-specific lexicon entries, the quality of the parses improves dramatically, precisely in the relevant places – at the mention of interesting domain-specific relations. The parsing quality for the rest of the text remains low, but since the rest of the text is of no interest for the information extraction purposes, it is not a drawback. The system is also more robust when handling big and complex sentences and in the presence of bad grammar. The domain-specific lexicon entries also carry semantic information, in the HPSG style. This allows immediate and straightforward extraction of relations and their slots as soon as a parse is generated.

5.1.2. Using CARE-II-HPSG for Sentiment Analysis

When used for sentiment analysis, either general purpose or domain specific, the lexicon is extended to include sentiment words and expressions, which include sentiment labels in the semantic parts of their HPSG feature structures. The labels may indicate polarity of expressions, their intensity, and their combined properties, such as behavior under negation. After a parse of a sentence is generated, it is post-processed by the SA post-processor, which merges sentiments from related expressions, performs coreference (anaphora) resolution, and attaches the sentiments to their targets, where appropriate. The post-processor is rule-based and deterministic.

There are several benefits in using full parsing for the SA task:

- precise identification of sentiment target in cases where several entities are available as possible targets,
- principled and uniform combining interdependent sentiment expressions and processing of negation,
- disambiguation for the cases where polarity of an expression depends on its syntactic role and/or part-of-speech (as in, for example, “fine” as a positive adjective vs. “fine” as a negative noun),
- principled and uniform way of defining multi-word sentiment expressions – using syntactic and semantic links instead of simple word proximity.

The disadvantages of using full parsing are slower processing speed, possible problems with bad grammar and typos, and generally low quality of parses, which may introduce SA errors instead of solving them. However, with a robust parser, these disadvantages should be minimized.

5.1.3. GenericSA, FinancialSA, and DictionarySA

In this paper, we mainly use and compare three different SA systems. GenericSA is described above. FinancialSA is the same as GenericSA, but extended by manually adding many domain-specific lexical entries (for the financial domain, investor sentiments in particular). Finally, DictionarySA is provided as a baseline. It is a very simple SA system that contains a dictionary of sentiment words and word sequences, and classifies a text by counting the number of occurrences of various polarity expressions within it. Whichever polarity occurs most frequently wins. If the number of occurrences is equal, the message is considered neutral. The initial dictionary is the same as used in GenericSA, without the additional syntactic information, such as parts-of-speech, valence, etc.

The type of SA determines what kind of sentiment expressions is available for learning. The simple DictionarySA can only learn words and multi-word sequences. The parser-based SA systems are able to learn more complex patterns. In the present system, the following types of patterns can be learned:

1. Word patterns: individual and compound non-proper nouns, verbs, and adjectives.
2. Valence patterns: head word together with its valence, which can be noun phrases or prepositional phrases complemented by noun phrases. Noun phrases are identified by their head noun, which can be either unrestricted, or restricted to a specific common noun, or restricted to the sentiment target entity type (company name or stock symbol for the financial domain).
3. Modifier patterns: a head word modified by an adjectival phrase or a prepositional phrase complemented by a noun phrase. Same restrictions apply as for valence patterns.

6 Experiments

For the experiments, we use a corpus of several million stock-related messages (twits) collected between May and October 2011. We only use the twits related to stocks for which we were able to collect the price information from Google Finance. For the test set, we use a manually-labeled set of randomly chosen 1500 twits. Another set of 500 twits was used as a development test set for estimating the marginal probabilities and for tuning the final threshold parameter.

6.1 Baseline

In the baseline experiment, we compare the results produced by the three seed SA systems that we use. For reference, we also show the results produced by the Opinion Observer System [7] on the same test set. This system is one of the state-of-the-art general-purpose SA systems. The results are shown in Table 2.

Table 1. Results with neutral messages included produced by several SA systems

	TP	FP	FN	Prec	Recall	F1
Opinion Observer	1239	295	418	0.808	0.748	0.777
Dictionary	1183	223	474	0.841	0.713	0.771
GenericSA	964	94	506	0.911	0.655	0.762
FinancialSA	1066	116	381	0.901	0.736	0.810

Table 2. Results produced by several SA systems

	TP	FP	FN	Prec	Recall	F1
Opinion Observer	365	295	418	0.553	0.466	0.506
Dictionary	308	223	474	0.580	0.393	0.468
GenericSA	159	94	506	0.628	0.239	0.346
FinancialSA	284	116	381	0.710	0.427	0.533

As can be seen from Table 2, the generic SA systems have relatively low accuracy, due to the specifics of the domain. Also notable is that the simple dictionary-based SA is not worse in precision than the much more sophisticated Opinion Observer System in this domain (although much worse in recall). Note, that in Table 2 we only consider positive and negative sentiments when counting the “true positives” (TP). Neutral sentiments are not included in the evaluation, except when they contribute to “false positives”. The results with neutrals included are shown in Table 1.

6.2 Learning

In this experiment, we compare the results produced by learning new sentimental expressions from the twitter messages corpus. In addition to comparing the learning capabilities of the three SA systems, we also compare three sets of external features:

Table 3. Learning capabilities of the SA systems using different external features

	Dictionary			Generic SA			Financial SA		
	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall	F1
Baseline	0.580	0.393	0.468	0.628	0.239	0.346	0.710	0.427	0.533
Price	0.599	0.427	0.498	0.635	0.252	0.361	0.697	0.443	0.541
SA-based	0.588	0.423	0.492	0.643	0.254	0.364	0.699	0.443	0.542
Full	0.589 (+0.9%)	0.428 (+3.5%)	0.495 (+2.7%)	0.635 (+0.7%)	0.262 (+2.3%)	0.370 (+2.4%)	0.700 (-1.0%)	0.518 (+9.1%)	0.595 (+6.2%)

1. PriceOnly set, which contains only features based on stock price;
2. SeedSA-Only set, which contains only features based on using the Seed-SA-produced classification of messages related to the same entities; and
3. Full set, containing both Price-related and SeedSA-related features, as well as their intersections.

The experiments are performed in the following way: given a SeedSA and an external feature set, we process the corpus with the SA, producing three separated sets of messages: SA-POS, SA-NEG, and SA-NEUTRAL. We then train an SVM classifier using the representations in the external feature set of SA-POS and SA-NEG as training data. This classifier is applied to the representations of the messages in SA-NEUTRAL, producing a score for each message within.

Then, for every candidate expression that appears at least ten times for different stocks (so as to eliminate local biases), we calculate its two final probability scores (of being positive and of being negative), using the formulas in Section 4. Given a probability threshold, we append all discovered expressions that pass the threshold to the corresponding seed system, and perform the test.

We found that the final score number, while good for ordering the expressions, is not very indicative numerically. Consequently, we select the best thresholds using a development test set. Finally, we apply the extended SA system on the test set. The results are shown in Table 3.

As it can be seen from Table 3, improvements in accuracy are achieved for all of the seed SAs using any of the feature sets. Somewhat surprisingly, FinancialSA shows the biggest improvement, even though it was the best of the systems from the beginning. This is probably due to the fact that it produces significantly better initial training sets.

For GenericSA and FinancialSA, as expected, the Full feature set produces significantly better results than using either Price or SA-based feature sets separately.

Unexpectedly, for DictionarySA, all three feature sets produce very similar results. The reason for the difference is unclear, and is under investigation.

7 Conclusions and Future Studies

In this paper, we experiment with several SA systems and different external features to identify domain-specific expressions in financial messages on social media. We propose a novel unsupervised multi-view-based approach that uses a seed SA system together with domain-specific external information in order to mine a large corpus of messages for domain-specific sentiment expressions, which, in turn, can extend the SA to improve its accuracy.

The paper is expected to contribute to the body of knowledge on sentiment analysis, in general, and on sentiment analysis of financial social media messages, in particular.

The proposed unsupervised methodological approach to sentimental analysis, which uses multiple views, may be adapted to other studies from other domains requiring the solving of sophisticated sentiment analysis problems.

Our experimental results indicate that our method is successful in integrating diverse sources of external information for the learning purposes. The sources we compare are stock prices on the one hand, and SA results on messages related by subject or by author, on the other hand. Our results show that, when combined in our approach, the sources produce much better accuracy than individually, at least for the best-performing SA systems.

Future studies may incorporate further external views' features, such as events known to have a positive or negative effect on companies, or may address both the time reference mentioned in the messages and the price movement in a corresponding period in order to generate the domain-specific lexicon.

Acknowledgements

We thank Bing Liu for sharing his Opinion Observer System's output with us.

This work is supported by the Israel Ministry of Science and Technology Center of Knowledge in Machine Learning and Artificial Intelligence and the Israel Ministry of Defense.

References

1. **Abarbanell, J.S. & Bushee, B.J. (1997).** Fundamental Analysis , Future Earnings, and Stock Prices. *Journal of Accounting Research*, 35(1), 1–24.
2. **Antweiler, W. & Frank, M.Z. (2004).** Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards. *The Journal of Finance,* 59(3), 1259–1294.
3. **Bar-haim, R., Dinur, E., Feldman, R., Fresko, M., & Goldstein, G. (2011).** Identifying and Following Expert Investors in Stock Microblogs. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*, Edinburgh, Scotland, UK, 1310–1319.
4. **Bollen, J., Mao, H., & Zeng, X.J. (2011).** Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1), 1–8.
5. **De Choudhury, M., Sundaram, H., John, A., & Seligmann, D.D. (2008).** Can blog communication dynamics be correlated with stock market activity?. *Proceedings of the nineteenth ACM conference on Hypertext and hypermedia (HT'08*), Pittsburgh, PA, USA, 55–60.
6. **Connor, B.O., Balasubramanyan, R., Routledge, B.R., & Smith, N.A. (2010).** From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. *Fourth International AAAI Conference on Weblogs and Social Media*, Washington, DC, 122–129.
7. **Ding, X., Liu, B., & Yu, P.S. (2008).** A holistic lexicon-based approach to opinion mining. *2008 International Conference on Web Search and Web Data Mining*, Palo Alto, California, USA, 231–240.
8. **Feldman, R., Rosenfeld, B., Bar-haim, R., & Fresko, M. (2011).** The Stock Sonar - Sentiment Analysis of Stocks Based on a Hybrid Approach. *Proceedings of the Twenty-Third Innovative Applications of Artificial Intelligence Conference*, San Francisco, California, USA.
9. **Giannini, R., Irvine, P.J., & Shu, T.** (**2012**). The Impact of Divergence of Opinions about Earnings using a Social Network.
10. **Gilbert, E. & Karahalios, K. (2010).** Widespread Worry and the Stock Market. *Fourth International AAAI Conference on Weblogs and Social Media*, Washington, DC, USA, 58–65.
11. **Lev, B. & Thiagarajan, S.R. (1993).** Fundamental Information Analysis. *Journal of Accounting Research*, 31(2), 190–215.
12. **Liu, B. (2012).** *Sentiment Analysis and Opinion Mining*. San Rafael, Calif: Morgan & Claypool Publishers.
13. **Loughran, T. & Mcdonald, B. (2010).** When is a Liability not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *Journal of Finance*, 66(1), 35–65.
14. **Sprenger, T.O. & Welpe, I.M. (2010).** *Tweets and Trades: The Information Content of Stock Microblogs* (Early View. Online Version of Record published before inclusion in an issue).
15. **Oh, C. & Sheng, O.R.L. (2011).** Investigating Predictive Power of Stock Micro Blog Sentiment in Forecasting Future Stock Price Directional Movement. *International Conference on Information Systems* (*ICIS 2011)*, Shanghai, China, 1–18.
16. **Pang, B. & Lee, L. (2004).** A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. *42nd*

Annual Meeting on Association for Computational Linguistics (ACL '04), Barcelona, Spain, 271–278.

17. **Pang, B., Lee, L., & Vaithyanathan, S. (2002).** Thumbs up? Sentiment Classification using Machine Learning Techniques. *Conference on Empirical Methods in Natural Language Processing (ACL-02),* Stroudsburg, PA, USA, 10, 79–86.
18. **Rozenfeld, B. & Feldman, R. (2011).** Unsupervised Lexicon Acquisition for HPSG-based Relation Extraction. *Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI'11),* Catalonia, Spain, 1890–1895.
19. **Ruiz, E.J., Hristidis, V., Castillo, C., Gionis, A., & Jaimes, A. (2012).** Correlating financial time series with micro-blogging activity. *Fifth ACM international conference on Web search and data mining (WSDM '12)*, Seattle, Washington, 513–522.
20. **Salton, G. & Buckley, C. (1988).** Term-weighting approaches in automatic text retrieval. *Information Processing & Management: an International Journal,* 24(5), 513–523.
21. **Sprenger, T.O. & Welpe, I.M. (2010).** *Tweets and Trades: The Information Content of Stock Microblogs.*
22. **Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011).** Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics*, 37(2), 267–307.
23. **Turney, P.D. (2002).** Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *40th Annual Meeting on Association for Computational Linguistics (ACL'02)*, Stroudsburg, PA, USA, 417–424.
24. **Turney, P. & Littman, M.L. (2003).** Measuring Praise and Criticism: Inference of Semantic Orientation from Association. *ACM Transactions on Information Systems,* 21(4), 315–346.

Zvi Ben-Ami is a Doctoral candidate at the School of Business Administration of the Hebrew University of Jerusalem. He received his B.A in Insurance form Netanya Academic College in 2000 and his M.Com. in Business Management form University of Port Elizabeth in 2004.

Ronen Feldman is an Associate Professor of Information Systems at the Business School of the Hebrew University in Jerusalem. He received his B.Sc. in Math, Physics and Computer Science from the Hebrew University in 1984 and his Ph.D. in Computer Science from Cornell University in NY in 1993.

Binyamin Rosenfeld is a research scientist at Digital Trowel. He received his B.Sc. in Mathematics and Computer Science from Bar-Ilan University in 1998.

Article received on 07/01/2014, accepted on 01/02/2014.