Combining Machine Learning and Semantic Features in the Classification of Corporate Disclosures

Stefan Evert[1], Philipp Heinrich[1], Klaus Henselmann[2], Ulrich Rabenstein[3], Elisabeth Scherr[2], and Lutz Schröder[3]

[1]Dept. Germanistik und Komparatistik, FAU Erlangen-Nürnberg
[2]School of Business and Economics, FAU Erlangen-Nürnberg
[3]Dept. of Computer Science, FAU Erlangen-Nürnberg

Abstract

We investigate an approach of improving statistical text classification by combining machine learners with an ontology-based identification of domain-specific topic categories. We apply this approach to ad hoc disclosures by public companies. This form of obligatory publicity concerns all information that might affect the stock price; relevant topic categories are governed by stringent regulations. Our goal is to classify disclosures according to their effect on stock prices (negative, neutral, positive). In the feasibility study reported here, we combine natural language parsing with a formal background ontology to recognize disclosures concerning a particular topic, viz. retirement of key personnel. The semantic analysis identifies such disclosures with high precision and recall. We then demonstrate that machine learners benefit from the additional ontology-based information in different prediction tasks.

1 Introduction

With the amount of electronically available information rising, there is increasing interest in developing new means of assessing the semantics of corporate disclosures in order to better handle the high information load. Prior research successfully explores the use of textual analysis to predict stock performance (Bollen et al. 2010; Verchow 2011; Jegadeesh and Wu 2013; Ding et al. 2015), often based on "big data" sources such as Twitter trends. Although our use case also involves the prediction of an econometric variable, accurate prediction of stock prices is *not* our main goal. The broader aim of our work is to extract hidden information from financial texts; we therefore do not make use of additional data sets such as social media to enhance the performance of our task solvers.

In the feasibility study reported here, we aim to improve the performance of a statistical text classifier by integrating knowledge retrieved from the text by an ontology-based reasoner. Our use case is the prediction of stock market reactions after the publication of corporate events according to German law. In so-called *ad hoc disclosures*, companies

have to report any important event that might affect the stock price. Although the relevant types of events are essentially predefined by law,[1] this information is not indicated explicitly in the disclosures and has to be derived from the textual content.

Ad hoc disclosures are a suitable object of the investigation of combining machine learning with ontological reasoning for two reasons: Firstly, these disclosures are supposed to provide information relevant to the stock price and thus offer a straightforward task for machine learning and evaluation (the prediction of stock prices from text). Secondly, companies have an incentive to downplay negative events and hide them "between the lines". Aiming to reveal hidden indicators in the ad hoc disclosures, we focus entirely on their textual content and do not make use of external information from social media or other sources. This makes the prediction task fairly hard and it is thus astonishing that our trained classifiers provide an effective trading strategy. Except for Verchow (2011), who aims at analyzing capital market efficiency and whose unsophisticated computational linguistic methodology leads to rather poor results, we are not aware of any prior work that attempts stock market prediction from ad hoc disclosures.

2 Methodology

The present methodological section is structured as follows: Section 2.1 gives an overview of our corpus, the target variable and the associated prediction task. Section 2.2 briefly introduces the machine learning techniques and their evaluation; section 2.3 motivates the necessity of creating an ontology and outlines its design. Section 2.4 concludes the methodological part by explaining the different ways of integrating machine learners with ontological features[2]; this section also presents further evaluation techniques for the combination of machine learning (ML) with ontological information.

2.1 Data basis and prediction tasks

Corpus We use a sample of announcements of corporate events provided by the DGAP service of the Equity Story AG. Our sample selection starts with over 80,000 mandatory announcements of material events that have been disseminated via the DGAP between mid-1996 and mid-2012. We restrict our analyses to those disclosures that are written in English[3] and that are machine-readable. Due to these constraints and further restrictions on available metadata (see the following paragraph), we obtain a final corpus of 28,287 documents ("textual units") such as the following example:

> Montabaur, December 31, 2001. Michael Scheeren, CFO of United Internet AG and with the company for 11 years, will retire from his position on the Executive Board as of December 31, 2001. It is planned that he will replace Mr. Hans-Peter

[1] See the guideline issued by the Federal Financial Supervisory Authority (BaFin) (2009) for a list of possible price-sensitive events.

[2] We call statistical classifiers and their ontologically enhanced versions more generally "task solvers".

[3] German law requires the material event disclosures to be in German, in another accepted language or in English depending on specific criteria.

Bachmann on the Supervisory Board from January 1, 2002. Scheeren will retain his close ties to the Group as he remains Chairman of the Supervisory Boards of AdLINK AG, 1&1 Internet AG and twenty4help AG. He will also represent United Internet AG on the Supervisory Boards of GMX AG, jobpilot AG and NTplus AG. Mr. Norbert Lang has been named as successor for Michael Scheeren. Lang has been with United Internet since 1994. After first heading the financial department, he joined the United Internet Executive Board one year ago.

Target variable For each ad hoc disclosure i, we measure its effect on the stock market using an event study following prior literature (Strong 1992; McWilliams and Siegel 1997; Corrado 2011). In particular, the market model is used to calculate the market-adjusted stock return surrounding the disclosure date t of the material event:

$$AR_{it} = R_{it} - E(R_{it}) = R_{it} - (\hat{\alpha}_i + \hat{\beta}_i \cdot R_{Mt}) \quad (1)$$

Daily market-adjusted returns or abnormal returns (AR_{it}) are calculated as the deviation between the observed stock return of each individual company (R_{it}) and the expected stock return ($E(R_{it})$). We use the return of the CDAX index as a proxy for the market return and estimate $E(R_{it})$ by regressing a historic series of observed daily stock returns (R_{it}) on the corresponding daily market returns (R_{Mt}) using ordinary least squares (OLS) estimation. The estimation period starts 6 days ($t-6$) and spans up to 155 days ($t-155$) prior to the event date. The estimated intercept ($\hat{\alpha}_i$) and slope ($\hat{\beta}_i$) of the OLS model are then inserted into equation (1) to calculate the abnormal return (AR_{it}).

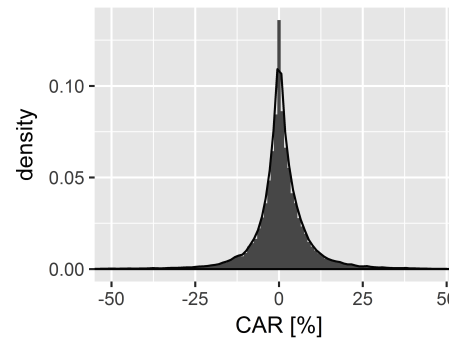

Figure 1: Distribution of the target variable CAR in the corpus (excluding outliers with $|CAR_{it}| > 50$).

We use daily return index data from Thomson Reuters Datastream that is adjusted for capital events (e.g., dividends, stock splits); daily returns are calculated as logarithmic returns (i.e. as $\log(V_f/V_i)$ where V_i is the initial and V_f the final value; use of this quantity is standard to ensure symmetry).[4] In order to account for the fact that part of the information relating to the event is priced early or late, we use an event window of three trading days. Hence, the cumulative abnormal return (CAR_{it}) surrounding each event announcement date (t) is calculated as the sum of the abnormal returns between one day prior ($t-1$) and one day after ($t+1$) the disclosure of the event. The distribution of the target variable is heavy-tailed, slightly skewed, and concentrates around 0 (cf. Figure 1).

Prediction tasks Although the target variable is metric, we abstain from regression analysis for two reasons: Firstly, the data shows heavy tails, which makes it difficult

[4]Non-trading days are excluded.

for regressors to find suitable weights. Secondly, it is far more important in practice to distinguish between positive and negative reactions than to predict the exact degree of the reaction. We hence set ourselves the prediction task of recognizing negative, neutral and positive responses based on a ternary categorization (1/3 of disclosures each); the categories are constructed by means of the respective quantiles of the empirical distribution of CAR.

Since these artificially created categories are hard to distinguish for machine learners – especially if the true CAR value is close to a category boundary – we also analyze the performance of the task solvers in a slightly modified prediction task with more clear-cut categories, i.e. ternary categorization into well-separated categories (20% of most negative and most positive reactions and the 20% closest to the median). We thus refer to the first one of these tasks as the *difficult* prediction task and to its modified version as the *easy* one (see Table 1 for an overview).

	negative	neutral	positive	corpus size
difficult	9,433	9,436	9,418	28,287 (100%)
retirements	*413*	*341*	*292*	*1046 (3.7%)*
easy	5,661	5,645	5,648	16,954 (60%)
retirements	*267*	*205*	*167*	*639 (3.8%)*

Table 1: The two prediction tasks to be solved by the machine learning classifiers and their combinations with ontological features. The **easy** prediction task is a slight modification of the **difficult** one, involving more sharply separated categories. The rows labeled *retirements* show the number of disclosures concerning key personnel turnover in each category (cf. section 2.3).

2.2 ML classification

Our ML classifiers for solving the prediction tasks in table 1 are based on a simple bag-of-words feature set (FM1, cf. section 2.4) with tf.idf weighting.[5] After heuristic deletion of boilerplate footers and headers, removal of stop words, e-mail addresses, URLs, punctuation and numbers, as well as lower-casing and lemmatization, the resulting feature vocabulary contains $n_{voc} = 32{,}401$ lemmas.

We present results for Multinomial Naïve Bayes (MNB) and Logistic Regression (MaxEnt) with $\ell 1$-penalty tuned by 10-fold cross-validation on the training set (for implementation details see Pedregosa et al. 2011). Other machine learning algorithms such as Support Vector Machines and a modified MNB used by Verchow (2011) yielded similar results.

[5]Preliminary experiments including longer n-grams and part of speech tags in the feature matrices did not lead to consistently higher performance.

Evaluation of the ML approach We use accuracy in 10-fold stratified cross-validation (90% training, 10% test data) as a performance measure and compute 95% confidence intervals for the mean accuracy across all 10 folds (based on a normal approximation). Since we have equally-sized categories and stratify the class distribution in the cross-validation,[6] a random baseline classifier achieves an accuracy of $1/3 = 33.3\%$ in our ternary classification tasks.

In order to demonstrate the practical usefulness of our ML approach, we also evaluate the machine learning classifier by means of a simple trading strategy: (1) *buying* if the ML approach predicts category *positive*, (2) *short selling* if it yields *negative*, and (3) *holding* if the result is *neutral*. A scalar performance measure is given by the sum of all individual net gains of CAR values.[7] In this setting, we use constant classifiers that always make the same decision as baselines.

2.3 Ontological feature extraction

Our idea is to use the semantic event categories that regulate the emission of disclosures in the first place in order to improve the ML classifiers. Recall that the disclosures are sent out for very specific reasons, but these are not explicitly mentioned in the text of a disclosure or in the associated metadata. Although the boundaries between different event categories are somewhat fuzzy, most of the disclosures are sent out for one particular reason: manual analysis of a sample of 1,000 disclosures showed that only about 15% fall into more than one topic category.

Motivation for ontological feature extraction The background information about the initial reason to send out the disclosures is valuable and provides a different sort of knowledge than the sort of "semantic information" that can be retrieved from the text itself by unsupervised learning (e.g. automatic clustering of the disclosures). Techniques such as Latent Dirichlet Allocation (LDA) or Latent Semantic Indexing are often found to be helpful in text classification because they reduce the high-dimensional bag-of-words feature space (with $n_{voc} = 32{,}401$ dimensions in our case) to a comparatively small number of *latent semantic* dimensions. Machine learners are expected to perform better because information is packaged more densely into the latent features and a smaller number of parameters needs to be trained. Exploratory tests showed that our ML classifier does not benefit from such dimensionality reduction techniques, though.

We might also use the latent semantic information to pre-classify the disclosures into meaningful categories, viz. the pre-defined set of approximately 40 topics regulating their emission. As a matter of fact, the most prominent topic in our corpus (according to an LDA model) with a mean latency of almost 25%[8], is made up of rather generic lemmas

[6] That is to say: all categories contain equal numbers of disclosures in each fold of the cross-validation.

[7] The trading strategy rests on the assumption that we can buy or sell the shares after the material event and thus indeed collect the net gain of CAR values.

[8] The result of an LDA is a probability vector for each document comprising the probabilities with which each of the topics has contributed to the creation of the document. The "mean latency" of a topic is thus the average probability of that topic across all documents.

such as

> *product, service, technology, lead, position, agreement, new, future, work, solution, system, customer, provide, production, subsidiary, focus, ceo, industry, develop,* and *management.*

The topic above could e.g. be interpreted as *future products and contracts* or alike, yet there are clearly ambiguous and noisy terms such as *ceo*, *industry*, etc., which make the interpretation very speculative. The second most prominent topic with more than 19% mean latency contains the following lemmas:

> *earnings, previous, tax, ebit, figure, quarter, compare, profit, revenue, net, positive, income, first, month, rise, increase, operating, ebitda, result, fiscal.*

This topic points towards quarterly reports. However, neither of the topics point towards a clearly recognizable reason for the emission of a disclosure.[9] Furthermore, the distribution of LDA topics on the particulary interesting subset of disclosures that inform about retirements (see the following section 2.3) is almost identical to their distribution on the full corpus. The topics that can be retrieved from an LDA analysis thus do not help in recognizing particular topics that can be identified manually.

We thus develop a formal ontology to retrieve meaningful semantic features. Since this is expensive with regards to implemenation effort, we concentrate on one frequent and particularly interesting category, namely disclosures concerned with the retirement of key personnel.

NLP pre-processing The first step of operationalizing the corporate texts is a preprocessing stage in which disclosures are analyzed using various off-the-shelf natural language processing techniques, including part-of-speech tagging, morphological analysis, named entity recognition, syntactic parsing, coreference resolution and word sense disambiguation.

The Stanford CoreNLP suite (Manning et al. 2014) offers publicly available tools for the first five tasks. They are part of a pipeline architecture, i.e. every component can access the results of the previous components. For word sense disambiguation, we use the algorithm described in Banerjee and Pedersen (2002) and the sense inventory of the lexical semantic database WordNet (Miller 1995). In the ontological representation, the disambiguated words are mapped to WordNet concepts (*synsets*[10]). The ontology consists of three components:[11]

- A TBox capturing relations among concepts, essentially obtained by extracting relevant information from WordNet for the terms encountered in the text.
- A manually maintained TBox capturing domain-specific background knowledge.

[9] See also Feuerriegel et al. (2015) for a more thorough analysis of the semantic space of ad hoc disclosures.

[10] *Synsets* are sets of synonyms representing a lexical semantic concept or word sense.

[11] *ABox* and *TBox* are the assertion and terminological components of the ontology, respectively.

- An ABox recording the content of the parsed disclosures, generated from the NLP results.

We discuss these parts in more detail below.

Ontology creation from NLP results We first describe the automatically generated parts of the ontology. It has to be emphasized that this ontology is not learned in any sense; rather, the procedure is essentially aimed at transforming linguistically analyzed texts into the Web Ontology Language (OWL), additionally taking into account lexical semantic information from WordNet. WordNet information is provided in terms of a chain of subclass or subproperty inclusions connecting the word form actually appearing in a text to its synset identified by the word sense disambiguation module. E.g. for the word form *leaves* (possibly indicating a retirement event) this takes the following shape (Listing 1):

ObjectProperty: leave

ObjectProperty: leaves
  SubPropertyOf: leave

ObjectProperty: 2383440_Leave_depart_pull_up_stakes
  SubPropertyOf: leave

Listing 1: OWL representation of WordNet information for word form *leaves*.

The last, most specific object property relates to the synset corresponding to the relevant sense of *leaves*. It is composed of the synset's unique WordNet identifier (2383440), followed by the list of all synonyms in the set (to ensure human readability).

As indicated above, the NLP results are transformed into an ABox. The default procedure is to map subjects and objects of sentences, identified by the dependency analysis, to individuals in the ABox, whereas the verbs connecting subjects and objects become object properties. For prepositional objects, the preposition is made part of the object property, which is then named in the form ⟨verb⟩_⟨preposition⟩. If the auxiliary verb *will* is detected in connection with the verb (e.g. if the disclosure states that the CEO *will* resign rather than that he has already resigned), the object property is named announced_⟨X⟩, where X is the original name of the object property, and marked as being a subproperty of both announced and X. Subjects and objects receive as a type the concept generated from their synset according to the word sense disambiguation, and receive as facts their mutual relationship as specified by the synset of the verb. For example, the sentence *John Doe leaves the company* with the syntactic dependency analysis in Figure 2 is translated into the ABox depicted in Listing 2.

Note that each syntactic dependency connects only two words. For compound nouns, the rightmost noun is regarded as the head noun, and the other component nouns are linked to the head noun via a *compound* relation. Compound nouns have to be recomposed from the syntactic dependencies, which results in the individual *John Doe* rather than just *Doe*. Coreferences are resolved while creating the ontology, so facts referring to a pronoun are attached to the corresponding individual.

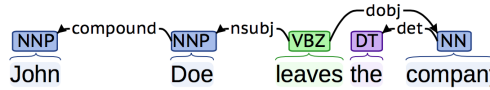

Figure 2: Dependency parse of *John Doe leaves the company.*

Individual: John_Doe
Types: Person
Facts: 2383440_leave_depart_pull_up_stakes company

Individual: company
Types: 8058098_Company

Listing 2: ABox representation of sentence *John Doe leaves the company.*

In this case, the types of the individuals are inferred from named entity recognition (Person) and morphological analysis (Company). Appositions are also used to infer types: the dependent of an apposition determines an additional type for its governor, and triples describing the dependent are assigned to the governor. Prepositional triples are prefixed by the dependent of the apposition. For instance, from the phrase *John Doe, CFO of the company*, one obtains the dependency relations

$$\text{appos}(\textit{Doe}, \textit{CFO}) \quad \text{and} \quad \text{of}(\textit{CFO}, \textit{company}),$$

which extend the knowledge about John Doe in the way depicted in Listing 3.

The previous examples always contained the main piece of information, e.g. on someone leaving a company, or doing something in general, in a subject-predicate-object-like structure. A sentence like "He announced the retirement of John Doe" does not fit into this pattern. Therefore our system uses derivational relations from WordNet to transform triples like *of(retirement, John Doe)* into a subject-predicate-object structure, *retire(John Doe, dummy)*. The dummy individual is needed because the intransitive verb *retire* (from which *retirement* is derived) does not take an object. This type of normalization simplifies querying the assertional knowledge parsed from the text in subsequent steps.

Background knowledge The system is supported by a static, manually maintained background ontology capturing general and domain knowledge that is not explicit in the

Individual: John_Doe
Types: Person, CFO
Facts: 2383440_leave_depart_pull_up_stakes company.
CFO_of company

Listing 3: Extended ABox for sentence *John Doe leaves the company.*

Class: 9916601_chief_financial_officer_cfo
EquivalentTo: works_on **some** Cfo_position
SubClassOf: works_on **exactly** 1 Executive_board_position

Class: Cfo_leave1
EquivalentTo: leave **some** Cfo_position,
Cfo **and** leave **some** Executive_board_position

Class: Cfo_leave2
EquivalentTo: Cfo **and** (leave **some** Executive_board),
leave **some** Cfo_position

Class: leave3
EquivalentTo: (have **some** (Contract **and** expire **some** owl:Thing)),
SubClassOf: leave **some** Position

Class: leave4
EquivalentTo: agree **some** (Termination **and** (of **some** Mandate)),
SubClassOf: leave **some** Position

Class: leave5
EquivalentTo: submit **some** Resignation,
SubClassOf: leave **some** Position

Listing 4: Excerpt from the background ontology.

text of the disclosures. Some of the relevant facts are quite simple, e.g. that stepping down is a form of leaving and that "Executive Board" and "Management Board" are synonyms. Other axioms are more interesting and capture combinations of standard jargon with basic knowledge of the domain. E.g. at the domain-specific level we include axioms saying that CFOs work on exactly one executive board position, and that they retire from their CFO position iff they retire from their executive board position.[12] At a less specific level there are axioms saying that, e.g., letting your contract expire, agreeing to the termination of your mandate, and submitting your resignation all amount to leaving your current position. The formulation of statements such as these is illustrated in Listing 4.

Querying With the ontology in place, we can now detect disclosures concerning retirement of key personnel by querying the ABox generated from the disclosure for persons leaving from something. The formulation of the corresponding query is shown in Listing 5. The filter statements serve to eliminate multiple results that differ only in the value of the *?leave* variable, i.e. use different subproperties of *leave* but refer to the same person and position. That is, the query is set up in such a way as to return only the triples with the most specific object property as the instantiation for *?leave*.

In case the query returns any result, the ad hoc disclosure is marked as containing

[12]The universal validity of these axioms may be debatable but since OWL does not incorporate default reasoning, there appears to be no realistic way to ensure stricter accuracy.

```
SELECT DISTINCT ?person ?leave ?object WHERE{
    ?person ?leave ?object.
    ?person a :Person.
    ?leave rdfs:subPropertyOf :leave.
    FILTER NOT EXISTS{ ?person ?leave2 ?object.
        ?leave2 rdfs:subPropertyOf ?leave.
        FILTER NOT EXISTS{?leave2 owl:equivalentProperty ?leave. }}}
```

Listing 5: The main query used to detect leaving persons.

a message about a retirement. In case the instantiation of *?leave* is a subproperty of *announced*, the disclosure is additionally annotated as being (only) an announcement.

Evaluation of the ontological approach The ontological detection of retirement events and announcements among all ad hoc disclosures was tested on a set of 300 messages containing any inflected form of the words *leave* or *retire*. The disclosures were categorized manually as retirement (178 messages) or non-retirement (122 messages). The low baseline accuracy of 59.3% shows that the mere occurrence of the keywords *leave* and *retire* is not a reliable predictor. Our algorithm obtained recall and precision values of 90.4% and 97% for retirement events, respectively.[13] Regarding the additional property of retirements being only announced rather than already realized, 75 of 139 messages were successfully identified as (only) announcing at least one retirement, and 6 were falsely classified as (mere) announcements (recall 54%, precision 92.5%). It is, of course, not entirely surprising that automated detection of the event ("retirement") as such works better than automated detection of the much more abstract question of its factuality. Subsequent to these tests, we ran the ontological classifier on the whole dataset of 28,287 disclosures, obtaining a set of 1,046 disclosures (ca. 3.7%) classified as retirement events.

2.4 Integration methods

We now turn to equipping the ML classifiers with the ontologically extracted retirement feature in order to improve their performance. Our idea is that the ontological information about the types of material events that regulate the dissemination of the disclosures in the first place can be used for splitting the overall problem into smaller sub-problems: A machine learner trained solely on retirement disclosures is confronted with an easier task than a system that does not have any information about the reasons for the dissemination of the disclosures at hand; just as a human expert confronted with very specific disclosures has an easier task than someone who is confronted with an unstructured bulk of disclosures.

Our first combination of ML and ontology is by means of adding a single "retirement" feature to the document-lemma feature matrix (**FM2**). However, since a single retirement

[13] 161 of the 178 true retirement messages were detected correctly by the algorithm (true positives) while 5 disclosures were incorrectly marked as retirements (false positives).

feature can easily be overseen amongst other features, we experiment with a separation of the vocabulary of the retirement disclosures from the vocabulary of non-retirements: If a disclosure is recognized as a retirement by the ontological model, the string `retirement` is appended to each lemma in the text (**FM3**).

This method has the disadvantage that the ML classifier cannot generalize information about the general meaning of lemmas (e.g. *risk* or *losses*) gathered from the much larger remainder of the corpus to the retirement disclosures. It is likely to underperform in this setting because it is effectively restricted to a small training corpus. We thus consider a third combination method that mirrors the retirement vocabulary (**FM4**): All retirement disclosures now retain the original lemmas, but are complemented with an *additional* retirement vocabulary. In the example disclosure above, lemmas such as $Montabaur_{\text{retirement}}$, $CFO_{\text{retirement}}$, and $company_{\text{retirement}}$ are added without deleting the original lemmas.

To put it in other words, the reasoning behind separate and mirrored vocabularies is as follows: A *single* retirement feature might not be recognized efficiently by a machine learner. A *separate* vocabulary, moreover, discriminates against retirement disclosures, since the machine learner cannot exploit features from the much larger non-retirement part of the training corpus for the retirements; as a result, the amount of training data for lemmas in the retirement vocabulary is drastically reduced. A separate vocabulary is equivalent to two separate machine learners being trained for the retirement disclosures and the non-retirements, respectively. Last but not least, the retirement features are weight *adjustments* in the case of the mirrored vocabulary: Here the machine learner can learn features both on retirements and non-retirements and can then exploit this knowledge for all disclosures. Including the basic feature matrix (**FM1**), there are thus four feature matrices that can be used for prediction (see Table 2 for an overview).

features	*description*	n_{voc}
FM1	vanilla feature matrix without retirement feature	32,401
FM2	FM1 with a single *additional retirement feature*	32,402
FM3	FM1 with a *separate vocabulary* for the retirement disclosures	37,652
FM4	FM1 with a *mirrored vocabulary* for the retirement disclosures	38,762

Table 2: The four different feature matrices used for prediction.

Evaluation of integration methods Since the integration methods essentially differ in feature matrices, they can be compared to the original classifiers (and their respective baselines) in a straightforward manner by means of accuracy in 10-fold cross-validation. Moreover, for retirement disclosures, another baseline is readily at hand: Since the retirement feature is weakly yet significantly associated with the target variable, retirement disclosures can be classified *ontologically*: The greater part of retirement disclosures lead to a negative stock market reaction (cf. Table 1), so that the ontology already outperforms the baseline by assigning the category *negative* to all retirement disclosures.

3 Results and Discussion

There are two kinds of effects to be analyzed: Firstly, the effect of ontological information on the prediction quality of task solvers can be quantified. Secondly, one can observe how the feature weights are affected by the additional information, which gives interesting insights into the different usage of language in particular discourse topic domains.

3.1 Prediction results

The complete results for the four prediction tasks can be found in Tables 3 and 4 for the hard and the easy prediction task, respectively. The left hand tables show performance on the whole corpus, the right hand table considers solely retirement disclosures.[14] One row of a table shows the performance of the different classifiers for given feature matrices. In most cases, accuracy values are higher for MaxEnt than for MNB. Comparing the different prediction tasks with one another, one can unsurprisingly see that performance is higher in the case of clear-cut categories. Generally, the machine learners consistently outperform the random baseline (33.3%) and the ontological baseline (which varies for each split into training and test corpus since we do not stratify the number of retirement disclosures in the cross-validation).

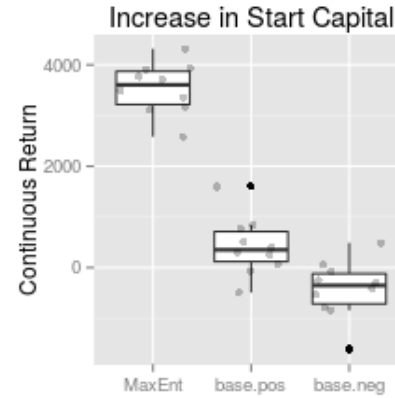

Figure 3: Profit made by simple trading strategy based on classification of disclosures in the easy prediction task (task solver: MaxEnt using FM1).

For instance, MaxEnt with feature set FM1 significantly outperforms the 33.3% baseline in the easy prediction task with an accuracy of **51.9%** (± **1.9%**). Moreover, Figure 3 shows substantial net gain when using MaxEnt with FM1 and applying the trading strategy outlined in section 2.2: Compared to the baseline traders which always opt for *sell* (base.neg) and *buy* (base.pos), respectively, the accumulated continuous returns of the task solver result in a mean profit of more than 0.2% per disclosure and a considerable increase of our start capital despite the simple approach.

More interestingly, one column of a table allows for comparison between the different feature matrices. FM4 (mirrored vocabulary) consistently outperforms all the other feature matrices. The effect is higher on the sub-corpus of retirements for the reasons elaborated above. In the case of the retirement disclosures, it is worth mentioning that the ontological baseline (predicting class *negative* for all retirement disclosures, given at the bottom lines of the right hand panels of the performance tables 3 and 4) presents

[14] Since less than 4% of the disclosures deal with key personnel turnover in all prediction tasks, "zooming" onto the subcorpus of retirement disclosures is more likely to reveal the effect of including ontological features.

difficult	*full*		*retirements*	
	MNB	MaxEnt	MNB	MaxEnt
FM1	.437 (±.016)	.456 (±.018)	.395 (±.101)	.426 (±.092)
FM2	.437 (±.016)	.456 (±.016)	.395 (±.101)	.426 (±.092)
FM3	.437 (±.015)	.456 (±.019)	.429 (±.124)	.414 (±.091)
FM4	.439 (±.014)	**.459** (±.025)	.445 (±.106)	**.450** (±.128)
baseline	1/3 = .333		.396 (±.086)	

Table 3: Performance (mean accuracy and 95% confidence interval) of the task solvers in the difficult prediction task on the whole corpus (left panel) and on the subcorpus of retirement disclosures (right panel), using different feature matrices (FM1–4). The naïve baseline (majority classifier) is given by 1/3 = 33.3%, an improved (ontological) baseline is given for the retirement corpus.

easy	*full*		*retirements*	
	MNB	MaxEnt	MNB	MaxEnt
FM1	.485 (±.024)	**.519** (±.019)	.467 (±.126)	.479 (±.115)
FM2	.485 (±.024)	.518 (±.014)	.467 (±.123)	.481 (±.117)
FM3	.482 (±.022)	**.519** (±.021)	.431 (±.090)	.470 (±.098)
FM4	.486 (±.022)	**.519** (±.018)	.477 (±.111)	**.500** (±.092)
baseline	1/3 = .333		.419 (±.087)	

Table 4: Performance of the task solvers in the easier prediction task (with well-defined categories). Performance increases consistently as in the difficult prediction task when mirroring the vocabulary (FM4). A separate vocabulary (FM3) decreases performance, a single retirment feature (FM2) does not change performance at all.

itself as a strong competitor for the machine learners. Statistical learning is, however, in almost all cases better than this ontological baseline.

3.2 Feature weight analysis

We identified lemmas whose feature weights are substantially different in retirement disclosures than in non-retirements (which can be seen from FM3), or which obtain a relatively high "adjustment" weight in the mirrored retirement vocabulary (in the best-performing feature set FM4). Table 5 shows such lemmas based on feature weights for the category *positive*. Results for category *negative* are omitted since they show similar patterns.

For example, the lemmas *exceed* (1.293 for category *positive* in FM1) and *improvement* (0.708 in FM1) are generally associated with a positive CAR response. However, FM4 reduces these weights in retirement disclosures by −0.019 for *exceed* and −0.014 for *improvement*, showing that they imply a different outcome for this event type (see

section 2.4 for an explanation of weight adjustments).[15] The relatively small adjustement is probably due to the low overall proportion of retirements, and the effect becomes much clearer with a separate vocabulary in FM3: the feature weights on retirement disclosures are −0.021 (*exceed*) and −0.018 (*improvement*), respectively, showing that these lemmas no longer indicate a positive stock market reaction. Similarly, the lemma *insolvency* is generally associated with a negative reaction, but indicates a positive response when used in retirement disclosures.

	FM1	**FM3**		**FM4**	
lemma		non-ret.	ret.	non-ret.	ret.
exceed	1.293	1.293	-0.021	1.293	-0.019
fall	-0.864	-0.842	-0.034	-0.855	-0.027
career	0.090	-0.033	0.115	0.044	0.089
improvement	0.708	0.696	-0.018	0.700	-0.014
rise	0.612	0.616	-0.024	0.614	-0.023
weak	-0.769	-0.766	-0.012	-0.769	-0.009
lower	-1.022	-1.012	-0.041	-1.018	-0.028
positive	1.149	1.130	-0.007	1.137	-0.015
insolvency	-0.386	-0.447	0.081	-0.417	0.059

Table 5: Lemmas whose feature weights for category *positive* are substantially different in retirement disclosures (additional feature weights in FM3 and FM4) from their overall feature weights (FM1).

4 Conclusion

Machine learners are used in many prediction tasks of computational linguistics. We have combined a semantics-based approach to recognition of message content with a machine-learning classification of documents, specifically of corporate disclosures according to their effect on the stock price.

Machine learners benefit from ontological information since they can thus deal with a more specific realm of language use. The core idea tested in our feasibility study is that words are used more consistently within the specific domain of retirement disclosures. The effect on prediction accuracy is small, yet consistent. Testing for statistical significance by use of a McNemar test shows that some of the improvements are indeed significant.

Future work will be aimed partly at refining the ontological approach to improve its precision and recall (both already above 90% on the main target feature, retirements, but

[15] FM2 is omitted here because its lemma feature weights are almost identical to FM1. Recall that FM2 just adds a single feature indicating retirement disclosures (with a correct indication of category *negative*), so it cannot account for the differences in language use between retirements and other messages that we are interested in here.

not achieving comparable performance for more fine-grained information such as detecting the position that the retiree steps down from). On the other hand, the results of our study make it seem likely that broadening the ontological model to recognize additional features (e.g. patents granted, loans taken up) will further improve the prediction accuracy for the eventual target, the effect of a disclosure on the stock price. Last but not least, we will strive to develop new methods for exploiting the subjective use of language in different domains in order to improve prediction accuracy.

References

Satanjeev Banerjee and Ted Pedersen. An adapted Lesk algorithm for word sense disambiguation using WordNet. In *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '02, pages 136–145, London, UK, 2002. Springer-Verlag.

Johann Bollen, Huina Mao, and Xiao-Jun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, October 2010.

C. Corrado. Event studies: A methodology review. *Accounting and Finance*, 51:207–234, 2011.

Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Deep learning for event-driven stock prediction. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (ICJAI)*, pages 2327–2333, 2015.

Federal Financial Supervisory Authority (BaFin). Issuer guidelines, 2009.

Stefan Feuerriegel, Antal Ratku, and Dirk Neumann. Which News Disclosures Matter? News Reception Compared Across Topics Extracted from the Latent Dirichlet Allocation. *News Reception Compared Across Topics Extracted from the Latent Dirichlet Allocation (February 13, 2015)*, 2015.

N. Jegadeesh and D. Wu. Word power: A new approach for content analysis. *J. Financial Economics*, 110:712–729, 2013.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014.

A. McWilliams and D Siegel. Event studies in management research: Theoretical and empirical issues. *Academy of Management J.*, 40:626–657, 1997.

George A Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39, 1995.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, October 2011.

N. Strong. Modelling abnormal returns: A review article. *J. Business Finance & Accounting*, 19:533–553, 1992.

Thomas Verchow. *Ad-hoc-Publizität und Kapitalmarkteffizienz: Eine Untersuchung basierend auf der Textanalyse von Ad-hoc-Mitteilungen.* PhD thesis, Ulm University, Faculty of Mathematics and Economics, 2011.