Automatic Detection of Uncertain Statements in the Financial Domain

Christoph Kilian Theil, Sanja Štajner, Heiner Stuckenschmidt, and Simone Paolo Ponzetto

Data and Web Science Group
University of Mannheim, Germany
{christoph,sanja,heiner,simone}@informatik.uni-mannheim.de

Abstract. The automatic detection of uncertain statements can benefit NLP tasks such as deception detection and information extraction. Furthermore, it can enable new analyses in social sciences such as business where the quantification of uncertainty or risk plays a significant role. We approached the automatic detection of uncertain statements as a binary sentence classification task on the transcripts of spoken language for the first time in the financial domain. We created a new dataset and – besides using bag-of-words, part-of-speech tags, and dictionaries – developed rule-based features tailored to our task. Finally, we analyzed systematically, which features perform best in the financial domain as opposed to the previously researched encyclopedic domain.

Keywords: Automatic uncertainty detection · Binary sentence classification · Financial domain

1 Introduction

In linguistics, the use of uncertain statements is described by the phenomenon of "hedging" which is defined as "any linguistic means used to indicate either a) a lack of complete commitment to the truth value of an accompanying proposition, or b) a desire not to express that commitment categorically" [1, p. 1]. As can be seen, this definition is centered on a speaker or writer. For the scope of this paper – an application of uncertainty detection in social sciences, and more specifically, the financial domain – we adjust this definition slightly. As we bear in mind to predict market reactions in future work, we establish a definition of uncertainty which also keeps in mind the recipient's side of the communication process.

1.1 Uncertainty as Opposed to Linguistic Hedging

In addition to the sentences encompassed by the aforementioned linguistic definition, we also classify sentences as uncertain:

- If their truth value cannot be determined (e.g. statements about the future)
- If they refer to uncertain factors (e.g. statements about market volatility)

– If they show uninformedness (e.g. statements conveying lack of knowledge)

How uncertainty could be further broken down into more granular categories will be introduced in Section 4.2.

1.2 Opportunities of Automatic Uncertainty Detection

Automatic detection of uncertain statements can benefit NLP tasks such as deception detection [2, 3], information extraction [4, 5], and summarization [6]. Furthermore, automatic uncertainty detection can enable new analyses in social sciences where the quantification of uncertainty or risk plays a significant role. Disciplines like business and economics would profit from an automatically extractable measure of uncertainty which does not depend on manual analysis.

As of now, automatic uncertainty detection has been limited to detecting hedges (as opposed to our broader concept of uncertainty) in biomedical scientific texts and Wikipedia articles. The results of the CoNNL-2010 shared task [7, pp. 6–8] indicate that this task is easier to solve for the former than for the latter.

Loughran & McDonald specifically proposed to investigate "whether or not managers using high levels of uncertain or weak modal [...] words during conference calls experience worse subsequent stock or operating performance" [8, p. 43]. Within this paper, we address the first part of this suggestion by providing a classifier of uncertainty suited to analyze earnings calls.[1]

1.3 Contributions

We performed the classification of spoken uncertain statements for the first time in the financial domain. For this purpose, we gathered and annotated a new financial domain dataset. Adapting the definition of "linguistic hedging", we developed a new concept of uncertainty which fits the domain-specific needs. In contrast to previous work, our concept of uncertainty encompasses how the use of uncertain statements can have an impact on other social agents and thus enables predictions of market reactions.

To achieve our goal of comparing the automatic uncertainty detection in the financial to the encyclopedic domain, we pose four research questions (RQ 1–4):

- **RQ 1**: How do all of our feature sets separately perform on both our new financial and the existing Wikipedia datasets?
- **RQ 2**: How do lexical and syntactic features perform on both the financial and the encyclopedic domain?
- **RQ 3**: How do knowledge-poor and knowledge-rich features perform on both the financial and the encyclopedic domain?
- **RQ 4**: How do our new, domain-specific rules contribute to the classification for the financial domain? Are they applicable to the encyclopedic domain?

[1] Earnings calls are publicly accessible teleconferences or webcasts in which executives of a public company present the financial results of the last quarter.

2 Related Work

The NLP task of detecting uncertain statements has already been addressed in the biomedical and the encyclopedic domain – for example in the CoNLL-2010 shared task [7]. However, the topic has not been explored in the social sciences, let alone finance. Hence, within this section, we give a short overview about the existing approaches of uncertainty detection in NLP and then cover the closest related applications in the financial domain.

2.1 Approaches of Uncertainty Detection in NLP

The first to perform uncertainty detection in the biomedical domain where Light *et al.*, which have shown that a substring matching approach (with 14 manually selected hedge cues) slightly outperforms an SVM classifier with bag-of-word (BoW) vectors in terms of accuracy (95% vs. 92%) [9, p. 22]. Following up, Medlock & Briscoe presented a weakly supervised machine learning approach which outperformed Light *et al.*'s best classifier (76% vs. 60% accuracy) [5, p. 998]. Yielding better results than both of these, Szarvas presented a maximum entropy classifier achieving an F_1 score of 0.85 [4, p. 287]. He showed that a classifier using only unigrams instead of bi- and trigrams performs significantly worse ($F_1 = 0.80$) [4, p. 286]. Furthermore, he proved that even only a slight out-of-domain application (bioinformatics articles instead of biomedical papers) yields a high drop in performance ($F_1 = 0.75$) [4, p. 286].

The first classifier of Wikipedia sentences was presented by Ganter & Strube and made use of both corpus statistics and syntactic patterns [10, p. 173] . Subsequently, uncertainty detection both within the biomedical and the encyclopedic domain has been resumed in the CoNLL-2010 shared task, where an extensive array of features was used: dictionaries, orthographic token information, lemmas/stems, part-of-speech (POS) tags, syntactic chunk information, dependency parsing, and the position of the token within the document were addressed with sequence labeling (SL), token classification and BoW approaches [7, p. 9]. While the best performing system for the biomedical domain made use of an SL approach, the best classifier for the encyclopedic domain used a BoW approach with a dictionary [7, p. 8].

These insights motivated our proposed feature sets. We evaluated the features that performed best for the Wikipedia dataset (BoW vectors in combination with a dictionary-based approach) on our new financial domain dataset. Furthermore, we enriched the BoW vectors with POS tags and explored the possibilities of applying hand-written syntactic rules. A systematic overview of all features is provided in the beginning of Section 4.

2.2 NLP in the Financial Domain

While already a few surveys provide a literature overview of NLP in finance [11, 12, 13], the most recent one was presented by Loughran & McDonald [8].

Most applications of NLP in finance focus on formal disclosures such as 10-Ks or 10-Qs[2] as opposed to earnings calls, e.g. Li [14, 15] or Loughran & McDonald [16, 17, 18]. Larcker & Zakolyukina [2] summarize the limitations of the former textual forms such as a relative uniformity of the content over time and little spontaneity [2, p. 499]. Hence, we are further motivated to investigate earnings calls instead of formal disclosures.

Loughran & McDonald [17] extracted their own dictionary of uncertainty triggers from a sample of ∼50,000 10-Ks [17]. As using a dictionary has shown to yield good results for classifying the uncertainty of Wikipedia sentences [7, p. 9], we use this domain-specific uncertainty dictionary for our experiments.

3 New Uncertainty Dataset

An earnings call consists of one or more executives (e.g. the CEO or CFO) presenting the company's financial results of the ending quarter to the public via a teleconference and/or a webcast. In a second section, the call is opened for a question-and-answer-session (Q&A) with investors and banking analysts. In addition to the mentioned protagonists, an operator takes care of technical requirements such as opening and ending the call or moderating the Q&A.

Since the first part of the call closely follows the accompanying press release, it is highly formalized and provides little opportunity for the executives to speak freely. Hence, our analyses solely focus on the second part of the call, the Q&A session. Moreover, as we are interested in obtaining data that might help to gain insight about the company's financial (un)certainty itself, we only include sentences uttered by the executives (the answers), instead of the analyst's questions or the operator's technical remarks.

As we analyze free speech instead of written, formalized text, we expect our problem to be more challenging to solve than e.g. classifying biomedical or encyclopedic sentences. Consider, for example, the following statement:

Example 1. "And increasingly look as you are sort of describe us [sic] as well, we look to focus where we can really make a difference [...]"

In spite containing the hedge "sort of", this sentence was annotated as *certain*, according to our methodology. In this case, the adverb of degree "sort of" is used in a colloquial sense as inherent in any free speech. The first part "as you [...] sort of describe us" is highly unspeculative and easily verifiable/falsifiable by a potential listener. In contrast, consider the following example:

Example 2. "Now, what we don't know is what's going to happen at the end of the third quarter."

While this sentence does not contain any hedges such as adverbials of degree or of possibility, it indicates a lack of knowledge of the speaker, which is why

[2] 10-Ks/10-Qs are standardized annual/quarterly reports providing an overview of a company's financial results, which are required by the U. S. Securities and Exchange Commission.

we annotated it as *uncertain*. These two examples might give an idea why the task of detecting uncertain statements in spoken language within the financial domain is of particular complexity.

As basis for the dataset, we used the Standard & Poor's 500 Index (S&P 500)[3] as one of the most important equity indices. Since the webpage Seeking Alpha supplies a large database of publicly available earnings call transcripts,[4] we obtained all data from there. For our dataset, we took a total of 7,725 transcripts from 217 different S&P 500 companies belonging to a wide array of industries such as Financials, Industrials or Information Technology.

Out of the dataset, we randomly sampled 1,800 sentences and annotated them to either be *certain* or *uncertain*. Habitual utterances such as greetings, expressions of thanks, farewells etc. were excluded from the sampling process, as they would dilute the results of our task. Out of these 1,800, 100 sentences were randomly selected and independently annotated by a second annotator of financial background. As the inter-annotator agreement measured as Cohen's kappa (κ) [19] was 0.81, which – depending on the source – can be considered as "almost perfect" [20, p. 165] or "excellent" [21, p. 218], the rest of the annotation was carried out only by the first annotator, which is of linguistic background. Afterwards, we split the set in two: 800 sentences (683 *certain*, 117 *uncertain*) were taken to develop the syntactic rules (see Section 4.2), while the remaining 1,000 (829 *certain*, 171 *uncertain*) were used for the classification experiments.

4 Methodology

We addressed the problem of automatic uncertainty detection as a binary sentence classification task. As BoW vectors, POS tags, and a comprehensive list of uncertainty cues have been used before [7], we explored the possibility of adding novel features. Due to its domain-specificity, we used Loughran & McDonald's uncertainty dictionary [17]. In addition, we applied a set of hand-written rules specifically designed for our task. All features can be classified along lexical vs. syntactic and knowledge-poor vs. knowledge-rich dimensions, which yields a feature set matrix as depicted in Figure 1.

We lemmatized the BoW vectors with NLTK's WordNet implementation [22] and normalized them via tf-idf weighting. Additionally, we extracted POS tags with NLTK 3.2.1's standard POS tagger, which is based on Honnibal's implementation of the Averaged Perceptron tagger [23, 24].[5] The following Sections 4.1 and 4.2 further elaborate on the features unique to our approach.

4.1 Lists of Speculation Triggers

Within the experiments, we used the following lists of speculation triggers:

[3] `http://us.spindices.com/indices/equity/sp-500`

[4] `http://seekingalpha.com/earnings/earnings-call-transcripts`

[5] This tagger reached an accuracy of 96.80% when applied to an evaluation set of 130,000 words taken from The Wall Street Journal [25].

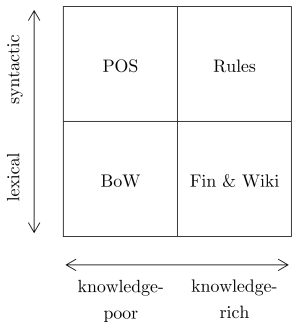

Fig. 1. Feature set matrix.

- **Fin**: Loughran & McDonald's list of 297 unigrams indicating uncertainty in the financial domain (e.g. "fluctuation", "recalculation") based on a sample of ∼50,000 10-Ks.[6] After lemmatization, the list totaled 192 items.
- **Wiki**: 1,984 uncertainty triggers of arbitrary length (e.g. "a matter in dispute", "some prehistoric cultures") were extracted from the CoNLL-2010 shared task's Wikipedia training set.[7] After lemmatization, the list totaled 1,868 unique items.

4.2 Rules

We developed a set of 95 hand-written rules according to which a sentence can be classified as *uncertain* based on 800 randomly selected sentences. A rule is always characterized by syntactic criteria (POS tags, phrase chunks), which can additionally be refined by lexical features (lemmas, word lists). The word lists define more granular word classes such as *adverbs of degree* (e.g. "kind of", "quite"), *adverbs of probability* (e.g. "potentially", "probably"), *fuzzy quantifiers* (e.g. "about half of", "close to 100"), and *verbs of expectation* (e.g. "anticipate", "expect"). All rules can be assigned to seven different categories which are defined as presented in Table 1. According to our methodology, *Example 2* presented in Section 3 is captured by the rules category "Uninformedness".

In addition, we applied the rules to 30 random samples of 1,000 Wikipedia test sentences to check their applicability for a general domain as opposed to the domain-specific context of our new dataset. To guarantee the greatest possible comparability, for each of the samples, the class-distribution of 829 *certain* to 171 *uncertain* sentences of our dataset was maintained. The results of this experiment are shown in Table 2.

As expected, the rules match substantially fewer sentences (9.70 on average) in the Wikipedia test set than in the financial domain dataset (54 matches).

[6] `http://www3.nd.edu/~mcdonald/Word_Lists.html`
[7] `http://rgai.inf.u-szeged.hu/conll2010st/download.html`

Table 1. Categorization of the rules.

Category	Count	Example
Expectation	29	"I expect our maintenance capital [...] to probably be"
Assumption	25	"I think it's pretty mature"
Probability	12	"perhaps by the end of this year"
Uninformedness	10	"we really don't know what ultimately it's going to sell"
Subjunction	9	"it might be a few hundred thousand dollars"
Volatility	6	"the volatility of where we are"
Unspecifity	4	"somewhere in the 40% range"

Table 2. Descriptive statistics for the number of times the rules match 30 random samples of the Wikipedia test set.

n	Min	Max	Mean	Median	Mode	SD	SK
30	3.00	15.00	9.70	9.50	8.00	2.76	0.04

4.3 Experiments

For each sentence in the dataset, we defined a vector containing each feature's occurrences. Afterwards, we applied seven machine learning algorithms in WEKA experimenter [26] using a 10-fold cross-validation setup with 10 repetitions: Logistic Regression [27], Naïve Bayes [28], Support Vector Machines (SVM) [29], k-Nearest Neighbors [30], JRip [31], C4.5 [32], and Random Forest [33]. We evaluated the performance for all eleven feature sets used in the subsequent experiments and compared the weighted average F_1 scores. Since SVM achieved the best results in all cases, we used this algorithm for the subsequent experiments.

Addressing our research questions (see Section 1.3), we carried out four sets of experiments as shown in Table 3. We applied the SVM algorithm to both our dataset and the Wikipedia test set with different feature set combinations across the matrix presented in Figure 1. Thus, we evaluated the performance of our domain-specific classifier on the general domain. To ensure comparability of the data, we used the 30 random samples of the Wikipedia test set as shown in Table 2 and calculated the means of the respective performance measures.

As Farkas *et al.* have summarized [7], pure BoW vectors have proven to be a strong feature set in the encyclopedic domain, which is why we used it, too, and additionally contrasted it to POS-enriched BoW vectors ("POSBoW"). Apart from all individual features (RQ 1, see 5.1), we were interested in how the dimensions lexical vs. syntactic (RQ 2, see 5.2) and knowledge-poor vs. knowledge-rich (RQ 3, see 5.3) would compare. Lastly, we investigated how the rules benefit the overall performance of the classification task (RQ 4, see 5.4).

5 Results and Discussion

In this section, we present the results for each set of experiments. We conducted corrected paired t-tests with $\alpha = 0.05$ to check for significant differences in

Table 3. Sets of experiments.

Set	Description	Feature Sets
1	Separate features (RQ 1)	BoW, POSBoW, Fin(+Wiki), Wiki, Rules
2	Lexical vs. syntactic (RQ 2)	BoW+Fin+Wiki, POSBoW+Rules
3	Knowledge-poor vs. -rich (RQ 3)	POSBoW, Fin+Wiki+Rules
4	Contribution of the rules (RQ 4)	Rules, POSBoW+Fin+Wiki(+Rules)

classification performance. The performance was evaluated in terms of precision (P), recall (R), and F_1 score (F).

5.1 Separate Features (RQ 1)

With P = 0.77, the rules significantly outperform the other individual features of the *uncertain* class (see Table 4). As expected, this comes at the cost of a relatively low recall of 0.13. BoW reaches a recall significantly higher (0.37) than all features apart from POSBoW (0.35). The latter receives the highest F_1 score (0.41) which is insignificantly higher than the former's (0.40), yet significantly higher than the value of all other feature sets.

For the encyclopedic domain (see Table 5), the Wiki dictionary outperforms all other features. This is not surprising, as it was designed specifically for this domain. The domain-specificity can also explain why the rules prove to be the weakest feature set by far. As previously shown in Table 2, they rarely match any of the Wiki sentences which is reflected in a generally poor performance.

5.2 Lexical vs. Syntactic Features (RQ 2)

For the new dataset (see Table 4), syntactic features prove to perform only slightly better than lexical ones with no significant improvement across all performance measures. For the Wikipedia test set (see Table 5), in contrast, the lexical features perform noticeably better, especially in terms of recall (0.39 vs. 0.31) and F_1 score (0.47 vs. 0.41) of the *uncertain* class. This can probably be attributed to the high performance of the Wiki dictionary when treated separately.

As Wikipedia attempts to provide an unbiased source of encyclopedic knowledge, the sentence structure is highly formalized. Hence, rule-based and other features leaning more towards the syntactic side are likely to have little applicability. Instead, lexical choices seem to reflect degrees of uncertainty better in this case. Since the opposite case holds for our dataset (i.e. the sentence structure is relatively free and spontaneous), the results are in line with our expectations.

5.3 Knowledge-Poor vs. Knowledge-Rich Features (RQ 3)

On both datasets, the knowledge-rich approaches perform slightly better than the knowledge-poor ones. However, the difference is again more noticeable for

Table 4. Results of the classification task on our financial domain dataset (the best results are presented in bold).

Features	Uncertain			Certain			Accuracy
	P	R	F	P	R	F	
RQ 1: Separate Features							
BoW	0.46	**0.37**	0.40	0.87	0.91	0.89	81.54%
POSBoW	0.53	0.35	**0.41**	**0.88**	0.93	0.90	83.18%
Fin	0.53	0.14	0.21	0.85	0.97	**0.91**	82.90%
Wiki	0.48	0.17	0.23	0.84	0.97	**0.91**	82.68%
Fin+Wiki	0.53	0.26	0.34	0.87	0.95	**0.91**	83.09%
Rules	**0.77**	0.13	0.21	0.84	**1.00**	**0.91**	**84.58%**
RQ 2: Lexical vs. Syntactic							
BoW+Fin+Wiki	0.53	**0.39**	**0.44**	**0.88**	0.92	0.90	83.34%
POSBoW+Rules	**0.56**	0.37	0.43	0.87	**0.94**	**0.91**	**84.01%**
RQ 3: Knowledge-Poor vs. -Rich							
POSBoW	0.53	**0.35**	**0.41**	**0.88**	0.93	0.90	83.18%
Fin+Wiki+Rules	**0.58**	0.24	0.32	0.86	**0.96**	**0.91**	**83.76%**
RQ 4: Contribution of the Rules							
Rules	**0.77**	0.13	0.21	0.84	**1.00**	0.91	84.58%
POSBoW+Fin+Wiki	0.57	**0.40**	0.46	**0.88**	0.93	0.91	84.25%
POSBoW+Fin+Wiki+Rules	0.59	**0.40**	**0.47**	**0.88**	0.94	**0.92**	**84.71%**
Majority Class (certain)	0.00	0.00	0.00	0.83	1.00	0.90	82.90%

Table 5. Results of the classification task on 30 random samples of the Wikipedia test set (the results are averages and the best results are presented in bold).

Features	Uncertain			Certain			Accuracy
	P	R	F	P	R	F	
RQ 1: Separate Features							
BoW	0.59	0.34	0.42	0.87	0.95	0.91	84.59%
POSBoW	0.63	0.31	0.41	0.87	0.96	0.91	85.00%
Fin	0.41	0.05	0.09	0.83	0.99	0.90	82.97%
Wiki	**0.66**	0.40	**0.49**	**0.89**	0.96	**0.92**	**86.16%**
Fin+Wiki	**0.66**	**0.41**	**0.49**	**0.89**	0.95	**0.92**	86.10%
Rules	0.13	0.01	0.02	0.83	**1.00**	0.91	82.89%
RQ 2: Lexical vs. Syntactic							
BoW+Fin+Wiki	**0.63**	**0.39**	**0.47**	**0.88**	0.95	**0.92**	**85.46%**
POSBoW+Rules	**0.63**	0.31	0.41	0.87	**0.96**	0.91	85.01%
RQ 3: Knowledge-Poor vs. -Rich							
POSBoW	0.63	0.31	0.41	0.87	**0.96**	0.91	85.00%
Fin+Wiki+Rules	**0.65**	**0.41**	**0.49**	**0.89**	0.95	**0.92**	**86.04%**
RQ 4: Contribution of the Rules							
Rules	0.13	0.01	0.02	0.83	**1.00**	0.91	82.89%
POSBoW+Fin+Wiki	**0.66**	0.37	**0.47**	**0.88**	0.96	**0.92**	**85.83%**
POSBoW+Fin+Wiki+Rules	**0.66**	**0.38**	**0.47**	**0.88**	0.96	**0.92**	**85.83%**
Majority Class (certain)	0.00	0.00	0.00	0.83	1.00	0.90	82.90%

the encyclopedic domain. For the financial domain dataset (see Table 4), the relatively high precision of the knowledge-rich (0.58) compared to the knowledge-poor features (0.53) comes at the cost of a significantly lower recall (0.24 vs. 0.35) and an insignificantly lower F_1 score (0.32 vs. 0.41). For the Wikipedia test set (see Table 5), a slightly higher precision of the knowledge-rich features (0.65 vs. 0.63) is accompanied by a distinctively higher recall (0.41 vs. 0.31) and F_1 score (0.49 vs. 0.41).

We argue, again, that the comparably good performance in case of the Wikipedia test set can be attributed to the Wiki dictionary. It is not only specific to domain and written (instead of spoken) language but also considerably larger than our tailored set of rules (1,984 vs. 95 features), which results in a relatively high recall.

5.4 Contribution of the Rules (RQ 4)

The rules' relatively low recall of 0.13 gets outperformed by a combination of all knowledge-poor features (POSBoW+Fin+Wiki) yielding a recall of 0.40. Combining both yields the strongest feature set in terms of F_1 score of the *uncertain* class (0.47). However, this improvement in performance is rather small, with the slight increase of F_1 score and accuracy being only insignificant.

As proven in Sections 4.2 and 5.1, the rules are not applicable to the Wikipedia test set. This is also why – when being added to POSBoW+Fin+Wiki – they do not yield a noticeable performance change.

6 Conclusions

In this paper, we addressed the automatic detection of uncertain statements as a binary sentence classification task on the transcripts of spoken language in the financial domain. We presented a newly annotated dataset and introduced rule-based features specific to our task. Furthermore, we have proven that the SVM algorithm with a combination of BoW, POS, a general-domain as well as a domain-specific dictionary, and our handcrafted rules performs best.

We have shown that a rule-based approach is not applicable to the general encyclopedic domain. What is more, the domain-specific rules neither increase the classification performance of our in-domain dataset noticeably. Hence, we argue that the efforts of future research should focus on developing an in-domain dictionary – possibly enriched with POS tags. This recommendation is in line with the relatively high performance of the POSBoW feature set. Added value compared to Louhgran & McDonald's dictionary of uncertainty [17] could be generated by also incorporating n-grams with $n > 1$. This idea is partly motivated by Szarvas, who found that his biomedical classifier's performance dropped significantly when using only unigrams [4, p. 286].

In the future, classification performance could be additionally improved by optimizing the parameters of the SVM – indeed, the best performing approach

applied to the CoNLL-2010's Wikipedia test set did precisely this [7, p. 9]. Moreover, the high dimensionality of feature set combinations such as the ones discussed in Section 5.4 indicates that a feature selection could decrease the risk of overfitting.

Given our concept of uncertainty, incorporating real-world knowledge into the classifier might also prove as another fruitful avenue of research. Lastly, the classifier could be applied to a larger-scale set of unseen data thus enabling the prediction of market dynamics such as stock performance.

Acknowledgments

We thank Alexander Diete for his help with the data acquisition and technical advice as well as Clemens Müller for his help with the annotation. This work was supported by the SFB 884 on the Political Economy of Reforms at the University of Mannheim (project C4), funded by the German Research Foundation (DFG).

References

1. Hyland, K.: Hedging in Scientific Research Articles. John Benjamins, Amsterdam/Philadelphia (1998)
2. Larcker, D.F., Zakolyukina, A.: Detecting deceptive disucssions in conference calls. Journal of Accounting Research **50** (2012) 494–540
3. Bachenko, J., Fitzpatrick, E., Schonwetter, M.: Verification and implementation of language-based deception indicators in civil and criminal narratives. In: Proceedings of the 22nd International Conference on Computational Linguistics, Manchester (2008) 25–32
4. Szarvas, G.: Hedge classification in biomedical texts with a weakly supervised selection of keywords. In: Proceedings of ACL-08: HLT, Columbus, OH (2008) 281–289
5. Medlock, B., Briscoe, T.: Weakly supervised learning for hedge classification in scientific literature. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague (2007) 992–999
6. Riloff, E., Wiebe, J., Wilson, T.: Learning subjective nouns using extraction pattern bootstrapping. In: Proceedings of the Seventh Conference on Natural Language Learning, Edmonton (2003) 25–32
7. Farkas, R., Vincze, V., Móra, G., Csirik, J., Szarvas, G.: The CoNLL-2010 shared task: Learning to detect hedges and their scope in natural language text. In: Proceedings of the Fourteenth Conference on Computational Natural Language Learning: Shared Task, Uppsala (2010) 1–12
8. Loughran, T., McDonald, B.: Textual analysis in accounting and finance: A survey. Journal of Accounting Research **54** (2016) 1187–1230
9. Light, M., Qiu, X.Y., Srinivasan, P.: The language of bioscience: Facts, speculations, and statements in between. In: HLT-NAACL 2004 Workshop: BioLINK 2004, Linking Biological Literature, Ontologies and Databases, Boston, MA (2004) 17–24
10. Ganter, V., Strube, M.: Finding hedges by chasing weasels: Hedge detection using wikipedia tags and shallow linguistic features. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Singapore (2009) 173–176

11. Li, F.: Textual analysis of corporate disclosures: A survey of the literature. Journal of Accounting Literature **29** (2010) 143–165
12. Kearney, C., Liu, S.: Textual sentiment in finance: A survey of methods and models. International Review of Financial Analysis **33** (2014) 171–185
13. Das, S.R.: Text and context: Language analytics in finance. Foundations and Trends in Finance **8** (2014) 144–261
14. Li, F.: Annual report readability, current earnings, and earnings persistence. Journal of Accounting and Economics **45** (2008) 221–247
15. Li, F.: The information content of forward-looking statements in corporate filings: A naïve bayesian machine learning approach. Journal of Accounting Research **50** (2012) 494–540
16. Loughran, T., McDonald, B., Yun, H.: A wolf in sheeps clothing: The use of ethics-related terms in 10-K reports. The Journal of Business Ethics **89** (2009) 39–49
17. Loughran, T., McDonald, B.: When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. The Journal of Finance **66** (2011) 35–65
18. Loughran, T., McDonald, B.: Measuring readability in financial disclosures. The Journal of Finance **69** (2014) 1643–1671
19. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement **20** (1960) 41–48
20. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics **33** (1977) 159–174
21. Fleiss, J.L.: Statistical Methods for Rates and Proportions. 2 edn. John Wiley, New York (1981)
22. Bird, S., Loper, E., Klein, E.: Natural Language Processing with Python. O'Reilly Media, Sebastopol, CA (2009)
23. Bird, S., Loper, E.: Natural language toolkit: Taggers. `https://github.com/nltk/nltk/blob/develop/nltk/tag/__init__.py` (2017) Accessed on January 27, 2017.
24. Honnibal, M.: Averaged perceptron tagger. `https://github.com/nltk/nltk/blob/develop/nltk/tag/perceptron.py` (2013) Accessed on January 27, 2017.
25. Honnibal, M.: A good part-of-speech tagger in about 200 lines of python. `https://explosion.ai/blog/part-of-speech-pos-tagger-in-python` (2013) Accessed on January 27, 2017.
26. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: An update. SIGKDD Explorations Newsletter **11** (2009) 10–18
27. le Cessie, S., van Houwelingen, J.C.: Ridge estimators in logistic regression. Applied Statistics **41** (1992) 191–201
28. John, G.H., Langley, P.: Estimating continuous distributions in bayesian classifiers. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. (1995) 338–345
29. Platt, J.C.: Fast training of support vector machines using sequential minimal optimization. In: Advances in Kernel Methods – Support Vector Learning. (1999)
30. Aha, D.W., Kibler, D., Albert, M.K.: Instance-based learning algorithms. Machine Learning **6** (1991) 37–66
31. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning. (1995) 115–123
32. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Francisco, CA (1993)
33. Breiman, L.: Random forests. Machine Learning **45** (2001) 5–32