Efficient Statement Identification for Automatic Market Forecasting

Henning Wachsmuth
Universität Paderborn
Software Quality Lab
hwachsmuth@slab.upb.de

Peter Prettenhofer and **Benno Stein**
Bauhaus-Universität Weimar
Web Technology & Information Systems
benno.stein@uni-weimar.de

Abstract

Strategic business decision making involves the analysis of market forecasts. Today, the identification and aggregation of relevant market statements is done by human experts, often by analyzing documents from the World Wide Web. We present an efficient information extraction chain to automate this complex natural language processing task and show results for the identification part. Based on time and money extraction, we identify sentences that represent statements on revenue using support vector classification. We provide a corpus with German online news articles, in which more than 2,000 such sentences are annotated by domain experts from the industry. On the test data, our statement identification algorithm achieves an overall precision and recall of 0.86 and 0.87 respectively.

1 Introduction

Touch screen market to hit $9B by 2015. 50 suppliers provide multi-touch screens, and that number is likely to rise.[1]

Strategic business decision making is a highly complex process that requires experience as well as an overall view of economics, politics, and technological developments. Clearly, for the time being this process cannot be done by a computer at the level of a human expert. However, important tasks may be automated such as market forecasting, which relies on identifying and aggregating relevant information from the World Wide Web (Berekoven et. al., 2001). An analyst who interprets the respective data can get a reasonable idea about the future market volume, for example. The problem is that a manually conducted Web search is time-consuming and usually far from being exhaustive. With our research we seek to develop an efficient system that finds and analyzes market forecast information with retrieval, extraction and natural language processing (NLP) techniques.

We contribute to the following situation. For a given product, technology, or industry sector we identify and aggregate statements on its market development found on relevant websites. In particular, we extract time information ("by 2015") and money information ("$9B") and use support vector classification to identify sentences that represent market statements. The statements' subjects ("touch screen") are found by relating recognized named entities to the time and money information, which we then normalize and aggregate. In this paper we report on results for the statement identification. To the best of our knowledge no data for the investigation of such market analysis tasks has been made publicly available until now. We provide such a corpus with statements on revenue annotated in news articles from the Web; the corpus was created in close collaboration with our industry partner *Resolto Informatik GmbH*.

We pursue two objectives, namely, to support human experts with respect to the effectiveness and completeness of their analysis, and to establish a technological basis upon which more intricate analysis tasks can be automated. To summarize, the main contributions of this paper are:

1. We show how to decompose the identification and aggregation of forecasts into retrieval, extraction, and normalization tasks.
2. We introduce a manually annotated German corpus for computational linguistics research on market information.
3. We offer empirical evidence that classification and extraction techniques can be com-

[1] Adapted from http://industry.bnet.com.

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 1128–1136,
Beijing, August 2010

bined to precisely identify statements on revenue.

1.1 Related Work

Stein et. al. (2005) were among the first to consider information extraction for automatic market forecasting. Unlike us, the authors put much emphasis on retrieval aspects and applied dependency grammar parsing to identify market statements. As a consequence their approach suffers from the limitation to a small number of predefined sentence structures.

While we obtain market forecasts by extracting expert statements from the Web, related approaches derive them from past market behavior and quantitative news data. Koppel and Shtrimberg (2004) studied the effect of news on financial markets. Lavrenko et al. (2000) used time-series analysis and language models to predict stock market prices and, similarly, Lerman et al. (2008) proposed a system for forecasting public opinion based on concurrent modeling of news articles and market history. Another related field is opinion mining in the sense that it relies on the aggregation of individual statements. Glance et al. (2005) inferred marketing intelligence from opinions in online discussions. Liu et al. (2007) examined the effect of Weblogs on box office revenues and combined time-series with sentiment analysis to predict the sales performance of movies.

The mentioned approaches are intended to reflect or to predict present developments and, therefore, primarily help for *operative* decision making. In contrast, we aim at predicting long-term market developments, which are essential for *strategic* decision making.

2 The Problem

Market forecasts depend on two parameters, the *topic* of interest and the *criterion* to look at. A topic is either an organization or a market. Under a market we unite branches, products, and technologies, because the distinction between these is not clear in general (e.g., for semiconductors). In contrast, we define a criterion to be a metric attribute that can be measured over time. Here we are interested in financial criteria such as revenue, profit, and the like. The ambitious overall task that we want to solve is as follows:

Task description: Given a topic τ and a financial criterion χ, find information for τ on the development of χ. Aggregate the found values on χ with respect to time.

We omit the limitation to forecasts because we could miss useful information otherwise:

(1) *In 2008*, the Egyptian automobile industry achieved *US$ 9.96bn* in sales.

(2) Egypt's automotive sales will rise *by 97% from 2008 to 2013*.

Both sentences have the same topic. In Particular, the 2008 amount of money from example (1) can be aggregated with the forecast in (2) to infer the predicted amount in 2013.

As in these examples, market information can often only be found in running text; the major source for this is the Web. Thus, we seek to find web pages with sentences that represent *statements on a financial criterion* χ and to make these statements processable. Conceptually, such a statement is a 5-tuple $\mathcal{S}_\chi = (S, g, T, M, t_d)$, where S is the topical subject, which may have a geographic scope g, T is a period of time, M consists of a growth rate and/or an amount of money to be achieved during T with respect to χ, and t_d is the statement time, i.e., the point in time when the statement was made.

3 Approach

Our goal is to find and aggregate statements on a criterion χ for a topic τ. In close collaboration with two companies from the semantic technology field, we identified eight high-level subtasks in the overall process as explained in the following. An overview is given in Table 1.

3.1 Find Candidate Documents

To find web pages that are likely to contain statements on χ and τ, we propose to perform a meta-search by starting from a set of characteristic terms of the domain and then using query expansion techniques such as local context analysis (Xu and Croft, 2000). As Stein et. al. (2005) describe,

Subtask	Applied technologies
1 Find candidate documents	meta-search, query expansion, genre analysis
2 Preprocess content	content extraction, sentence splitting, tokenization, POS tagging and chunking
3 Extract entities	time and money extraction, named entity recognition of organizations and markets
4 Identify statements	statistical classification based on lexical and distance features
5 Determine statement type	relation extraction based on dependency parse trees, matching of word lists
6 Fill statement templates	template filling, anaphora resolution, matching of word lists
7 Normalize values	time and money normalization, coreference resolution
8 Aggregate information	chronological merging and averaging, inference from subtopic to topic

Table 1: Subtasks of the identification and aggregation of market statements for a specified topic. Experiments in this paper cover the subtasks written in black.

a genre analysis, which classifies a document with respect to its form, style, and targeted audience, may be deployed afterwards to further improve the quality of the result list efficiently. In this way, we only maintain candidate documents that look promising on the surface.

3.2 Preprocess Content

Preprocessing is needed for accurate access to the document text. Our overall task incorporates relating information from different document areas, so mixing up a web page's main frame and sidebars should be avoided. We choose Document Slope Curve (DSC) for content detection, which looks for plateaus in the HTML tag distribution. Gottron (2007) has offered evidence that DSC is currently the best algorithm in terms of precision. Afterwards, the sentences are split with rules that consider the specific characteristics of reports, press releases and the like, such as headlines between short paragraphs. In succeeding subtasks, tokens as well as their Part-of-Speech and chunk tags are also used, but we see no point in not relying on standard algorithms here.

3.3 Extract Entities

The key to identify a statement $\mathcal{S}_\chi$ on a financial criterion χ is the extraction of temporal and monetary entities. Recent works report that statistical approaches to this task can compete with hand-crafted rules (Ahn et. al., 2005; Cramer et. al., 2007). In the financial domain, however, the focus is only on dates and periods as time information, along with currency numbers, currency terms, or fractions as money information. We found that with regular expressions, which represent the complex but finite structures of such phrases, we can achieve nearly perfect recall in recognition (see Section 5).

We apply named entity recognition (NER) of organizations and markets in this stage, too, so we can relate statements to the appropriate subjects, later on. Note that market names do not follow a unique naming scheme, but we observed that they often involve similar phrase patterns that can be exploited as features. NER is usually done by sequence labeling, and we use heuristic beam search due to our effort to design a highly efficient overall system. Ratinov and Roth (2009) have shown for the CoNLL-2003 shared task that Greedy decoding (i.e., beam search of width 1) is competitive to the widely used Viterbi algorithm while being over 100 times faster at the same time.

3.4 Identify Statements

Based on time and money information, sentences that represent a statement $\mathcal{S}_\chi$ can be identified. Such a sentence gives us valuable hints on which temporal and monetary entity stick together and how to interpret them in relation. Additionally, it serves as evidence for the statement's correctness (or incorrectness). Every sentence with at least one temporal and one monetary entity is a candidate. Criteria such as revenue usually imply small core vocabularies $\mathcal{L}_{pos}$, which indicate that a sentence is on that criterion or which often appear close to it. On the contrary, there are sets of words $\mathcal{L}_{neg}$ that suggest a different criterion. For a given text collection with known statements on χ, both $\mathcal{L}_{pos}$ and $\mathcal{L}_{neg}$ can be found by computing the most discriminant terms with respect to χ. A reasonable first approach is then to filter sentences

that contain terms from $\mathcal{L}_{pos}$ and lack terms from $\mathcal{L}_{neg}$, but problems arise when terms from different vocabularies co-occur or statements on different criteria are attached to one another.

Instead, we propose a statistical learning approach. Support Vector Machines (SVMs) have been proven to yield very good performance in both general classification and sentence extraction while being immune to overfitting (Steinwart and Christmann, 2008; Hirao et. al., 2001). For our candidates, we compute lexical and distance features based on $\mathcal{L}_{pos}$, $\mathcal{L}_{neg}$, and the time and money information. Then we let an SVM use these features to distinguish between sentences with statements on χ and others. At least for online news articles, this works reasonably well as we demonstrate in Section 5. Note that classification is not used to match the right entities, but to filter the small set of sentences on χ.

3.5 Determine Statement Type

The statement type implies what information we can process. If a sentence contains more than one temporal or monetary entity, we need to relate the correct T and M to each $\mathcal{S}_\chi$, now. The type of $\mathcal{S}_\chi$ then depends on the available money information, its *trend* and the *time direction*.

We consider four types of money information. χ refers to a period of time that results in a *new amount* A of money in contrast to its *preceding amount* A_p. The difference between A and A_p may be specified as an *incremental amount* Δ_A or as a *relative growth rate* r. M can span any combination of A, A_p, Δ_A and r, and at least A and r constitute a reasonable entity on their own. Sometimes the trend of r (i.e. decreasing or increasing) cannot be derived from the given values. However, this information can mostly be obtained from a nearby indicator word (e.g. "plus" or "decreased") and, therefore, we address this problem with appropriate word lists. Once the trend is known, any two types imply the others.

Though we are predominantly interested in *forecasts*, statements also often represent a *declaration* on achieved results. This distinction is essential and can be based on time-directional indicators (e.g. "next") and the tense of leading verbs. For this, we test both feature and kernel methods on dependency parse trees, thereby determining T and M at the same time. We only parse the identified sentences, though. Hence, we avoid running into efficiency problems.

3.6 Fill Statement Templates

The remaining subtasks are ongoing work, so we only present basic concepts here.

Besides T and M, the subject S and the statement time t_d have to be determined. S may be found within the previously extracted named entities using the dependency parse tree from Section 3.5 or by anaphora resolution. Possible limitations to a geographic scope g can be recognized with word lists. In market analysis, the approximate t_d suffices, and for most news articles t_d is similar to their release date. Thus, if no date is in the parse tree, we search the extracted temporal entities for the release date, which is often mentioned at the beginning or end of the document's content. We fill one template (S, g, T, M, t_d) for each $\mathcal{S}_\chi$ where we have at least S, T, and M.

3.7 Normalize Values

Since we base the extraction on regular expressions, we can normalize most monetary entities with a predefined set of rules. Section 3.5 implies that $M^* = (A^*, r^*)$ is a reasonable normalized form where A^* is A specified in million US-$ and r^* is r as percentage with a fixed number of decimals.[2] Time normalization is more complex. Any period should be transformed to $T^* = (t_s^*, t_e^*)$ consisting of the start date t_s^* and end date t_e^*. Following Ahn et. al. (2005), we consider fully qualified, deictic and anaphoric periods. While normalization of fully qualified periods like "from Apr to Jun 1999" is straightforward, deictic (e.g. "since 2005", "next year") and anaphoric mentions (e.g. "in the reported time") require a reference time. Approaches to resolve such references rely on dates or fully qualified periods in the preceding text (Saquete et. al., 2003; Mani and Wilson, 2000).[3]

[2] Translating the currency requires exchange rates at statement time. We need access to such information or omit the translation if only one currency is relevant.

[3] References to fiscal years even involve a whole search problem if no look-up table on such data is available.

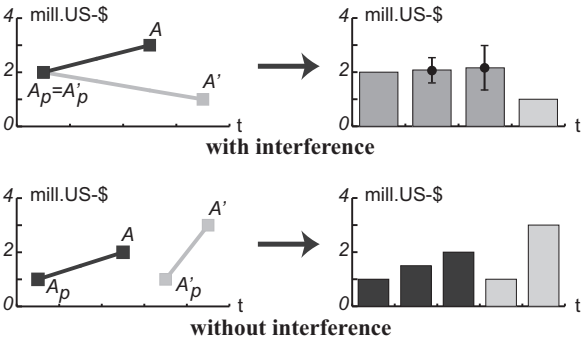

Figure 1: Example for merging monetary values.

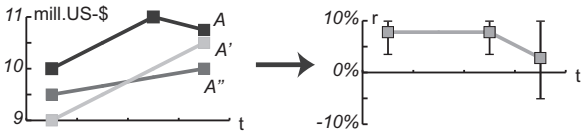

Figure 2: Example for the inference of relative information from absolute values.

If we cannot normalize M or T, we discard the corresponding statement templates. For the others, we have to resolve synonymous co-references (e.g. "Loewe AG" and "Loewe") before we can proceed to the last step.

3.8 Aggregate Information

We can aggregate the normalized values in either two or three dimensions depending on whether to separate statements with respect to t_d. Aggregation then incorporates two challenges, namely, how to merge values and how to infer information on a topic from values of a subtopic.

We say that two statements on the same topic τ and criterion χ *interfere* if the contained periods of time intersect and the according monetary values do not coincide. In case of declarations, this means that we extracted incorrect values or extracted values incorrectly. For forecasts, on the contrary, we are exactly onto such information. In both cases, an intuitive solution is to compute the average (or median) and deviations. Figure 1 graphically illustrates such merging. The subtopic challenge is based on the assumption that a meaningful number of statements on a certain subtopic of τ implies relative information on τ, as shown in Figure 2. One of the most interesting relations are organizations as subtopics of markets they produce for, because it is quite usual to search for information on a market, but only receive statements on companies. Approaches to this relation may rely e.g. on the web page co-occurrence and term frequencies of the markets and companies.

Statements	**Total**	**Forecasts**	**Declarations**
Complete corpus	2075	523 (25.2%)	1552 (74.8%)
Training set	1366	306 (22.4%)	1060 (77.6%)
Validation set	362	113 (31.2%)	249 (68.8%)
Test set	347	104 (30.0%)	243 (70.0%)

Table 2: Statements on revenue in the corpus.

Altogether, we return the aggregated values linked to the sentences in which we found them. In this way, we make the results verifiable and, thereby, compensate for possible inaccuracies.

4 Corpus

To evaluate the given and related tasks, we built a manually annotated corpus with online news articles on the revenues of organizations and markets. The compilation aims at being representative for target documents, a search engine returns to queries on revenue. The purpose of the corpus is to investigate both the structure of sentences on financial criteria and the distribution of associated information over the text.

The corpus consists of 1,128 German news articles from the years 2003 to 2009, which were taken from 29 news websites like *www.spiegel.de* or *www.capital.de*. The content of each document comes as unicode plain text with appended URL for access to the HTML source code. Annotations are given in a standard XMI file preformatted for the *Unstructured Information Management Architecture* (Ferrucci and Lally, 2004). We created a split, in which 2/3 of the documents constitute the training set and each 1/6 refers to the validation and test set. To simulate real conditions, the training documents were randomly chosen from only the seven most represented websites, while the validation and test data both cover all 29 sources. Table 2 shows some corpus statistics, which give a hint that the validation and test set differ significantly from the training set. The corpus is free for scientific use and can be downloaded at http://infexba.upb.de.

Loewe AG: Vorläufige Neun-Monats-Zahlen

Kronach, [6. November 2007]$_{REF}$ — Das Ergebnis vor Zinsen und Steuern (EBIT) des Loewe Konzerns konnte in den ersten 9 Monaten 2007 um 41% gesteigert werden. Vor diesem Hintergrund hebt die [Loewe AG]$_{ORG}$ ihre EBIT-Prognose für das laufende Geschäftsjahr auf 20 Mio. Euro an. ***Beim Umsatz strebt Konzernchef [Rainer Hecker]$_{AUTH}$ [für das Gesamtjahr]$_{TIME}$ ein höher als ursprünglich geplantes [Wachstum]$_{TREND}$ [von 10% auf ca. 380 Mio. Euro]$_{MONEY}$ an.*** *(...)*

Figure 3: An annotated document in the corpus. The text is taken from *www.boerse-online.de*, but has been modified for clarification.

4.1 Annotations

In each document, every sentence that includes a temporal entity T and a monetary entity M and that represents a *forecast* or *declaration* on the revenue of an organization or market is marked as such. T and M are annotated themselves and linked to the sentence. Accordingly, the *subject* is tagged (and linked) within the sentence boundaries if available, otherwise its last mention in the preceding text. The same holds for optional entities, namely a *reference time*, a *trend indicator* and the *author* of a statement. Altogether, 2,075 statements are tagged in this way. As in Figure 3, only information that refers to a statement on revenue (typed in bold face) is annotated. These annotations may be spread across the text.

The source documents were manually selected and prepared by our industrial partners, and two of their employees annotated the plain document text. With respect to the statement annotations, a preceding pilot study yielded substantial inter-annotator agreement, as indicated by the value $\kappa = 0.79$ of the conservative measure *Cohen's Kappa* (Carletta, 1996). Additionally, we performed a manual correction process for each annotated document to improve consistency.

5 Experiments

We now present experiments for the statement identification, which were conducted on our corpus. The goal was to evaluate whether our combined extraction and classification approach succeeds in the precise identification of sentences that comprise a statement on revenue, while keeping recall high. Only exact matches of the annotated text spans were considered to be correct identifications. Unlike in Section 3, we only worked on plain text, though.

5.1 Experimental Setup

To find candidate sentences, we implemented a sentence splitter that can handle article elements such as subheadings, URLs, or bracketed sentences. We then constructed sophisticated, but efficient regular expressions for time and money. They do not represent correct language, in general, but model the structure of temporal and monetary entities, and use word lists provided by domain experts on the lowest level.[4] For feature computation, we assumed that the closest pair of temporal and monetary entity refers to the enclosing candidate sentence.[5] Since only positive instances I_P of statements on revenue are annotated in our corpus, we declared all candidates, which have no counterpart in the annotated data, to constitute the negative class I_N, and balanced I_P and I_N by "randomly" (seed 42) removing instances from I_N.[6]

For the vocabularies $\mathcal{L}_{pos} = \{P_1, P_2\}$ we first counted the frequencies of all words in the unbalanced sets I_P and I_N. From these, we deleted named entities, numbers and adjectives. If the prefix (e.g. "Umsatz") of a word ("Umsatzplus") occurred, we only kept the prefix. We then filtered all terms that appeared in at least 1.25% of the instances in I_P and more than 3.5 times as much in I_P as in I_N. The remaining words were manually partitioned into two lists:

P_1 = {umgesetzt, Umsatz, Umsätze, setzte} (all of these are terms for revenue)

P_2 = {Billionen, meldet, Mitarbeiter, Verband} (trillions, announce, employee, association)

$\mathcal{L}_{neg} = \{N_1, N_2\}$ was built accordingly. In addition, we set up a list G_1 with genitive pronouns

[4] More details are given at http://infexba.upb.de.

[5] 55% of the candidate sentences in the training set contain more than one temporal and/or monetary entity, so this assumption may lead to errors.

[6] We both tested undersampling and oversampling techniques but saw no effective differences in the results.

and determiners. Based on $\mathcal{L}_{pos}$, $\mathcal{L}_{neg}$ and G_1, we computed the following 43 features for every candidate sentence s:

- **1-8:** Number of terms from P_1 (N_1) in s as well as in the two preceding sentences and in the following sentence.
- **9-10:** Number of terms from P_2 (N_2) in s.
- **11:** Occurrence of term from G_1 next to the monetary entity.
- **12-19:** Forward (backward) distance in tokens between the monetary (temporal) entity in s and a term from P_1 (N_1).
- **20-27:** Forward (backward) distance in number of symbols from $O_1 = \{$'.','?','!'$\}$ between the monetary (temporal) entity in s and a term from P_1 (N_1).
- **28-43:** Same as 20-27 for $O_2 = \{$':',';'$\}$ and $O_3 = \{$','$\}$, respectively.

We trained a linear SVM with cost parameter $C = 0.3$ (selected during validation) on these features using the *Weka* integration of *LibSVM* (Hall et. al., 2009; Fan et. al., 2001). Further features were evaluated, e.g. occurrences of contrapositions or comparisons, but they did not improve the classifier. Instead, we noticed that we can avoid some complex cases when we apply two rules after entity extraction:

R_1: Delete temporal and monetary entities that are directly surrounded by brackets.

R_2: Delete temporal entities that contain the word "Vorjahr" ("preceding year").

Now, we evaluated the following five statement identification algorithms:

- **Naïve:** Simply return all candidate sentences (to estimate the relative frequency of statements on revenue in the corpus).
- **Baseline:** Return all candidate sentences that contain a term from the list P_1.
- **NEG:** Use the results from Baseline. Return all sentences that lack terms from N_1.

Recall	Training	Validation	Test
Sentences	0.98	0.98	0.96
Temporal entities	0.97 (0.95)	0.97 (0.94)	0.98 (0.96)
Monetary entities	0.96 (0.96)	0.96 (0.96)	0.95 (0.94)

Table 3: Recall of sentence and entity extraction. In brackets: Recall after applying R_1 and R_2.

- **RB:** Filter candidates using R_1 and R_2. Then apply NEG.
- **SVM:** Filter candidates using R_1 and R_2. Then classify sentences with the SVM.

5.2 Results

Table 3 shows that we found at least 95% of the sentences, time and money information, which refer to a statement on revenue, in all datasets.[7] We could not measure precision for these since not all sentences and entities are annotated in the corpus, as mentioned in Section 4.

Results for the statement identification are given in Figure 4. Generally, the test values are somewhat lower than the validation values, but analog in distribution. Nearly all statements were recognized by the Naïve algorithm, but only with a precision of 0.35. In contrast, both for Baseline and NEG already around 80% of the found statements were correct. The latter paid a small gain in precision with a significant loss in recall. While RB and SVM both achieved 86% precision on the test set, SVM tends to be a little more precise as suggested by the validation results. In terms of recall, SVM clearly outperformed RB with values of 89% and 87% and was only a little worse than the Baseline. Altogether, the F_1-Measure values show that SVM was the best performing algorithm in our evaluation.

5.3 Error Analysis

To assess the influence of the sentence, time and money extraction, we compared precision and recall of the classifier on the manually annotated and the extracted data, respectively. Table 4 shows

[7] We intentionally did not search for unusual entities like "am 1. Handelstag nach dem Erntedankfest" ("the 1st trading day after Thanksgiving") in order not to develop techniques that are tailored to individual cases. Also, money amounts that lack a currency term were not recognized.

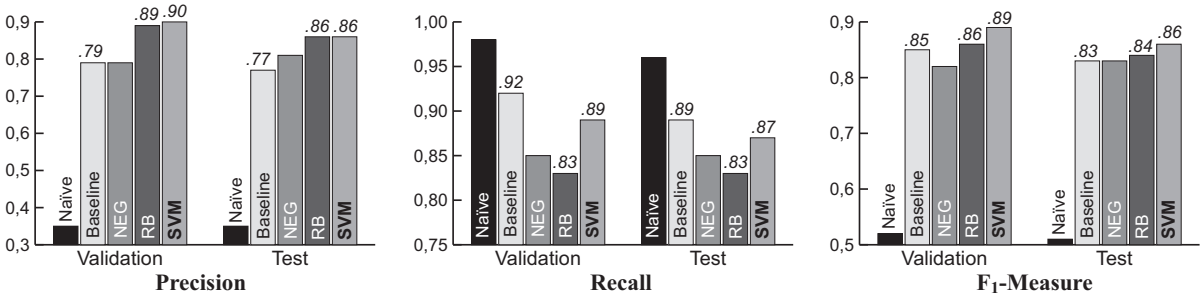

Figure 4: Precision, recall and F_1-Measure of the five evaluated statement identification algorithms. SVM is best in precision both on validation and test data and outperforms RB in recall significantly.

that only recall differs significantly. We found that false statement identifications referred to the following noteworthy error cases.

False match: Most false positives result from matchings of temporal and monetary entities that actually do not refer to the same statement.

Missing criterion: Some texts describe the development of revenue without ever mentioning revenue. Surrogate words like "market" may be used, but they are not discriminative enough.

Multiple criteria: Though we aimed at discarding sentences, in which revenue is mentioned without comprising a statement on it, in some cases our features did not work out, mainly due to intricate sentence structure.

Traps: Some sentences contain numeric values on revenue, but not the ones looked for, as in "10% of the revenue". We tackled these cases, but had still some false classifications left.

Hidden boundaries: Finally, we did not find all correct sentence boundaries, which can lead to both false positives and false negatives. The predominant problem was to separate headlines from paragraph beginnings and is partly caused by the missing access to markup tags.

5.4 Efficiency

We ran the identification algorithm on the whole corpus using a *2 GHz Intel Core 2 Duo MacBook* with *4 GB RAM*. The 1,128 corpus documents contain 33,370 sentences as counted by our algorithm itself. Tokenization, sentence splitting, time and money extraction took only 55.2 seconds, i.e., more than 20 documents or 600 sentences each second. Since our feature computation is not optimized yet, the complete identification process is a little less efficient with 7.35 documents or 218 sentences per second. However, it is fast enough to be used in online applications, which was our goal in the end.

Candidates	Data	Precision	Recall
Annotated	validation data	0.91	0.94
	test data	0.87	0.93
Extracted	validation data	0.90	0.89
	test data	0.86	0.87

Table 4: Precision and recall of the statement identification on manually annotated data and on automatically extracted data, respectively.

6 Conclusion

We presented a multi-stage approach for the automatic identification and aggregation of market statements and introduced a manually annotated German corpus for related tasks. The approach has been influenced by industry and is oriented towards practical applications, but is, in general, not specific to the German language. It relies on efficient retrieval, extraction and NLP techniques. By now, we can precisely identify most sentences that represent statements on revenue. This already allows for the support of strategists, e.g. by highlighting such sentences in web pages, which we currently implement as a Firefox extension. The overall problem is complex, though, and we are aware that human experts can do better at present. Nevertheless, time-consuming tasks can be automated and, in this respect, the results on our corpus are very promising.

Acknowledgement: This work was funded by the project "InfexBA" of the German Federal Ministry of Education and Research (BMBF) under contract number 01IS08007A.

References

Ahn, David, Sisay F. Adafre, and Maarten de Rijke. 2005. Extracting Temporal Information from Open Domain Text: A Comparative Exploration. *Journal of Digital Information Management*, 3(1): 14–20.

Berekoven, Ludwig, Werner Eckert, and Peter Ellenrieder. 2001. *Marktforschung: Methodische Grundlagen und praktische Anwendung*, 9th Edition, Gabler, Wiesbaden, Germany.

Carletta, Jean. 1996. Assessing Agreement on Classification Tasks: The Kappa Statistic. *Computational Linguistics*, 22: 249–254.

Cramer, Irene M., Stefan Schacht, and Andreas Merkel. 2007. Classifying Number Expressions in German Corpora. In *Proceedings of the 31st Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications*, pages 553–560.

Fan, Rong-En, Pai-Hsuen Chen, and Chih-Jen Lin. 2001. Working Set Selection Using Second Order Information for Training Support Vector Machines. *Journal of Machine Learning Research*, 6: 1889–1918.

Ferrucci, David and Adam Lally. 2004. UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3–4): pages 327–348.

Glance, Natalie, Matthew Hurst, Kamal Nigam, Matthew Siegler, Robert Stockton, and Takashi Tomokiyo. 2005. Deriving Marketing Intelligence from Online Discussion. In *Proceedings of the Eleventh International Conference on Knowledge Discovery in Data Mining*, pages 419–428.

Gottron, Thomas. 2007. Evaluating Content Extraction on HTML Documents. In *Proceedings of the 2nd International Conference on Internet Technologies and Applications*, pages 123–132.

Hall, Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1).

Hirao, Tsutomu, Hideki Isozaki, Eisaku Maeda and Yuji Matsumoto. 2002. Extracting Important Sentences with Support Vector Machines. In *Proceedings of the 19th International Conference on Computational linguistics*, pages 342–348.

Koppel, Moshe and Itai Shtrimberg. 2004. Good News or Bad News? Let the Market Decide. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, pages 86–88.

Lavrenko, Victor, Matt Schmill, Dawn Lawrie, Paul Ogilvie, David Jensen, and James Allan. 2000. Mining of Concurrent Text and Time Series. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Workshop on Text Mining*, pages 37–44.

Lerman, Kevin, Ari Gilder, Mark Dredze, and Fernando Pereira. 2008. Reading the Markets: Forecasting Public Opinion of Political Candidates by News Analysis. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 473–480.

Liu, Yang, Xiangji Huang, Aijun An, and Xiaohui Yu. 2007. Arsa: A Sentiment-Aware Model for Predicting Sales Performance Using Blogs. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 607–614.

Mani, Inderjeet and George Wilson. 2000. Robust Temporal Processing of News. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 69–76.

Ratinov, Lev and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155.

Saquete, Estela, Rafael Muñoz, and Patricio Martínez-Barco. 2003. TERSEO: Temporal Expression Resolution System Applied to Event Ordering. *Text, Speech and Dialogue*, Springer, Berlin / Heidelberg, Germany, pages 220–228.

Stein, Benno, Sven Meyer zu Eissen, Gernot Gräfe, and Frank Wissbrock. 2005. Automating Market Forecast Summarization from Internet Data. *Fourth International Conference on WWW/Internet*, pages 395–402.

Steinwart, Ingo and Andreas Christmann. 2008. *Support Vector Machines*, Springer, New York, NY.

Xu, Jinxi and Bruce W. Croft 2000. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18(1): 79-112.