Did I Really Say That? – Combining Machine Learning & Dependency Relations to Extract Statements from German News Articles

Thomas Bögel **Michael Gertz**
Institute of Computer Science
Heidelberg University
69120 Heidelberg, Germany
{thomas.boegel,gertz}@informatik.uni-heidelberg.de

Abstract

We present a system to extract statements of public figures from unstructured German news articles. We first motivate and define statements as a temporally-aware extension of quotations and present the three categories of statements: (1) direct, (2) indirect, and (3) mixed-style statements. We use a combination of machine learning and heuristics based on dependency parses to tackle all three types of statements. The quality of our extraction approach is compared to related work in quotation attribution showing that rules based on syntactic structures increase the extraction quality compared to lexical patterns. In addition, we apply the system on a corpus of German news articles and show that it is able to extract statements with high precision (82.4%).

1 Introduction

> So betonte Merkel im TV-Duell [...]: "Mit mir wird es eine Maut für Autofahrer im Inland nicht geben."[1]

Statements reveal the general attitude of people and groups towards a topic. In the political context, these "attitudes" are referred to as *policy positions* and represent an important factor in political decision-making processes (e.g., Klüver (2009)) and help voters, for instance, to judge their political alignment with parties and groups. In a world of constantly growing amounts of news in different media, it is very hard to track policy positions of individual people over time. While there are publicly accessible protocols of statements in political debates, statements are also uttered outside of the parliament, and there is no repository of the policies of, for instance, influence groups.

[1] source: http://spon.de/ad4xR

Statements consist of direct quotes that are quite easy to extract from texts but there are also more subtle, indirect utterances. While there are many systems to extract quotations from English texts, there is only very few work on German texts. Furthermore, the existing systems for German only extract quotations, neglecting statements that fall under a more broader category of utterances.

In this paper, we present the first system for extracting a broad variety of statements from unstructured, German news articles at a large scale based on both machine learning and heuristics, as well as syntactic dependency relations (Section 4). In order to take the dynamics of statements into account, we incorporate a temporal dimension into the definition of statements and include it into our extraction process. To our knowledge, this is the first system to apply dependency-based rules to extract temporally-aware statements for German.

Having motivated the need for statements and contrasting statements with the task of quotation attribution in the next section, we present our approach in Section 4 and evaluate the system in Section 5. We compare the extraction quality of our system against lexical patterns presented in related work to answer the question whether using syntactic relationship enhances the accuracy of statement extraction. In addition, we apply our approach to a manually assembled German news corpus and measure precision on the statement level. Finally, we will conclude our findings and present our ongoing work in Section 6.

2 Background and Definition of Statements

2.1 Statements vs. Quotations

The task of *quotation attribution* is to extract quotations of people from unstructured text (e.g., Pouliquen et al. (2007)). Following the definition of Pareti (2012), the task of quotation attri-

Proceedings of the Int. Conference of the German Society for Computational Linguistics and Language Technology,
pages 13–21, University of Duisburg-Essen, Germany, Sep 30–Oct 2 2015.

bution involves three components: extracting the *source*, the *cue*, and the *content*. The *source* represents the person or organisation that a quotation is attributed to, the *cue* is a verb or indicator for a quotation (e.g., "sagte") and *content* contains the actual quotation being uttered. Usually, systems distinguish between direct and indirect quotations and usually cover common verbs indicating a quotation (e.g., "sagte").

With this narrow definition of quotation attribution, statements that should be extracted are missed. Most systems restrict themselves to utterances with a small set of specific verbs that are associated with explicit quotations. The sentence "Angela Merkel kritisierte das EU-Abkommen zur…", for instance, reports about a statement that does actually not directly represent a quotation. Thus, despite being a noteworthy statement for the extraction of policy positions, it would not be extracted by existing systems for German quotation extraction.

(1) (a) Wie eine Sprecherin (am Montag)$_{timestamp}$ sagte, …
(b) Westerwelle kritisiert (NSA-Spähaktion)$_{target}$ …

Another aspect is the inherent temporal dimension of statements: in order to capture the evolution of statements (and thus, policies) over time, it is important to determine when a statement was made. While a common approach like using the publication date of an article might work for many cases, statements can be explicitly time-stamped and thus override the article timestamp as example 1(a) shows. We capture this in our definition of statements as well as during statement extraction. Finally, statements may have an explicit target, meaning they are directed against a specific person, topic (illustrated in example 1(b)) or organization. These targets are valuable for modeling policy positions (Van Atteveldt et al., 2008).

Definition of statements. Taking the above considerations into account, we extend the definition of quotations to develop a broader concept of utterances, called **statements**, as our extraction target. Overall, quotation extraction can be seen as a subpart of statement extraction.

We define a statement as a tuple

$$statement = \langle source, cue, content, target, timestamp \rangle$$

where *source*, *cue*, and *content* are identical to the definition of quotations (see Section 2.1). We add an optional element *target* representing the target of a statement. In addition, each statement is time-stamped. Note that the content can be identical to the cue, as illustrated below.

In the following, we present the differences between three types of statements and discuss how the elements of a statement defined above are realized.

2.1.1 Direct Statements

The typical case for direct statements is a reporting verb (*cue*) and the actual content of the statement in quotes, for example:

(2) "Die Kämpfe dauern unvermindert an", sagt$_{cue}$ (ein Sprecher)$_{source}$.

For direct statements, the *content* is represented by the quoted text passage. In this example, the direct statement is embedded into the sentence and thus the *cue* and *source* are realized within the same sentence. There are also scenarios where the sentence containing the content precedes or succeeds the cue and source. Note that quoted text passages in isolation do not always hint towards a statements but are also used to highlight text passages, for instance. We will deal with these cases in Section 4.1.

2.1.2 Indirect Statements

There are different ways of reporting statements indirectly. We will just present two representative examples showing commonly used sentence structures:

(3) (a) (Angela Merkel)$_{source}$ sagte$_{cue}$, dass dies nicht akzeptabel sei.
(b) (Angela Merkel)$_{source}$ kritisierte$_{cue}$ (die Wahlen in der Ostukraine)$_{target,content}$.

In example 3(a), the content is expressed in a subordinate clause governed by the cue "sagte". The mood of the subordinate clause is subjunctive. The conjunction for the subordinate clause "dass" is optional. Example 3(b) represents a statement that is not covered by a strict definition of quotation but nevertheless expresses an important statement. In this case, the content (i.e., the direct object of the cue) is actually identical to the target of the statement. Thus, a system for statement extraction should not only be able to extract phrasal elements but also complex noun phrases as statement contents.

2.1.3 Mixed Statements

Oftentimes, direct and indirect statements are mixed within the same sentence to combine summaries of statements with direct quotes. There are different variations of mixed statements (e.g., coordinated statements), the following representing a very common pattern:

(4) (Angela Merkel)$_{source}$ bezeichnete$_{cue}$ [(die Wahlen in der Ostukraine)$_{target}$ als "illegitim"]$_{content}$.

As the example shows, the elements of a statement are actually realized similarly to indirect statements. The only difference is that passages in direct speech are embedded into the content. This needs to be taken into account when extracting direct utterances: while the content "illegitim" in isolation is not meaningful by itself, the complete predicative construction is required. Note also that in this case the target is part of the content.

2.1.4 Multi-Sentence Statements

Up until now, we only investigated single sentences as statements. Naturally, a statement can consist of a sequence of sentences. There are again variations of multi-sentence statements. The following example illustrates a frequently occurring pattern:

(5) Angela Merkel bezeichnete die Wahlen in der Ostukraine als "illegitim". *Die Bundesregierung würde diese nicht akzeptieren.*

The sentence in italics in isolation can only be understood as a statement when context – in this case from the previous sentence – is used for interpretation, because the sentence itself does not contain any cue or source. A sentence in subjunctive mood succeeding a sentence containing a statement, however, is a reliable predictor that the statement in the previous sentence is continued.

3 Related Work

As indicated above, the task of statement extraction is highly related to the extraction and attribution of quotations. Thus, we build on similar strategies from related work in this area. While most systems performing quotation extraction focus on languages other than German, some of the fundamental strategies can be applied across different languages as well in an adapted form.

The systems performing quotation extraction can be broadly classified into two categories: rule-based or heuristic and supervised machine learning-based extraction. A comprehensive survey of related work can be found in (O'Keefe, 2014).

Heuristic quotation extraction. Sarmento et al. (2009) propose a simple heuristic system to extract quotes and corresponding sources based on pattern matching of 19 flat patterns from Portuguese news articles. With their simple method that neglects coreference resolution, they are only able to extract quotes for about 5% of all articles. Similarly, Pouliquen et al. (2007) use a small set of simple patterns involving utterance verbs to extract quotations in 11 different languages. Due to the limited number of patterns, they also aim at high precision at the loss of recall. Krestel et al. (2008) employ a two-step approach: after searching for reporting verbs (cues), 6 lexical patterns are applied to extract the content and source of a statement. Their evaluation on just 7 articles from the Wall Street Journal yields a good precision (99%) with reasonable recall (74%).

Overall, while flat, lexical patterns work well for languages with a relatively fixed sentence construction (like English), we will show that a dependency-based approach captures the nature of languages with more variations in sentence construction like German or French in a better way. While de La Clergerie et al. (2011) employ syntactic patterns to extract quotes like we do, they always rely on the complement of an utterance verb to extract the statement content. In contrast, we encode possible realizations of statement elements for each verb individually.

Supervised approaches. Pareti et al. (2013) implement a supervised system for extracting quotes in three steps. First, a classifier predicts whether a verb is a valid cue for a statement based on a set of 20 features. To extract the content of a quote, they employ a token-based as well as a constituent-based classifier. The former performs sequence tagging using Conditional Random Fields to predict for each token whether it is part of a quote. In the constituent-based approach, each syntactic constituent is classified with a MaxEnt classifier, followed by a final post-processing step to unify individual predictions. Their evaluation on the Penn Attribution Relations Corpus (Pareti, 2012) shows that sequence tagging on the token-level performs

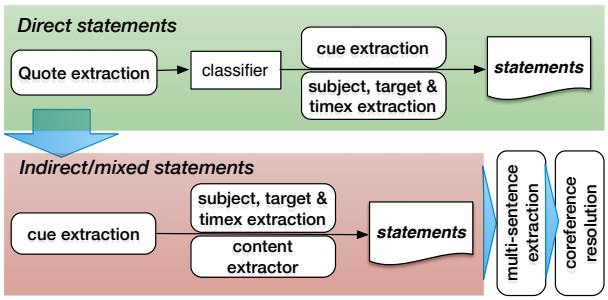

Figure 1: Workflow of statement extraction.

better than classifying constituents or rule-based approaches. The fact that dependency structures yield no improvement might, again, be due to the relatively fixed English sentence structure.

There are only two systems that extract quotes from German texts: as illustrated above, Pouliquen et al. (2007) present a multi-lingual approach suffering from low recall due to the low number of employed patterns. Ploch (2015) have recently presented a system to extract direct, indirect and mixed quotations based on lexical patterns. They only cover a relatively small number (25) of utterance verbs, yielding a low coverage. We, in contrast, follow a broader definition of statement to greatly extend coverage. In addition, instead of rule-based filtering of direct quotations, we use machine learning to filter out misleading quoted text passages. Finally, we employ syntactic dependency relations instead of lexical patterns to extract elements of a statement.

To the best of our knowledge, we present the first system that extracts a broad variety of temporally-aware statements from German texts while explicitly taking into account the target of a statement.

4 Methods

After preprocessing, each document is passed through a sequential pipeline illustrated in Figure 1 that first extracts direct statements using a machine learning-based approach (Section 4.1). For the remaining sentences, indirect and mixed statements are searched (Section 4.2). In a final step, multi-sentence statements are detected and coreference resolution is performed to resolve the sources of statements.

Preprocessing. We implement a modular preprocessing pipeline based on UIMA[2]. After PoS-tagging with the TreeTagger (Schmid, 1999), we perform a morphological analysis using Morphisto (Zielinski and Simon, 2009). To extract the timestamp of a statement, temporal expressions are extracted with HeidelTime (Strötgen and Gertz, 2013). Named Entities are extracted using StanfordNER (Finkel et al., 2005) and a German model (Bingel and Haider, 2014). Finally, the text is parsed with ParZu (Sennrich et al., 2009) to obtain dependency structures and processed with CorZu (Klenner and Tuggener, 2011) to resolve coreferences.

4.1 Direct statement extraction

Extracting direct statements based on quotation marks and heuristics to filter out quoted text passages that do not denote actual statements yields reasonable performance (Pouliquen et al., 2007). There are, however, cases when simple heuristics fail. Many heuristics for filtering quoted strings take into account the length of the text passage. While the quoted string in example (6) consists of only 3 tokens, it is nevertheless a valid statement. Conversely, there are many instances, especially for mixed statements, where longer quoted text passages themselves are only part of a complex statement.

(6) “Deutschland ist Wachstumsmotor”

Instead of manually defining a complex rule set to distinguish between direct statements and highlights etc., we use machine learning to derive rules and thresholds automatically.

As mentioned in Section 2.1.1, the content of direct statements equals text passage enclosed by quotation marks. To extract the remaining elements of a statement, we determine heuristically which sentence reports the statement, i.e., which sentence contains the cue and the other elements of the statement. We first check whether the direct statement is embedded in a sentence.[3] Otherwise, we determine whether the preceding sentence is the reporting sentence by searching for a cue or a colon at the end of the sentence. Finally, the reporting sentence is processed as described in the following section.

[2] https://uima.apache.org/

[3] Quoted passages consisting of multiple sentences are also taken into account.

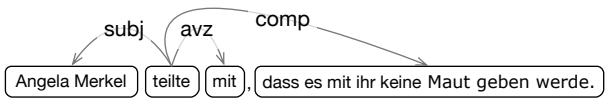

Figure 2: Dependency relations for statement cues.

4.2 Statement Extraction using Dependency Trees

While there are annotated corpora for statement extraction in English (e.g., Pareti (2012)), this is not the case for German. To our knowledge, there exists just one data set from Ploch (2015) that only captures a small amount of statements. Due to the lack of training data and the fact that a rule-based system allows for fine-grained control of encoding context information, we implement a rule-based approach to extract elements of statements. In addition, while we are currently working with news articles from the politics section, switching to another category (such as sports or economy) might require additional statement cues or phrases and thus – in the case of machine learning – additional training data. As our rules are completely independent from the code base, switching to a different domain can easily be done by extending the rule set.

Our approach makes use of dependency relations between the cue verb and other elements of a statement. Figure 2 shows a dependency representation of a sentence containing a statement. The dependency relations governed by the cue (*teilte*) represent the different elements of a statement: the *subject* represents the source and the statement content is realised by the *comp* node.

Syntactic dependency structures present multiple advantages over lexical patterns. In contrast to purely string-based patterns – such as a heuristic regarding everything in a sub-sentence beginning with "dass" as the content of a statement – rules based on dependency relations can be much more fine-grained with respect to elements that should be included in the content string. In addition, syntactic structures are better able to identify the source of a statement by automatically extracting the subject of sentences.

Lexicon of statement cues. To recognize possible cues of statements, we compiled a lexicon consisting of more than 200 verbs indicating a statement by looking at suitable synsets for utterance verbs in GermaNet (Hamp and Feldweg, 1997). In addition to the lemma of each verb, we encoded how elements of statements are realised with respect to dependency relations. Details are explained below.

Extracting the cue. The extraction process begins with extracting the main verb (head) of each sentence. Auxiliary and modal constructions are also taken into account and resolved (e.g., "wollte betonen"). As indicated in Figure 2, particle verbs are also correctly handled. This is necessary to disambiguate verbs that have different meanings with different verb particles. In the example above, "teilen" can only be used as a statement cue in combination with the particle "mit", resulting in "mitteilen". Without proper handling of verb particles, either coverage or quality would suffer due to missed or erroneous statements, respectively. If a match in our verb lexicon is found for the head of the sentence, the remaining elements of the statement are extracted next.

Extracting the source. In most cases, the subject of the cue is identical to the source of a statement, but there are exceptions. In the sentence "Er zitierte Angela Merkel", for instance, the subject of the cue does not actually represent the source. The source is realised as the direct object. These exceptions are encoded in our verb lexicon. Additionally, we normalize passive constructions like "Angela Merkel wurde von Horst Seehofer kritisiert" resulting in the source of a statement being expressed as a prepositional construction. Attributes of the source are not extracted, and if a source partially matches a named entity of type `PERSON` predicted by StanfordNER, we link the statement to the named entity and optionally correct the span of the respective source so that it aligns with the named entity.

Extracting the content. As the content for direct speech statements is already defined by the quoted text passage, content extraction is only performed for indirect statements. For each verb, we encode how the content of an utterance is typically expressed in terms of dependency relations. While the content is oftentimes expressed as a phrasal complement (such as in Figure 2), there are cases where different patterns need to be employed: the verb "bezeichnen", for example, does not take phrasal complements but employs a predicative construction to express the statement content.

Extracting the target and timestamp. Statements often directly refer to a person or a specific topic. Some utterance verbs directly encode the target, such as "kritisieren" or "verlangen". For verbs with an explicit syntactic slot for a target, we encode the corresponding dependency relation in our lexicon. "verlangen", for example, can represent the target of a statement as a prepositional object with the preposition "von".

To extract the timestamp of a statement, we use the output of the temporal tagger HeidelTime: we check whether a temporal expression fills a modifier position of the cue (as in Example 7(a)). In contrast, temporal expressions within the utterance *content* are neglected, as they do not indicate the point in time of a statement (see Example 7(b)). If no timestamp can be found, the timestamp of the article is used.

(7) (a) Wie eine Sprecherin (am Montag)$_{timestamp}$ sagte, ...
(b) Wie eine Sprecherin sagte, war am (am Montag)$_{timestamp}$ ...

Mixed statements. As explained in Section 4.1, quoted strings that are part of a mixed statement should be filtered out by our classifier. As mixed statements are usually governed by the same cues as indirect statements, mixed statements are tackled as indirect statements and the quoted text passage is simply integrated into the content string.

4.3 Multi-sentence Statements and Coreference

After all components have been passed through, we iterate over all sentences to determine whether a sentence should be attached to an already existing statement as explained in Section 2.1.4. If a sentence in subjunctive mood directly succeeds a statement, it is attached to the previous statement. We rely on the morphology component to determine whether a verb is in subjunctive mood, taking into account the subject of the verb for ambiguous verb forms. For continued statements, the cue, source, target and timestamp are often not explicitly mentioned again. In this case, we adopt all four elements from the immediately preceding statement.

Coreference Resolution. The source of statements is often represented indirectly by pronouns or noun references. In order to attribute statements to the correct person, we cluster all mentions of an entity in a text using a modified[4] version of CorZu to perform coreference resolution.

5 Experiments and Evaluation

To evaluate our system, we use two different approaches and data sets: we first assess the quality of extracted quotes by measuring token-based precision and recall of statement components. In addition, we compute the precision of our system with respect to extracted statements.

In the next section, we will first present the two data sets that are used for evaluation, followed by experiments and results for filtering direct quotations in Section 5.2. Finally, we perform a token-based (Section 5.3) and statement-based (Section 5.4) evaluation of our system.

5.1 Data Sets.

Token-based evaluation. In Ploch (2015), two annotators manually annotated the source, cue and the content of quotes in 287 random news articles. For the source, coreference should be resolved and the fully specified name be used as the speaker. The annotation resulted in 383 quotations. We use this data set to compare our approach to the system by Ploch (2015). While the annotations are suitable for evaluating the correctness of *extracted* quotations – i.e., how well the predicted spans match manual annotations – there are two issues: first, only quotations are covered, meaning a sub-set of all statements. Second, due to the focus on token-based accuracy, only a fraction of all quotations are annotated in the data set, meaning it cannot be used to evaluate performance on the statement level.

NSA data set. We manually created an additional data set covering a specific topic – the NSA spying scandal. We chose a well-defined topic to be better able to judge statements in contrast to random news articles. We collected all articles related to the topic between 06-2013 and 12-2014 from two major German news sites, *Spiegel Online*[5] and *FAZ*[6], resulting in about 1200 articles.

5.2 Filtering Direct Quotations

To train a random forests classifier (Breiman, 2001) predicting whether quoted strings are valid statements, we manually annotated 1000 quoted text

[4] We extended the list of proper names and corresponding gender assignments.

[5] www.spiegel.de/

[6] www.faz.net/

	Precision	Recall	F_1
Baseline	71.3	94.7	81.4
Random Forest	96.9	98.0	**97.5**

Table 1: Results of predicting direct statements using machine learning compared to a rule-based baseline approach.

passages from 80 documents. For each text passage, an annotator should decide whether the text passage represents a direct statement. Thus, highlights and misused quoted strings were annotated as erroneous. As mentioned in Section 2.1.3, quoted text passages within mixed statements in isolation are also not sufficient and were thus marked as erroneous, too.

We used a combination of 12 features. Besides the literal string that is quoted, we determine whether (2) the quoted string contains a verb and (3) the preceding sentence ends with a colon. The length of the quoted string is counted in (4) token and (5) characters and both counts are also estimated (6,7) relative to the sentence length. Finally, we check whether the (8) preceding and (9) proceeding sentences also contain a quoted string or an (10,11) utterance verb. A (12) *type* feature determines whether the quotation spans the whole sentence or it is embedded.

Baseline. To check if our machine learning approach performs better than a simple rule set, we compare the trained models to a baseline approach filtering all quoted text passages that consist of at most 3 tokens and do not contain any verb.

Results. To measure the impact of training data size, we first fitted the model on 500 quoted strings and then gradually increased the number to 1000, measuring performance differences with 10-fold cross-validation. For more than 800 instances, the results differed only marginally. The results of 10-fold cross-validation for predicting direct quotations in Table 1 show that the classifier outperforms the baseline by far. The drastic increase in precision of about 15% shows that using simple rules to filter out quoted strings yields many more false positives erroneously being predicted as statements. Overall, with a relatively low number of manual annotations, good results can be achieved.

5.3 Token-Based Evaluation of Quotations

The quality of extracted statements with respect to the number of tokens correctly annotated is summarised in Table 2. We report the numbers of Ploch (2015) alongside our results. For all elements of direct statements, our approach clearly outperforms the baseline system. Regarding indirect and mixed statements, our system improves recall of the cue and source extraction at a slight loss of precision, resulting in a higher F-Score. The performance of content extraction is especially promising with a high increase of precision and recall for all statement types. The F-Score of content extraction for indirect statements, for example, increased by more than 9 percentage points from 76.4% to 85.5%.

Error analysis. Manual error inspection revealed that the loss of precision for the source is often due to erroneous coreference resolution or named entity recognition. It seems that simple rules to resolve pronoun coreference might perform better than applying full-fledged coreference resolution. Errors in the content string are mostly caused by parser errors, especially for complex sentence structures. While errors like these are obviously inherent to a method that relies on syntactic dependencies and could be resolved by a sequence tagging approach, complex cases and sentence structures will probably require many more training data. Another issue are sentence constructions that could be resolved by individual rules but occur very rarely, such as the one given in example (8) where the content of a mixed statement in subjunctive mood precedes the cue and source and the content is referred to by a pronoun.

(8) Dieser Aufdruck sei “kein Wegwerfdatum [...]”. Das sagte [...] Ilse Aigner [...]

Overall, the evaluation showed that an approach based on dependency relations performs better than lexical pattern matching. We believe that this is mostly due to the power of dependency relations being able to capture discontinuities that often occur in German sentences.

5.4 Evaluating Precision of Statement Extraction

As fully annotating a complete data set with statements to measure precision and recall was not feasible, we ran the system on the entire NSA data set described above and focus on evaluating precision of statement extraction. While recall can

		cue			*source*			*content*		
		Prec.	Rec.	F_1	Prec.	Rec.	F_1	Prec.	Rec.	F_1
direct	Ploch (2015)	91.4	67.9	77.9	79.1	67.2	72.7	89.0	89.5	89.2
	our	91.6	86.3	**88.9**	79.1	75.0	**77.0**	96.4	93.4	**94.9**
indirect	Ploch (2015)	98.9	81.5	89.3	85.2	59.6	70.1	74.7	78.2	76.4
	our	90.3	89.9	**90.1**	77.1	76.0	**76.5**	83.2	88.6	**85.8**
mixed	Ploch (2015)	85.2	76.7	80.7	72.2	65.3	68.8	91.3	50.5	65.0
	our	87.1	81.3	**84.1**	75.0	69.1	**71.9**	92.6	77.1	**84.1**

Table 2: Evaluation of token-based precision and recall. The gray entries are the results reported in Ploch (2015).

	direct	indirect	mixed	all
statements	2415	3701	928	7044
persons	43.1%	40.9%	39.8%	41.5%
targets	8.2%	10.7%	9.3%	9.7%
time stamp	14.4%	15.8%	17.0%	15.5%
precision_{50}	97.2%	72.1%	81.8%	82.4%

Table 3: Data set statistics of the entire NSA data set as well as precision of extracted statements for 50 randomly sampled documents.

be compensated with redundancy in the data set, extracting erroneous statements should be avoided. Table 3 shows how many statements were extracted, as well as the fraction of statements for which persons (named entities of type PERSON) as sources, targets and explicit timestamps could be extracted. The numbers support our hypothesis that it is worth integrating the timestamp and target into the extraction process as, for instance, 17% of all mixed statements encode an explicit timestamp.

To measure how many of the extracted statements are actually valid, we randomly sampled 50 documents from the NSA data set and checked how many of the extracted statements represent true statements, regarding overlapping annotations as correct. The resulting precision for each statement type is given in Table 3. Overall, over 82% of all extracted statements are correct, thus showing that our system achieves high precision combined with a higher coverage due to a significantly enhanced lexicon of utterance verbs.

6 Conclusion and Ongoing Work

This paper presented an approach to extract statements from unstructured German news articles. Our temporally-aware definition of statements allows for creating and exploring a timeline of statements over time. We presented a two-stage approach consisting of a machine learning-based system to extract direct statements and a heuristic component based on a large lexicon of utterance verbs and corresponding dependency relations to extract statement components. Comparing our approach to the only other existing system for quotation extraction in German revealed that dependency relations are better suited to extract quotes and statements than lexical patterns. In addition, our extended database of statement cues extends coverage of statements while simultaneously maintaining high precision.

We are currently working on a bootstrap implementation to realize sequence tagging for statement extraction despite the low number of training instances. The final version of our system will be published as an open source project.

Acknowledgments

We would like to thank Danuta Ploch for sharing the annotated data set of quotes with us, making it possible to compare our system to hers. In addition, we thank the anonymous reviewers for their helpful remarks and suggestions.

References

Joachim Bingel and Thomas Haider. 2014. Named Entity Tagging a Very Large Unbalanced Corpus: Training and Evaluating NE Classifiers. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Éric de La Clergerie, Benoît Sagot, Rosa Stern, Pascal Denis, Gaëlle Recourcé, and Victor Mignot. 2011. Extracting and Visualizing Quotations from News Wires. In Zygmunt Vetulani, editor, *Human Language Technology. Challenges for Computer Science and Linguistics*, Lecture Notes in Computer Science, pages 522–532. Springer.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Birgit Hamp and Helmut Feldweg. 1997. Germanet – a Lexical-Semantic Net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Manfred Klenner and Don Tuggener. 2011. An Incremental Entity-Mention Model for Coreference Resolution with Restrictive Antecedent Accessibility. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2011)*, pages 178–185.

Heike Klüver. 2009. Measuring Interest Group Influence Using Quantitative Text Analysis. *European Union Politics*, 10(4):535–549.

Ralf Krestel, Sabine Bergler, and Ren Witte. 2008. Minding the Source: Automatic Tagging of Reported Speech in Newspaper Articles. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Tim O'Keefe. 2014. *Extracting and Attributing Quotes in Text and Assessing them as Opinions*. Ph.D. thesis, University of Sydney.

Silvia Pareti, Timothy O'Keefe, Ioannis Konstas, James R Curran, and Irena Koprinska. 2013. Automatically Detecting and Attributing Indirect Quotations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 989–999.

Silvia Pareti. 2012. A Database of Attribution Relations. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.

Danuta Ploch. 2015. Intelligent News Aggregator for German with Sentiment Analysis. In *Smart Information Systems*, pages 5–46. Springer.

Bruno Pouliquen, Ralf Steinberger, and Clive Best. 2007. Automatic Detection of Quotations in Multilingual News. In *Proceedings of Recent Advances in Natural Language Processing*, pages 487–492.

Luis Sarmento, Sergio Nunes, and E Oliveira. 2009. Automatic Extraction of Quotes and Topics from News Feeds. In *4th Doctoral Symposium on Informatics Engineering*.

Helmut Schmid. 1999. Improvements in Part-of-Speech Tagging with an Application to German. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, volume 11 of *Text, Speech and Language Technology*, pages 13–25. Springer Netherlands.

Rico Sennrich, Gerold Schneider, Martin Volk, and Martin Warin. 2009. A new hybrid dependency parser for german. In *Proceedings of the German Society for Computational Linguistics and Language Technology*, pages 115–124.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298.

Wouter Van Atteveldt, Jan Kleinnijenhuis, and Nel Ruigrok. 2008. Parsing, semantic networks, and political authority using syntactic analysis to extract semantic relations from dutch newspaper articles. *Political Analysis*, 16(4):428–446.

Andrea Zielinski and Christian Simon. 2009. Morphisto – An Open Source Morphological Analyzer for German. In *Proceedings of the 2009 Conference on Finite-State Methods and Natural Language Processing: Post-proceedings of the 7th International Workshop FSMNLP 2008*, pages 224–231.