LawStats – Large-Scale German Court Decision Evaluation Using Web Service Classifiers

Eugen Ruppert, Dirk Hartung, Phillip Sittig, Tjorben Gschwander, Lennart Rönneburg, Tobias Killing, Chris Biemann

LawStats – Large-scale German Court Decision Evaluation using Web Service Classifiers

Eugen Ruppert[1], Dirk Hartung[2], Phillip Sittig[1], Tjorben Gschwander[1], Lennart Rönneburg[1], Tobias Killing[1], and Chris Biemann[1]

[1] LT Group, MIN Faculty, Dept. of Computer Science, Universität Hamburg, Germany
{ruppert,5sittig,4gschwan,6roenneb,6killing,biemann}@informatik.uni-hamburg.de
https://lt.informatik.uni-hamburg.de/
[2] Bucerius Law School, Hamburg, Germany
dirk.hartung@law-school.de
https://www.law-school.de

Abstract. *LawStats* provides quantitative insights into court decisions from the Bundesgerichtshof – Federal Court of Justice (BGH), the Federal Court of Justice in Germany. Using Watson Web Services and approaches from Sentiment Analysis (SA), we can automatically classify the revision outcome and offer statistics on judges, senates, previous instances etc. via faceted search. These statistics are accessible through a open web interface to aid law professionals. With a clear focus on interpretability, users can not only explore statistics, but can also understand, which sentences in the decision are responsible for the machine's decision; links to the original texts provide more context. This is the first large-scale application of Machine Learning (ML) based Natural Language Processing (NLP) for German in the analysis of ordinary court decisions in Germany that we are aware of. We have analyzed over 50,000 court decisions and extracted the outcomes and relevant entities. The modular architecture of the application allows continuous improvements of the ML model as more annotations become available over time. The tool can provide a critical foundation for further quantitative research in the legal domain and can be used as a proof-of-concept for similar efforts.

Keywords: Law Domain · Web APIs · Text classification · Cognitive Services · Faceted Search.

1 Introduction

Legal professionals have become accustomed to the use of digital media and tools in their practice and Natural Language Processing (NLP) and Information Extraction (IE) generally offer a lot of potential benefits for many domains. However, their application to the legal domain is extremely limited to date. The legal profession needs exact and correct decisions. Thus, many struggle to accept Machine Learning (ML) techniques with a reported performance below 100%. In digital systems, rule-based IE is still dominant as it offers a high precision. But while rule-based systems allow you to get detailed insights in a document

collection, a meaningful understanding on document level is hardly achievable without ML methods. In addition, law is traditionally regarded as a normative and consensus-based science and only recently quantitative analysis and empirical methodology have become popular [5]. An aspiring school of thought classifies law as a complex adaptive system [13] and therefore deems technology absolutely necessary in order to tackle this complexity [14].

The project *LawStats* is the result of a collaboration between the Language Technology group at the University of Hamburg with the Bucerius Law School. Combining an entity extraction model trained by law students using the IBM Watson Knowledge Studio[3], and a tool for Aspect-Based Sentiment Analysis (ABSA) [15], the *LawStats* application analyzes court decisions and offers a faceted search interface to aid law practitioners. Users can explore the court decision database from the Bundesgerichtshof – Federal Court of Justice (BGH). The web application offers facetes for searching by judges, senates and lower courts like higher regional courts or district courts as well as by period of time. The user has the option to sort and search in all categories to look up information about court decisions and their components in a court decision database containing currently more than 50,000 court decisions. Users can upload and analyze additional court decision files to enlarge the database and test the application's analytical performance.

2 Related Work

The application of NLP tools and analysis on legal problems is a rather young area of research in Germany.[4] In the U.S., empirical and NLP-based analysis of court decisions has led to impressive results such as predictive modeling of Supreme Court decisions [8]. In Germany however, analysis of court decisions has so far been limited to special jurisdictions[5], albeit with impressive results if ML techniques were used [20]. Our procedure is not aimed at court decision predictions and thereby differs from *Waltl's* approach [20]. We are also not using any pre-existing meta-data but extract all of the entities from the document text using our ML model and the outcome classification is solely based on a text classifier. In these regards our approach substantially differs from previous academic ventures in both method and mere size of the corpus.

Corpus Linguistic approaches to law studies exist as well [19], most notably the *JuReKo* corpus [4], and enable statistical analysis and evaluation [9]. These works are related to our paper as they use NLP techniques to analyze court decisions, they differ, however, substantially from our work as for them ML techniques have not played an important role so far.

IE of document collections is often performed in journalism.[6] Journalists search for Named Entities (NEs) and their relations in a corpus. Then, faceted

[3] `https://www.ibm.com/watson/services/knowledge-studio/`
[4] For an introduction to computer assisted, linguistic research in law, see [18].
[5] For Labour Law, see [17].
[6] See, e.g. *Overview* (`https://www.overviewdocs.com/`) or *New/s/leak* [22].

search [16] is used instead of a simple keyword search, as it is more effective for professionals. Even though these frameworks offer impressive visualizations [22, 3], they cannot be used for document classification, as it would require training for particular domains. With a set law domain and expert annotators, we are able to perform polarity classification as well. Since the task is similar to SA [2, 11], we utilize a system originally developed for Sentiment Analysis (SA) and re-train it on our dataset annotations.

The presented system aids a Human in the Loop (HiL) working style, which is required for domains with 1) a lot of textual data and 2) the need for explainable ML classifications [6]. Professionals explore pre-annotated data using a faceted search interface and can add annotations. E.g., HiL is being employed in the biomedical domain [21], which needs an entity-centric access (bottom-up). In our use-case, we are concentrating on revision outcomes (top-down classification), that can be explored by different meta information.

3 Document Processing

3.1 Pre-processing

We perform two pre-processing steps to enhance the quality of the training data, which translates into better performance of the resulting ML models. In a normalization step, we replace abbreviations and inconsistently formatted expressions with a standardized form to reduce sparsity in the model. This step is necessary as annotator time is limited and we are striving for a high recall in IE. Note that we perform preprocessing on the annotation set as well as on every other document that later enters the system to ensure consistency.

The second preprocessing step is to replace all dots that are not full stops. Since the Watson Knowledge Studio (WKS) has difficulties with German sentence splitting and over-segments document on abbreviation dots (such as *bzw.*), we use our own sentence splitter and replace all non-sentence-end dots with underscores. This is necessary since we heavily build on the notion of a sentence in our setup and annotation in WKS is currently only possible within sentence boundaries.

3.2 Information Extraction

We extract and store the information from the BGH decisions. First, we analyze the document to determine the decision outcome, i.e. whether the revision was successful or rejected. Here, we make use of the particular structure of court decisions, as the operative provisions of decision are typically set at the beginning of the document. To determine the verdict decision, we classify the first ten sentences of a decision and use the one with the highest confidence score as the indicator for the outcome.

Additionally, we extract the entities *Gericht (court), Richter (judge), Aktenzeichen (docket number)* and dates from the text. These are sorted to determine

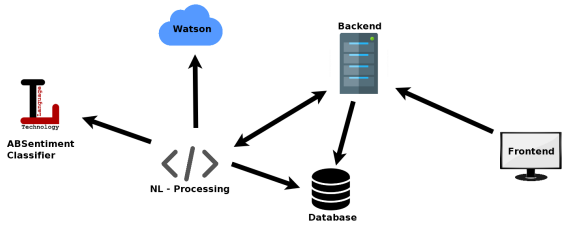

Fig. 1. System Architecture of *LawStats*

the relevant docket number, the correct procedural process and the temporal sequence. To determine the facetes for the search interface, we combine these with the decision outcome. WKS is used to train the NE extraction model and to generate the training data for the outcome model in one annotation step (see Section 5.1).

4 System Architecture

Overview The architecture of the application (Figure 1) consists of a front-end website and a back-end web server using Spring Boot and Spring Data. Document storage is performed by an Apache Solr instance. For text analysis, we use a Java API[7] to send and receive data from Watson Natural Language Understanding (NLU) in order to extract relevant entities from the court decisions. The outcome of the decisions is determined by a text classifier (see Section 5.2 for details).

Data Flow The data flow is presented in Figure 2. Once the PDF verdict document is uploaded, the document text is extracted and a normalization of sentence boundaries and dates is performed. Then, we send the document to the Watson NLU API, while at the same time analyzing the verdict decision. After analysis, the verdict document is constructed from both sources and stored in the Solr index.

[7] `https://www.ibm.com/watson/developercloud/natural-language-understanding/`

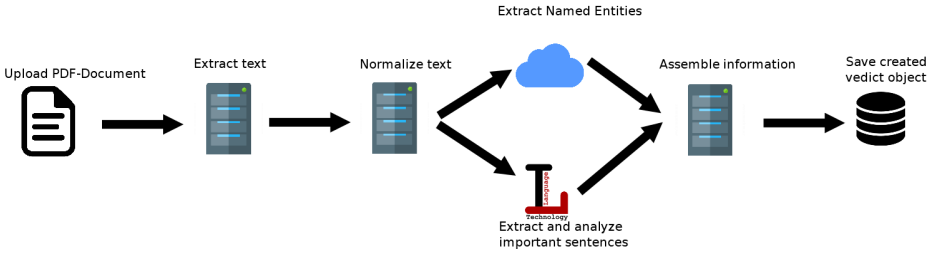

Fig. 2. Data flow pipeline

Table 1. Evaluation of Entity Recognition

Entity	Precision	Recall	F_1
Docket Number	0.99	0.97	0.98
Date	0.99	0.91	0.95
Court	0.92	0.68	0.78
Judge	0.95	0.95	0.95

5 Machine Learning

5.1 Named Entity Recognition (NER)

For NER, we use the Watson NLU API with a custom model. Internally, WKS employs the Statistical Information and Relation Extraction (SIRE)[8] classifier for sequential annotation and extraction of entities. It works in a similar way to a standard Conditional Random Field (CRF) [10] by employing symbolic feature combinations.

Annotation: Watson Knowledge Studio We define two different entity sets to be annotated by our team of seven domain experts. The first set contains all entities listed above (see Section 3.2). As courts and judges are finite and docket numbers and dates follow a definable pattern, we use dictionaries and regular expressions for pre-annotation. Our annotators have to correct false positives, limit annotations to relevant entities (i.e. not all courts, but only those, who were part of the procedural process) and annotate irregular mentions. Only annotations remaining after this manual step were used for training. Further, annotators identify the phrases used to indicate the outcome of the case. This task is done without pre-annotation. Our annotators could perform both tasks − correction and annotation − in one single pass.

In total, 1850 court decisions were annotated. The decisions were randomly sampled on the corpus. We set the Inter Annotator Agreement (IAA) threshold at 0.8[9] and have 20 % of all documents annotated by at least two different annotators. Before training the entity extraction model and deploying it to NLU, we remove the phrase-based outcome expressions from the training data to avoid confusing the sequential classifier.

Evaluation We use the WKS performance tool with a training set of 1260 documents, a dev set of 414 documents and a test set of 126 documents. The results in Table 1 show that the results are generally reliable with the exception of the extraction of court names, as their recall is only at 0.68. Since the document text

[8] `https://www.ibm.com/blogs/insights-on-business/government/relationship-extraction/`

[9] Only documents with high-quality annotations are chosen for ML training.

is normalized, dates and docket numbers can be identified with a pattern feature extractor. Additionally, they mostly occur in very confined contexts, where they are preceded by a few different keywords (e.g. "Aktenzeichen"). Further investigation has shown that courts appear in two entirely different functions in the decisions: as the deciding court (our target) and as lists of courts involved in previous relevant jurisprudence. This problem could be solved by limiting the IE to particular sections of the decisions such as the beginning or the very end.

5.2 Revision Outcome Classification

To evaluate the revision outcome of a court decision, we classify single sentences into the classes "Revisionserfolg" (revision successful), "Revisionsmisserfolg" (revision not successful) and "irrelevant". As described in Section 3.2, we take the first ten sentences of a decision, classify them independently and use the classification with the highest confidence score as the evaluation of the whole document. Here, we use an open-source text classification framework for German [15][10].

Annotation and Training Training data is obtained from the WKS annotations. We extract the annotated sentences as well as a random set of irrelevant sentences and train a multi-class SVM [1] classifier. For the feature set, we compute TF-IDF (Term Frequency Inverse Document Frequency) scores and word embeddings [12] on an in-domain revision corpus. The corpus contains all BGH court decisions available online. We build a feature vector based on the TF-IDF values and concatenate it with the averaged word vectors in a sentence. Furthermore, we induce features on the training data. We obtain a list of 30 highest-scoring (TF-IDF) terms per label (positive, negative, irrelevant) and add the relative frequencies of these terms to the feature vector. The training data consists of 2,200 labeled sentences. We use a balanced ratio of sentences for the two classes of successful/non-successful revision and use twice as much of irrelevant sentences for training. For testing, we use 550 sentences.

Table 2. Sentence-level evaluation of revision outcomes

Classifier	Precision	Recall	F-score
Majority class baseline	.46	.46	.46
LT-ABSA out-domain	.71	.70	.70
LT-ABSA in-domain	.91	.91	.91

Evaluation A simple baseline of choosing the majority class (irrelevant) scores 0.46 F_1. When we train the classifier on a standard out-domain feature set[11], we reach 0.70 F-score. By pre-training the TF-IDF vectors and the word2vec model on the in-domain collection of revision decisions, we reach a score of 0.91

[10] `https://github.com/uhh-lt/LT-ABSA`

[11] A news corpus is used for TF-IDF estimation, off-the-shelf German embeddings.

Table 3. Document-level evaluation of revision outcomes

Annotation Set	Correct	Wrong	Irrelevant
Set 1	85	12	3
Set 2	88	11	1
Overall Percentage	.87	.12	.02

(see Table 2). Error analysis shows that the major factor limiting the performance is the strong similarity between sentences indicating a successful and an unsuccessful revision. In most documents, the long sentences follow a rigid structure. Variation in the expression of the final outcome requires additional training data. Especially the edge cases (partially successful) show a lot of variation in the verdict. Since we classify on document-level, we have performed a document-level evaluation of the revision outcomes to verify that our sentence extraction approach works as expected. Two expert annotators annotated 100 documents each. The possible error cases were wrong polarity (successful/not successful) and when the classifier picked an irrelevant sentence as the decision-bearing sentence. Results are presented in Table 3.

With a precision of 0.87, we obtain a comparable performance as on the sentence level. Furthermore, the document selection features the same distribution as the training set (Fischer's test $p < 0.0001$), making it a representative sample of the complete collection. Error analysis of the incorrectly classified documents shows an even distribution. 12 documents were wrongly classified as "not successful" versus 11 "successful". About a quarter of the wrongly classified documents are partly misclassified. E.g. the revision was partially successful but classified as "not successful". For training, we had added the partly successful class to the positive "successful" class. In about 2 % of the documents, the wrong sentence is selected by the classifier. These sentences are often short phrases containing judge names; the classifier learned their co-occurrence in training data. This could be alleviated by masking entities in this classification task.

6 User Interface

The publicly available web application can be divided into two main components: a web page where the user is able to upload his locally saved revisions and inspect them, and a section to filter and examine the existing database of revisions. On the upload page, external PDF revisions can be uploaded and analyzed. After the file has been analyzed, the result is added to the database and the user is redirected to a result page.

The application allows faceted search on metadata and automatically extracted information in the document collection. The user can search for judges, senates, the corresponding "Oberlandesgericht" (higher regional court equiv.), "Landesgericht" (state court equiv.), or "Amtsgericht" (district court equiv.) decisions as well as the docket number (see Figure 3). To assess diachronic developments of revision outcomes, users can search for a timespan in which the

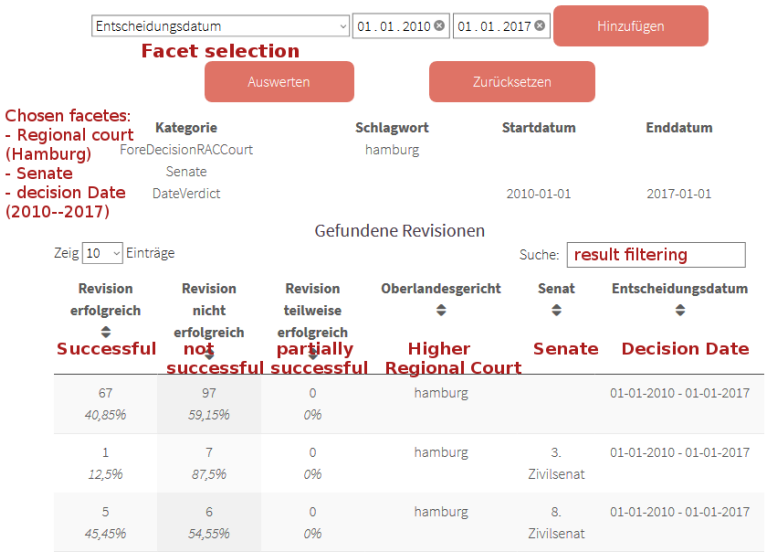

Fig. 3. Faceted search showing revision outcomes of the Higher Regional Court (OLG) Hamburg, faceted by the deciding senate; time range 2010–2017.

revisions or their respective previous court decisions were decided. To enable exploratory searches and comparison of verdict decisions by facet, facetes can be selected without query terms. Then, the application returns e.g. revision outcome statistics for all judges, courts, etc. Combinations of fields can be used here as well. The results page contains all extracted information about a decision such as courts, judges, etc. of the verdict file. Additionally, the page contains the classified revision outcome, the confidence score, and the sentence that determined the evaluation.

7 Conclusion

In this application paper, we have presented an application to access a large-scale corpus of BGH decisions, which is explorable by law professionals and publicly available online.[12] We have demonstrated that the most interesting part of a decision – the result – can be quite reliably determined using ML techniques for very large corpora of decisions. In future work, we would like to more tightly integrate the loop of annotation, model update and classification in order to enable a setup, in which the model can continuously improve on the basis of user's corrections in a human-in-the-loop setup. We plan to verify that the results are solid and helpful and ensure that our system aids professionals by reliable classification [7].

[12] `http://ltdemos.informatik.uni-hamburg.de/lawstats/` – Source code is available under a permissive license at `https://github.com/Kirikaku/LawStats`

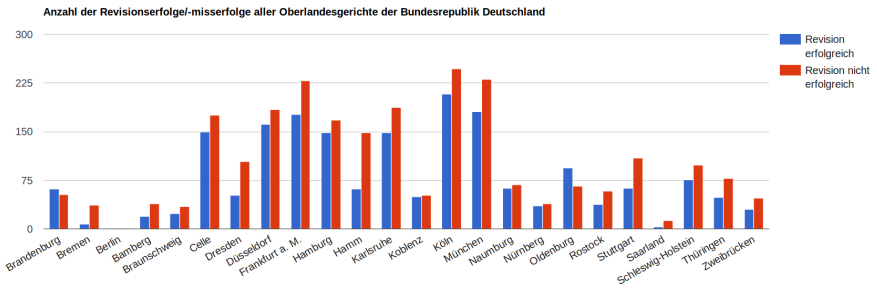

Fig. 4. Statistics, overview of successful vs. unsuccessful revisions per originating court.

While techniques in this work are rather standard, the value of this work lies in enabling a new field of application: an immense genuine added value from this application could be created with a thorough statistical analysis of factors correlating with success in front of the Federal Supreme Court. For this purpose, the quality of the entity extraction and the classification ought to be improved by different approaches and additional training. But even already now, the data set compiled using this application can be structured and analyzed profoundly by interdisciplinary teams. Both the confirmation of known influences like procedure types and yet unknown factors, e.g. duration of proceedings or geographical origin of the cases, would be an interesting starting point for substantial unprecedented large-scale legal analysis.

Acknowledgements

We would like to thank Ming-Hao Bobby Wu and Tim Fischer for their help in document conversion. We also would like to thank International Business Machines Corporation (IBM) for their ongoing support of the project.

References

1. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. pp. 144–152. COLT '92, ACM, New York, NY, USA (1992)
2. Cambria, E., Schuller, B., Xia, Y., Havasi, C.: New avenues in opinion mining and sentiment analysis. IEEE Intelligent Systems **28**(2), 15–21 (2013)
3. Dörk, M., Riche, N.H., Ramos, G., Dumais, S.: Pivotpaths: Strolling through faceted information spaces. IEEE Transactions on Visualization and Computer Graphics **18**(12), 2709–2718 (2012)
4. Gauer, I., Hamann, H., Vogel, F.: Das juristische Referenzkorpus (JuReko) – Computergestützte Rechtslinguistik als empirischer Beitrag zu Gesetzgebung und Justiz. In: DHd 2016: Modellierung - Vernetzung - Visualisierung. pp. 129–131. Leipzig, Germany (2016)

5. Hamann, H.: Evidenzbasierte Jurisprudenz: Methoden empirischer Forschung und ihr Erkenntniswert für das Recht am Beispiel des Gesellschaftsrechts. Mohr Siebeck, Heidelberg, Germany (2014)
6. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? CoRR **abs/1712.09923** (2017)
7. Holzinger, K., Mak, K., Kieseberg, P., Holzinger, A.: Can we trust machine learning results? artificial intelligence in safety-critical decision support. ERCIM News **112**(1), 42–43 (2018)
8. Katz, D.M., Bommarito, M.J., Blackman, J.: A General Approach for Predicting the Behavior of the Supreme Court of the United States. PLoS ONE **12**(4) (2017)
9. Kuhn, F.: Zugänge zur Rechtssemantik, chap. Inhaltliche Erschließung von Rechtsdokumenten auf Grundlage von Automaten. Walter de Gruyter, Berlin/New York (2015)
10. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the ICML 2001. pp. 282–289. Williamstown, MA, USA (2001)
11. Liu, B.: Sentiment analysis and subjectivity. Handbook of natural language processing **2**, 627–666 (2010)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Workshop at International Conference on Learning Representations (ICLR). pp. 1310–1318. Scottsdale, AZ, USA (2013)
13. Ruhl, J.B.: Law's Complexity - A Primer. Georgia State University Law Review **24**(4) (2008), `http://ssrn.com/abstract=1153514`
14. Ruhl, J.B., Katz, D.M., Bommarito, M.J.: Harnessing legal complexity. Science Magazine **355**(6332), 1377–1378 (2017). https://doi.org/10.1126/science.aag3013
15. Ruppert, E., Kumar, A., Biemann, C.: LT-ABSA: An extensible open-source system for document-level and aspect-based sentiment analysis. In: Proceedings of the GSCL GermEval Shared Task on Aspect-based Sentiment in Social Media Customer Feedback. pp. 55–60. Berlin, Germany (2017)
16. Tunkelang, D.: Faceted search. Synthesis Lectures on Information Concepts, Retrieval, and Services **1**(1), 1–80 (2009)
17. Vogel, F., Christensen, R., Pötters, S.: Richterrecht der Arbeit – empirisch untersucht. Möglichkeiten und Grenzen computergestützter Textanalyse am Beispiel des Arbeitnehmerbegriffs. Duncker & Humblot, Berlin (2015)
18. Vogel, F., Hamann, H., Gauer, J.: Computerassisted legal linguistics: Corpus analysis as a new tool for legal studies (2017). https://doi.org/10.1111/lsi.12305
19. Vogel, F.: The pragmatic turn in law. Inference and Interpretation, chap. Calculating legal meanings? Drawbacks and opportunities of corpus assisted legal linguistics to make the law (more) explicit. Mouton de Gruyter, New York, Boston (2017)
20. Waltl, B., Bonczek, G., Scepankova, E., Landthaler, J., Matthes, F.: Predicting the Outcome of Appeal Decisions in Germanys Tax Law. In: International Federation for Information Processing (IFIP): Policy Modeling and Policy Informatics. St. Petersburg, Russia (2017)
21. Yimam, S.M., Remus, S., Panchenko, A., Holzinger, A., Biemann, C.: Entity-centric information access with the human-in-the-loop for the biomedical domains. In: Biomedical NLP Workshop associated with RANLP 2017. pp. 42–48. Varna, Bulgaria (2016)
22. Yimam, S., Ulrich, H., von Landesberger, T., Rosenbach, M., Regneri, M., Panchenko, A., Lehmann, F., Fahrer, U., Biemann, C., Ballweg, K.: new/s/leak – information extraction and visualization for an investigative data journalists. In: ACL 2016 Demo Session. pp. 163–168. Berlin, Germany (2016)