Contribution to a thematic issue

Hans Ole Hatzel*, Haimo Stiemer, Chris Biemann and Evelyn Gius

Machine learning in computational literary studies

https://doi.org/10.1515/itit-2023-0041
Received May 21, 2023; accepted August 8, 2023;
published online August 25, 2023

Abstract: In this article, we provide an overview of machine learning as it is applied in computational literary studies, the field of computational analysis of literary texts and literature related phenomena. We survey a number of scientific publications for the machine learning methodology the scholars used and explain concepts of machine learning and natural language processing while discussing our findings. We establish that besides transformer-based language models, researchers still make frequent use of more traditional, feature-based machine learning approaches; possible reasons for this are to be found in the challenging application of modern methods to the literature domain and in the more transparent nature of traditional approaches. We shed light on how machine learning-based approaches are integrated into a research process, which often proceeds primarily from the non-quantitative, interpretative approaches of non-digital literary studies. Finally, we conclude that the application of large language models in the computational literary studies domain may simplify the application of machine learning methodology going forward, if adequate approaches for the analysis of literary texts are found.

Keywords: computational literary studies; language models; machine learning; natural language processing; transformers

ACM CCS: Artificial intelligence → Natural language processing

***Corresponding author: Hans Ole Hatzel**, Department of Informatics, Universität Hamburg, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany, E-mail: hans.ole.hatzel@uni-hamburg.de
Haimo Stiemer and Evelyn Gius, Technical University of Darmstadt, Institute of Linguistics and Literary Studies, Residenzschloss 1, 64283 Darmstadt, Germany, E-mail: stiemer@linglit.tu-darmstadt.de (H. Stiemer), evelyn.gius@tu-darmstadt.de (E. Gius)
Chris Biemann, Department of Informatics, Universität Hamburg, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany, E-mail: chris.biemann@uni-hamburg.de

1 Introduction

Literary studies is the academic field concerned with the analysis of literary texts and literature related artifacts. The focus is often on texts that were written with artistic intent, i.e. that deviate more or less from everyday language. In a broader sense, the subject of the discipline is also the reception of literature, the analysis of general trends in texts as well as the conditions of literary production. As such the field covers a wide range of text forms, from dramas to novels and poetry and can also rely on additional data. Computational literary studies (henceforth CLS) is the discipline of using computational methods for this analysis; this is by no means limited to machine-learning-based methods but can, for example, start from relatively simple word-count statistics or be based mainly on manual annotation.

In this survey, we will approach the current state of the discipline of CLS from a methodological perspective, focusing on machine learning. Before us, Helling et al. [1] have characterized the methodology used in CLS on the basis of researcher interviews. Recently a survey approaching the discipline from the perspective of basic concepts in literary studies, rather than the methodology, was released [2].

CLS can be considered a subfield of Digital Humanities, in this paper we focus exclusively on CLS, which focus on a comparatively narrow domain. The field of CLS has gained traction in the recent past. For example, in the most recent installment of the Computational Humanities Research Conference (CHR) that has been running since 2020, around half of the contributions dealt with literary texts and can therefore be considered CLS. In the German-speaking area, CLS has received a large push due to a priority program (SPP 2207) by the German Research Foundation (DFG). Further, in 2022 the JCLS was created as the first CLS journal since DLS (digital literary studies), which only published one issue overall, in 2016.

While not without contention [3–5], CLS has potentially much to contribute to literary studies. It can enable literary scholars to view their work from a new perspective, opening up new potential avenues of research. With the use of automated methods it is possible to scale the subject of research

to a large corpus of literary works. This has received rather little attention so far, simply because literary researchers traditionally focus on a limited well-researched set of so-called canonical texts [6]. The vast body of literary works, also referred to as "the great unread", can help inform concepts of literary theory as well as put them to empirical verification. In the field of CLS, automated methods are used to extract specific information from individual texts. From a macro perspective, salient features, patterns, and trends can then be explored in large text corpora, which often is referred to as "distant reading", a term coined, even though originally with a different meaning referring to a kind of meta-analysis, by Moretti [7]. From the macro level, it is also possible to zoom in on individual texts in order to analyze them on a reading-based level, called "close reading". The application of "distant" and "close reading" in conjunction, varying the level of analysis is in turn called "scalable reading" [8, 9].

Due to the application of scalable reading, balancing formalization and interpretation present a major challenge for CLS [10]. The upsides of CLS despite these challenges, especially in comparison to traditional literary studies, are perhaps best exemplified by the application of stylometry. Namely, these advantages are the ability to include a much larger amount of text in the analysis, to provide a new perspective on the text data, e.g. the opportunity to discover patterns that are not easily picked up by humans due to the sheer size of the data. To illustrate this, we can consider the study of style, which has always been a contested field in non-computational literary studies, as it is primarily concerned with aesthetic value judgements. Computational stylometry, by contrast, compares texts or text passages stylistically on the basis of statistical distributions of tokens or token sequences (for details see Section 3.7). Stylometry can be used in questions of authorship attribution. For example, with the help of stylometric analysis, Joanne K. Rowling was revealed as the author of "The Cuckoo's Calling", a novel published under the pseudonym Robert Galbraith [11]. Stylometry has also been used to approach the difficult question of literary quality; van Cranenburgh et al. [12], for example, compared the texts of the bestselling author Stephen King, who is classified as light literature according to conventional opinion, with texts from National Book Award-winning authors (usually labeled "high brow literatur"). Their results give King's "Dark Tower" books a high literariness and thus offer a differentiated view of the author's work. At the same time, the results show the effectiveness of a stylometry in quantifying literariness.

Work by Matthew Lee Jockers can help provide an another example of the value of CLS, specifically in the context of literary histography. He was able to, among other things, show with a metadata analysis that the state of research on Irish-American fiction had made false conclusions on the basis of arbitrarily selected examples [13]. Jockers compiled an extensive collection of novels by Irish-born authors published in the US and supplemented this dataset with information on the authors' gender and the stories' geographical settings. Using visualizations and statistics, he was able to show that where previous research had seen the focus of settings in these authors to lie on the metropolitan US East Coast, there was a large number of female authors publishing novels set in rural West Coast areas.

1.1 Needs of CLS

As with all subject-specific research interests in the broader field of digital humanities, a key challenge of CLS is to operationalize conventional literary studies categories and theories for computational analysis. It is therefore necessary to model categories that may, at first glance, seem abstract in such a way that they can be automatically inferred from the text surface. At the same time, many CLS approaches need to make the results of studies supported by machine language processing available for interpretation in literary studies. In line with C. P. Snow's thesis of the two diametrically opposed scientific cultures [14], it can be claimed that the interpretation of numerical, quantitative results brings together "explaining" as the central goal of the natural sciences and "understanding" as the goal of the humanities. The question of interpretability can, accordingly, also be understood more broadly as a question about the "translatability" of the results generated by computational methods into the field of traditional, non-computational literary theory. To bridge the gap between the two cultures of research, not only do the results have to be understandable but so does the process by which they were obtained. Here we see an interesting link to the field of machine learning, where explaining, understanding, and interpreting model outputs are all current fields of research.

1.2 CLS as applied NLP

The computational text analysis in CLS can be considered an application domain of natural language processing (NLP). Traditionally NLP, in conjunction with adjacent fields like computational linguistics, has been focused on analyzing all aspects of texts from the small-scale linguistic phenomena such as parts of speech[1] to semantics of entire works.

1 The word class of individual tokens, e.g. verb or noun.

Literature presents a very challenging domain due to the complexity and in the case of prose and drama often also length of such texts, especially since many existing approaches were primarily tested on news data. For example, the popular OntoNotes [15] dataset encompasses news texts in multiple languages annotated with a wide range of linguistic phenomena. That is to say, CLS can offer a test bed for challenging tasks of natural language processing while providing the tools needed to facilitate literary analysis. One example of literary works being used in their capacity as challenging NLP problems is in the domain of abstractive text summarization. Wu et al. [16] used modern language processing models to summarize books, solving the length problem by recursively summing up previous summaries starting from small sections of text.

2 The (computational) literary studies research process

Literary studies can be conducted using a large variety of methodologies, perhaps most prominent is the application of hermeneutic methods. Hermeneutics in a broader sense basically is an umbrella term for methodologies directed towards the interpretation and explanation of a text, i.e. the reconstruction of the meaning represented by the text [17, 18]. As such, the traditional research process involves the "close reading" of individual texts.

With the introduction of computational methods, these analytical methods can be further extended using corpus-linguistic methods. For example, measuring a variety of word-count-related phenomena such as the lexical diversity of texts, which is, generally speaking, reflected in the relative number unique words in a text [19]. Going beyond methods that are traditionally associated with corpus linguistics, a wide range of automated processing techniques from the field of NLP, often involving machine learning techniques, can be used to extract information for subsequent analysis. While the process of CLS research is not standardized and a diverse set of research paths exist, we found a typical approach to the application of machine learning to emerge in our review. We will go on to describe this research process in detail. A comparable process was previously described by Pichler et al. [20]. Schöch et al. [2], in their survey on methodology in CLS also present a description of the typical research steps required for CLS, before going on to approach the field from the perspective of multiple literary research problems (e.g. authorship attribution and gender analysis).

The prototypical formulation of the CLS research process that we propose, as seen in Figure 1, starts with theoretical literary work. After identifying the research question or more broadly an area of interest (1), the researcher conducts literature research (2), identifying which existing concepts are relevant to the question and could be adapted (3). Subsequently, the concept is operationalized (4) into either an annotation guideline or a clear set of rules. At this point, we identify two distinct approaches, the first (5) is the use of rule-based systems on top of existing machine learning infrastructure, this may for example be the application of part-of-speech or token-specific rules or alternatively the application of an existing sentiment classifier. Alternatively annotations are created on the basis of the operationalization in the form of annotation guidelines (6). If annotations were created, a machine learning system is trained on them (7). Regardless of which of the two options is picked, the existing annotations are now validated (8) — with the potential of going back to the annotation process or even refining the operationalization — and subsequently scaled to a larger corpus (9). On the edge of theoretical work, in

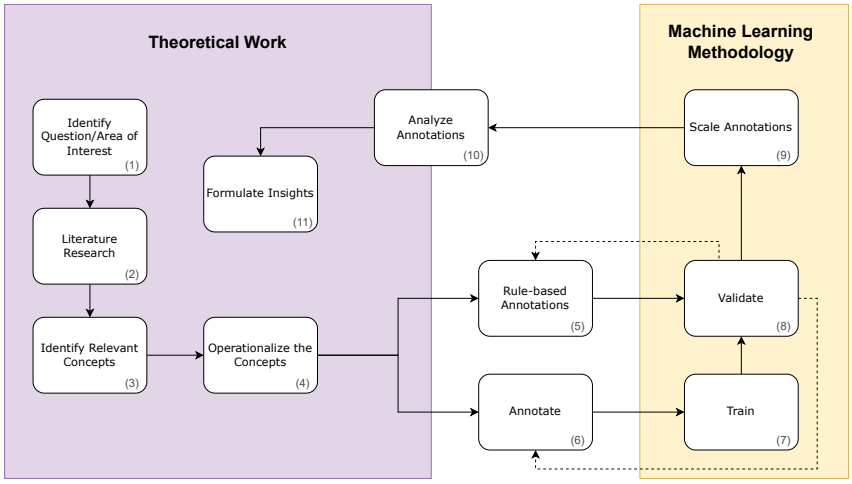

Figure 1: The prototypical research process when applying machine learning in CLS.

combination with quantitative analysis, the scaled annotations are analyzed using a variety of tools from statistics to visualizations (10). The results of this analysis can in turn be used to formulate insights of literary studies relating to the question at hand.

For this survey, the theoretical background of literary scholars' work is considered out of scope, instead, we will only detail theoretical concepts where required for the understanding of the machine learning methodology.

2.1 From operationalization to automation

Concepts of literary studies are not typically well-defined in a formal sense and rely on interpretation when applying them to an existing text. As a result, in non-digital literary studies, the focus is not on objective research findings, but on the "intersubjective comprehensibility" of the argumentation or interpretation [10], that is to say *verification* of a concept is done by forming a shared understanding of the subject. An operationalization specifies which aspects are to be considered in this process, it may also simplify a theoretical concept. One possible operationalization is an annotation guideline, sometimes also called a codebook.

Annotations in CLS are typically created either by domain experts (e.g. the researchers themselves) or trained annotators (e.g. student assistants), but not typically by untrained annotators such as crowd workers. Both document-level annotations and span-level annotations are common. Helling et al. [1] found the most used annotation tool among their surveyed researchers to be CATMA [21], their survey, however, only included researchers at German and Swiss Universities.

In Section 3, we will explore automation techniques in detail. Generally speaking though, in CLS, alongside modern neural-network architectures, a lot of more traditional machine learning approaches, like regressions or support vector machines (SVMs) are used; in these cases, some feature selection is typically required. Alongside these typically supervised approaches, unsupervised methods like clustering and topic modeling are also employed. Finally, not all of CLS relies on machine learning with many works instead using text mining, e.g. in the form of frequency analysis, to gain insights into the structure and content of texts.

3 Methods of machine learning

All CLS works making use of machine learning in some way can be placed on a continuum of technological innovation, from one extreme, using existing packaged software to the other, of innovating machine learning technologies. Those

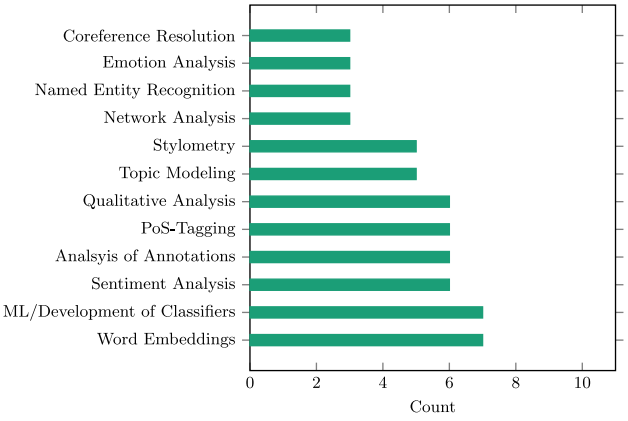

Figure 2: Methodologies used in the 11 projects of the SPP-CLS as collected by Helling et al. [1].

papers applying existing methodology provided by a software package do not even in all cases need to consider the details of the implementation as long as the method is well understood and a good match for the task at hand; for well-established packages, the output can be perfectly sufficient to answer research questions on a specific body of work. We will seek to explore both cases, but focus on methodologically innovative approaches.

The only quantitative overview of methods known to us is provided by Helling et al. [1] for the German CLS community. In a survey of researchers that were part of the DFG-funded priority program Computational Literary Studies,[2] the two methods or analysis tools most frequently reported to be used (by 7 out of 11 projects each) were word embeddings and machine learning classifiers, closely followed by sentiment analysis, analysis of annotated data, part-of-speech-tagging, and qualitative content analysis (by 6 out of 11 projects each). Further, topic modeling, and stylometry were reported to be used by 5 projects each. We visualize this data in Figure 2, omitting all methods that were reported to be used by less than three projects.

While this may give an initial overview of the methods, it is limited in that it is specific to the German community and in that results are based on the mere usage of the method without further context. It also does not focus on the specifics of machine learning methods.

Similarly, as mentioned in the introduction, Schöch et al. [2] explore applied methodologies in CLS from the perspective of specific fields of CLS, such as authorship attribution and genre analysis. In their introduction they distinguish between frequency analysis, searching (or retrieval)

2 https://dfg-spp-cls.github.io/.

methods and machine learning techniques (distinguishing between supervised and unsupervised approaches), subsequently detailing their use in the specific fields.

3.1 Our survey

To provide a broader, more concrete overview of the machine learning methods in use in the field of CLS, we select a number of scientific venues that are known for significant CLS contributions, with the aim of building a broad cross-section of the CLS community. While there is no clear boundary to what constitutes a machine learning method as compared to statistical analysis, we consider all regression methods to be machine learning techniques, whereas frequency analysis of texts and significance testing are considered statistical methods. We will discuss predominant methodologies in detail, but will not explore each individual method.

We only consider publications in English, which does exclude a variety of smaller language-specific or region-specific venues; works in English do, however, often have literature in other languages as their subject of study (e.g. [22–24]). Our selection includes the newly founded "Journal of Computational Literary" studies (JCLS), the Proceedings of the international "Digital Humanities" (DH) conference, as well as the SIGHUM Workshop "on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature", and the CHR "Computational Humanities Research" conference. The selection follows the goal of providing a cross-section of the community, with the Digital Humanities conference presenting the largest international conference on digital humanities and thereby ensuring geographic diversity. The CHR is an interdisciplinary conference focusing on all computational approaches in the humanities and receives submissions mostly from the European community. The SIGHUM workshop, on the other hand, is organized as part of the ACL and thereby connected to the NLP community. Lastly, the JCLS is the only journal in our lineup and unlike the other three is exclusively focused on CLS. We use the most recent issue of each publication, with the exception of SIGHUM, where the 2022 issue only features two CLS-focused entries, leading us to include the 2021 issue as well, as such methodological comparisons across venues, based on our survey, are potentially misleading due to the different timeframes. We cannot claim that our data is comprehensive and representative, yet it provides an overview that is not limited to a specific subcommunity.

Across all publications, we start with a selection of 215 papers, only 33 % of which we considered to feature CLS in our initial screening, based on both the titles and when in doubt, the contents of the papers. This ratio differs greatly depending on the publication. For example, all submissions in the JCLS journal concern CLS, while only about a quarter of all Digital Humanities conference "Long Paper" submissions were found to be CLS-specific. In the case of CHR, we found a surprising prevalence of literary studies at the conference, with 14 of 31 papers falling into the category of CLS. After this initial selection, we checked the papers for the application of machine learning techniques, narrowing our selection of 71 CLS-related papers down to 40, which we will consider in this overview. Importantly, we also removed stylometric analyses since they, with one exception [25] that we will discuss in the Section 3.7, do not apply methodologies of machine learning. Other work discarded in this step was based on statistical evaluation of word distributions, e.g. dispersion measures, as well as conceptual work discussing the CLS research process. Detailed numbers on the occurrences of CLS and machine learning specific works are listed in Table 1.

Our collected data on the papers' methodologies is presented in Table 2. The main machine learning methodologies are listed in the corresponding column, with the model name specifying the name of a pre-trained model, if applicable. The "LS Question/Topic" column denotes the question or topic of the paper with regard to literary studies. In some cases we did not identify a literary studies' question, as these heavily focus on the evaluation or improvement of technical methodologies instead [26–28]. For what we list as the method we only consider the actual decision-making method, e.g. when a text is classified using an SVM that operates on word embeddings, we list SVM as the methodology. The features that are used are tracked separately. The columns "Annotations" and "Rules" indicate which of these two categories the paper falls into: (1) approaches with rule-based processing, potentially with existing pre-trained models or (2) those training their own models and scaling their annotations to more texts. One example of (1), the

Table 1: We screened a total of 215 papers in our review of the CLS literature, narrowing our selection down to 40 after first removing all non-CLS related work and then all that did not include methods of machine learning.

	Number of papers			Percentage of papers	
	Total	CLS	CLS & ML	CLS of total	CLS & ML of CLS
JCLS	12	12	9	100.00 %	75.00 %
CHR	31	14	7	45.16 %	50.00 %
SIGHUM	36	12	10	33.33 %	83.33 %
DH	136	33	14	24.26 %	42.42 %
Total	215	71	40	33.02 %	56.34 %

Table 2: The 40 papers we reviewed are listed with their main machine learning method and additional details.

Author	Machine learning method	Model name	LS question/topic	Annotations	Rules	Features
			DH 2022			
Algee-Hewitt [32]	Linear discriminant analysis	–	Concepts vs objects	–	✓	–
Bonch-Osmolovskaya et al. [35]	Logistic regression	–	What are war diaries about?	✓	–	Tf-idf
Calvo Tello et al. [26]	Transformer	mBert	–	–	–	–
Camps et al. [36]	SVM, topic modeling	–	Song author features	✓	–	Chracter ngrams, lemmas, musical features
Ciotti [37]	K-means	–	Features of literary periods	–	–	LIWC vectors, MFW
Dennerlein et al. [38]	Transformer	gBert	Emotions in dramas	✓	–	–
Eder et al. [39]	SVM	–	Authorship attribution	–	–	–
Glass [40]	Transformer	USE, Bert	Adaptations of Robinson Crusoe	–	–	Tf-idf
Herrmann et al. [41]	–	–	Spatial entities	–	–	–
Ivanov [42]	SVM, MLP, random Forest	–	Concreteness as an author feature	–	–	Abstractness, character n-grams and more
Langlais et al. [43]	SVM	–	Genres in French fiction	–	–	Tf-idf
de la Rosa et al. [28]	Transformer	mT5, mByT5	–	✓	–	–
Schumacher [44]	CRF	–	Space in novels	✓	–	Named entities
Schumacher et al. [23]	Transformer, CRF	gBert	Gender in fiction	✓	–	
			SIGHUM 2022 & 2021			
Abdibayev et al. [31]	BiLSTM, CRF	–	Features of poetry	✓	✓	Phonemes
Cooper et al. [22]	Logistic regression, LDA	–	Storyteller characteristics	✓	–	Tf-idf
Karlińska et al. [29]	Transformers	LaBSE	Cities vs villages	–	✓	
Kunilovskaya et al. [45]	SVM	–	Translations vs original	✓	–	Dependency information
Schmidt et al. [46]	Transformer	c2f	Character networks	✓	–	–
Schmidt et al. [30]	Transformer, SVM, NB	Various German transformers	Emotions in dramas	✓	–	Bow
Schneider et al. [47]	Logistic regression	–	Chiasmus	✓	✓	Pos, dep, embeddings
Steg et al. [48]	Theil-Sen regressor, doc2vec	–	Narrative passages	✓	–	Concreteness, Tf-idf
Wöckener et al. [49]	Transformer, RNN	GPT-2	Phenomena of poetry	✓	–	
Xie et al. [24]	Transformer	Bert-base-Chinese	Adverbial markers	–	✓	Pos

Table 2: (continued)

Author	Machine learning method	Model name	LS question/topic	Annotations	Rules	Features
			CHR 2022			
Clérice [27]	GRU, BiLSTM, TextCNN	–	–	✓	–	–
Konle et al. [50]	Various	–	Plot models	–	–	Tf-idf, temporal graphs, various derived measures
Parigini et al. [51]	Transformer	mBert, Italian-xxl-cased	Dubitative passages	✓	–	–
Perri et al. [52]	GNN, GRL	–	Character interactions	–	–	–
Piper et al. [33]	Transformer	bookNLP	Which “things” are mentioned?	–	✓	WordNet categories
Zhang et al. [53]	Transformer	ECCO-BERT-Seq, bert-base-cased	Genre change	✓	–	Tf-idf
Zundert et al. [54]	Transformer	Multilingual USE	Are topics genres?	–	–	–
			JCLS 2022			
Abdibayev et al. [55]	Transformer	GPT-2, BERT, TransformerXL, XLNet	Features of limericks	–	–	–
Brottrager et al. [56]	SVM, XGBoost, transformer	–	Which works become canon?	✓	–	Distinctiveness, basic text features (like n-grams), complexity features
Du et al. [57]	SVM, NB, logistic regression, decision tree	–	Keyness of terms	✓	–	Tf-idf, Welch, Eta, Zeta, various others
Ehrmanntraut et al. [58]	Transformer	Paraphrase-XLM-Roberta, gBert, sentence encoders	Poem similarity	✓	–	–
Koolen et al. [59]	–	–	How does a book make you feel?	–	✓	Pos, lemma
Schröter et al. [60]	LDA	–	Literary concepts in topic models	–	–	–
Shin [61]	–	–	Sentiment towards “queer”	–	–	–
Völkl et al. [62]	LDA	–	Gender discourse	–	–	Filtered lemmatized tokens
Weimer et al. [63]	Logistic regression, decision tree	–	Literary comments	✓	–	Dependency, pos, sentiment

rule-based approach, can be found in the work by Karlińska et al. [29]. While they do not employ annotators themselves (outside of meta-data creation and validation) they instead use existing tools for named entity recognition, geographic information retrieval, and sentiment analysis. Using these tools, they analyze the sentiment towards cities as compared to other locations and conclude that, unlike suggested by previous work, cities actually are depicted more positively than villages and the countryside. While they note that this finding does need further analysis, this is an example of CLS potentially refuting a thesis previously set out. An example of (2), approaches that use their own annotations for training, can be found in the work on emotions in dramas by Schmidt et al. [30]. Some works fall into both categories, for example in the case of Abdibayev et al. [31], who use both annotations and rules for different aspects of their analysis. The column "Features" lists the features used in feature-based approaches. Although, in some cases, the list is very extensive with multiple feature sets being discussed and compared, in this case, we list them as "various".

In our survey, transformers are by far the most popular method, followed by SVMs and logistic regression (see Figure 3). While transformers, which we will explore in detail in Section 3.4, operate more or less directly on the input text, SVMs and logistic regression both require some sort of feature engineering. We have multiple cases in which Tf-idf (term frequency – inverse document frequency) is listed as a feature, this is typically calculated on the basis of individual tokens but not in all cases clearly specified. We will consider employed features in more detail in Section 3.5.

Overall we found just over half of the final selection of papers to use some sort of annotation-based setup (see Table 1), allowing for supervised learning. Others built rule-based systems on top of existing machine learning approaches [29, 32, 33]; for example Piper et al. [33] use an existing sense tag system (based on WordNet) to identify "things" and the category they belong to, finding that most things referred to in literature are "human-made and supportive" meaning such objects as "rooms, houses, doors, windows, […], roads, kitchens" and others.

Some of the techniques listed in the Table 2 are not explained in detail in this document, either because they are rarely used or because they are already very well established techniques. For reference, we expand the abbreviation of those that are not explained elsewhere here. In terms of model variants, RNN refers to recurrent neural networks with GRU referring to gated recurrent units, a simplified variant of the variant of the LSTM (long short-term memory), where BiLSTM refers to the LSTM's bidirectional variant. CRF refers to conditional random fields, a statistical model. LDA refers to Latent Dirichlet allocation a topic modeling approach, which is not to be confused with Linear Discriminant Analysis which we spell out in the one instance it is used. GRL refers to graph representation learning, NB to Naïve Bayes, and c2f to a specific coreference resolution system [34]. For the features, we use pos as short for part-of-speech, bow as short for bag-of-words and dep as short for dependency tree information.

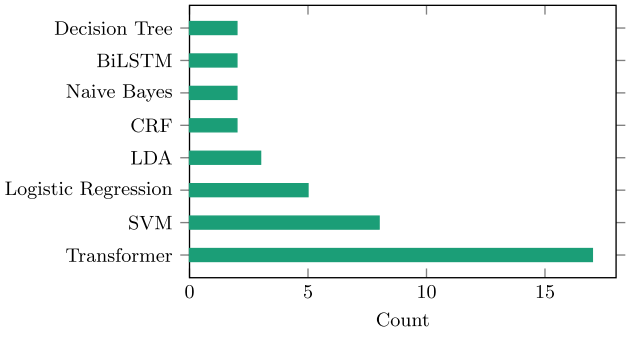

Figure 3: The distribution of methods in our literature review shows that transformers are by far the most popular method but still not used in even half the papers, demonstrating the diversity of methodology in CLS. We omit methods that only occured once.

3.2 Pre-processing

Recently, the NLP community has been slowly moving away from employing the so-called pipeline approach, where one language processing step builds on the previous ones towards end-to-end models such as transformers [64]. For example, a pipeline might consist of a tokenizer splitting the text into individual tokens, a part-of-speech tagger tagging each token, and a named entity tagger recognizing entities on the basis of the two other components' outputs. A trend away from this kind of architecture can also be observed in CLS with the adoption of transformers; pipeline-based approaches however seem to still be popular, as evidenced by a wide range of works making use of hand-crafted features for their classifiers. One of the most popular frameworks, albeit not specific to CLS, providing a range of preprocessing options is spaCy [65], employed by a multitude of authors in our survey [25, 29, 47, 50]. While the library provides high-level capabilities like coreference resolution, it can also be used for more basic preprocessing like sentence splitting and tokenization as well as dependency parsing and part-of-speech tagging. In terms of tools focused on the processing of literature, BookNLP[3] provides a pipeline-based approach to processing entire books (used by e.g.

3 https://github.com/booknlp/booknlp.

[48, 66]), incorporating such preprocessing tasks as named entity recognition, part-of-speech tagging (using SpaCy), and coreference resolution. Similarly, LLpro [67] provides such a pipeline approach for German language texts. The MONAPipe pipeline [68], integrates with spaCy and, in addition to some standard pre-processing steps, provides a host of literary analysis tools like event annotations, entity linking, coreference resolution, and speech attribution to name a few.

We see three main motivations for the CLS community still relying on pipeline-based approaches: (1) training data for CLS-specific problems often does not exist, meaning it either has to be time-consumingly annotated or a hybrid approach that relies on some form of rule-based inference may have to be used. Further, (2) as we discuss in detail in Section 3.4, existing language models are often not perfectly applicable to the literature domain. This fact potentially increases the viability of simpler approaches, for example on the basis of features extracted in an early pipeline step. Finally (3), feature-based methods can typically more easily be inspected with regard to their decision-making process, such as inspecting the importance of individual features, allowing researchers to interpret not only the models output but also the weight of features in its decision process.

3.3 Word embeddings

Word embeddings, in general, are vectors that represent the semantics of individual words by their position in space. Conceptually we distinguish between static word embeddings which represent each occurrence of a given word the same way and contextual word embeddings which represent each instance of a word differently, based on its context. Contextual word embeddings were, in the CLS literature we reviewed, typically produced by transformers. Overall, popular choices of static embeddings were GloVe, word2vec, and fastText. While GloVe [69] and word2vec [70] use slightly different techniques, they often perform similarly, fastText [71], on the other hand, has the added capability of processing sub-word tokens. Where any term that was not known during training to GloVe or word2vec can not be represented, fastText can represent unknown words by representing constituent character sequences individually.

A large number of works [32, 39, 47, 56] make use of static word embeddings in conjunction with classifiers, sometimes as one feature among many. In the simplest case, Algee-Hewitt [32] uses linear discriminant analysis on word embeddings to identify if a given noun refers to an object or a concept. The direct use of word embeddings and their distances as a means of analysis appears to be almost as common [41, 52, 61]. For example, Ehrmanntraut et al. [58] use GloVe and fastText embeddings to predict similarities of pairs of poems, comparing the results with human judgments. They find fastText embeddings to perform slightly better in their setup. Measuring accuracy in identifying which of two poems is overall more similar to a given anchor poem, the best fastText static embedding approach achieves scores of 0.66 without supervision and improves to 0.72 after supervised training of a siamese-network [72] that uses the existing fastText embeddings as input. Eder et al. [39] use artificially added document-specific tokens to explore the use of word embeddings in authorship attribution, with the idea that artificial tokens capture their context, which for static embeddings is true at training time on a corpus level, thereby representing the document or the author.

3.4 Transformer-based methods

Transformer-based architectures [73] have revolutionized the field of NLP in the last few years by drastically outperforming previous approaches. Recently generative models, also using transformer architectures, have demonstrated unprecedented text generation capabilities. The success of transformers as a neural-network architecture, even outside the domain of language processing, can be attributed to their attention mechanism. In contrast to previous recurrent neural network approaches, this allows the model to directly access the information of any combination of tokens at once, without relying on its own state, as a recurrent model would. The application of transformers to CLS does, however, bring a couple of challenges with it: for one, foundation models, which require vast computational resources, are typically primarily trained on data scraped from the internet, resulting in a domain mismatch. For example, only 16 % of GPT-3's training data is sourced from two book corpora. The BookCorpus, which was used for the training of a variety of foundation models including BERT and GPT variants, has been shown to exhibit a large bias in terms of genre distribution [74]. As the selection of texts for such models is not driven by literary studies' needs, the resulting models potentially perform poorly on the texts subject to analysis. Konle et al. [75] have shown that domain adaptation helps improving the performance of transformer models on downstream tasks in the domain of literary fiction. In another case, in the domain of German dramas, domain adaption was found to not be beneficial, with the authors hypothesizing that the additional training data was insufficient [30]. An additional issue in the application of transformers can be found in older literary texts, which often do not follow standard orthographic rules,

necessitating conversion approaches like the one by de la Rosa et al. [28].

One potential impediment to the application of transformers, especially to historical literary data, is the lack of training data, as transformer models are typically trained with at least billions of tokens (e.g. the original BERT implementation [73] used a corpus of more than three billion words). ECCO-Bert [76] is an example of a domain-specific transformer model, trained on a corpus of eighteenth-century data. In the work by Zhang et al. [53], it achieves better results than transformer models trained on modern data, with the accuracy increasing from 0.94 to 0.96, exemplifying the utility of time- and domain-specific models. Even without domain specific training data, transformers can, however, typically still outperform more traditional approaches. Schmidt et al. [30], for example, found transformer models trained on contemporary language to still clearly outperform their Naïve Bayes approach with the F1 score improving from 0.50 to 0.64 for an emotion classification task.

Many of the transformer models we encountered are specific to English. In addition, we encountered language-specific models for Chinese, German, Italian, and Spanish [23, 24, 30, 51]. For Dutch, German, and Italian we also found multilingual models being applied in monolingual contexts [28, 51, 58, 59], with Parigini et al. [51] finding an Italian specific model to clearly outperform the multilingual BERT variant with performance increasing from an F1 score 0.31 of to 0.49. Multilingual models were also applied to the analysis of multilingual datasets [26].

Beyond the training data-related issues raised so far, transformers have one other major shortcoming when applied to the literature domain: their context size. Early transformers like BERT [73] only had a context window size of 512 sub-word tokens. As longer words are typically represented as combinations of two or more sub-word tokens, this means that in practice, depending on the text, only about 300 individual words will be accepted at once. This is often enough to cover short-form poetry (e.g. [55, 58]), but woefully inadequate for entire novels, as even novellas will typically have tens of thousands of tokens with novels frequently reaching the hundred thousand mark [54]. While transformer models with arbitrary input lengths can theoretically be trained, the attention mechanism leads to a memory requirement that scales quadratically with the input, making much larger sizes infeasible. More recently, approaches like Longformers [77, 78] have proposed methods to limit memory consumption, by limiting the attention mechanism, and thereby enable longer input sequences; Longformers, however, also only allow for input sequences up to 4096 tokens. We are not aware of work applying any of these approaches to CLS, presumably, at least in part, because even 4000 tokens are not typically sufficient to cover the entire subject of analysis. Another alternative exists in hierarchical transformers [79], which can potentially handle even longer sequences. But again, we have not yet encountered them in the application context of CLS. More recently, in the machine learning community, architectures incorporating explicit communication between segments have been proposed, outperforming existing models on long sequence tasks [80, 81]. In practice, the length limitation does not always present a problem, for example, Parigini et al. [51] perform token-based annotations and process the entire text in chunks. Such window-based approaches, however, bar the model from considering any text outside the current window, which might be an issue for some annotation tasks.

Another variant of the transformer model that was employed in the surveyed literature is the sentence encoder. Such models are trained to produce embeddings of entire sentences or even short documents representing, similarly to word embeddings, the sentences' semantics in vector space. For example, Ehrmanntraut et al. [58] use them in addition to static word embeddings for identifying similarities in poetry. They find the transformer approaches to far outperform the static embedding-based ones, with performance on the overall similarity of embeddings reaching 0.79 as compared to the accuracy of 0.72 for the fastText approach. Similarly, Glass explores the use of sentence encoders to analyze if one text is an adaptation of another [40]. We did not find any instances of works making use of word mover's distance (WMD), a technique for quantifying document distances given individual word embeddings, in our surveyed papers. Perhaps this can be explained by the approach's time complexity, which can become prohibitive for long documents, although less expensive variants are available [82].

The mentioned shortcomings seem to not make transformers a natural fit for the domain of CLS. Despite this, as we found in our survey they are still the most prevalent machine learning methodology in recent literature in the CLS domain. We conjecture that this can be explained by the very good performance of transformers as compared to other techniques in language processing tasks (potentially outweighing domain adaptation issues) in combination with their relative ease of use.

3.5 Feature-based approaches

In our literature research, when leaving transformers aside, support vector machines (SVMs) were clearly the predominant means of approaching classification tasks. In terms of input features, we found uses of Tf-idf vectors, n-gram counts, and word counts for specific dictionary-based vocabularies. Some works also made use of preexisting classifiers or rule-sets to build features. SMVs, just as expected, perform well for classification tasks like genre identification [43].

Tf-idf is one of the most popular features in our review. It is a measure weighting the relative occurrence of a term (e.g. an individual word) in the current document by its rarity across the corpus, such that rare terms across the corpus that occur frequently in the document of interest receive a very high Tf-idf value. In this way, an individual document can be represented as a vector of real numbers where each element represents one term in the vocabulary. Orthogonal to this is the selection of what exactly constitutes a term in the vocabulary, typically individual tokens are chosen but alternatively character n-grams or word n-grams, that is to say all sequences of n characters or n tokens, could also be used. Brottrager et al. [56] employ a large variety of features to identify literary work that is likely to be externally reviewed, e.g. in a book review; from character-based metrics like the ratio of punctuation marks in the text to lexical ones like token n-grams, to semantic features like embedding cosine distance, and even complexity features like an ease of reading score. By contrast, Bonch-Osmolovskaya et al. [35] rely exclusively on Tf-idf, in their case the approach is likely to be token-based but details are not provided. With this comparatively simple method, they succeed in classifying diary entries into four classes, depending on what they describe.

Ciotti [37] made use of dictionary-based features, applying an Italian version of the LIWC [83] to collect the relative frequencies of words in specific categories (such as “Affect Words”, “Cognitive Processes”, or “Perceptual Processes”). Camps et al. [36] add musical features to their textual ones by, among others, adding bigrams of musical notes, as their subject of analysis is songs. They find character 3-g to be the best features for identifying song authors using an SVM on their dataset, with the feature reaching an F1 score of 0.79 outperforming word-lemmas at 0.67.

3.6 Topic modeling

Topic modeling is a technique to identify, for a corpus of texts, which topics are represented in which documents (see Sandhiya et al. [84] for a survey). Topics are generally inferred from the data in an unsupervised manner. In the case of Latent Dirichlet allocation (LDA), for example, topics are understood as a probability distribution across terms. A number of the works we surveyed [22, 60, 62] use LDA [85] as a means of topic modeling. Cooper et al. [22], for example, use it to segment their texts into topics to represent individual storytellers by means of the topic distribution their texts exhibit. Top2Vec [86] is an approach that, rather than relying on term distributions, encompasses existing embedding models to represent documents; it can be used in conjunction with universal sentence encoder (USE) embeddings, among other options. In our selection of papers, Top2Vec was employed by Zundert et al. [54], exploring if topic models can be used to identify literary genres as defined by publishers.

3.7 Stylometry

While stylometry does, strictly speaking, typically not qualify as a machine learning technique, we include it here for context as it is a very common technique. Stylometry in CLS is often performed using most-frequent-word (MFW) analysis, that is only the n most frequent words in the corpus are considered. Each document is represented by a vector with each element representing the occurrence count of one of the n MFWs. After normalization to account for the frequency of each token, one of a range of different distance metrics can be used. In this way, either individual documents or collections of documents (by taking the average vector), can be compared with a given document [87].

Eder [25] takes a fresh approach to this task by normalizing the number of tokens with semantically relevant words, which are extracted using static embeddings.

3.8 Metrics

In general terms, many of the metrics used in CLS are well-established in the machine learning community. For example, accuracy, F1 score, (e.g. [30]) but also BLEU (e.g. [28]), which was originally developed for machine translation evaluation. One metric that seems to generally be popular in the CLS community, as evidenced by its inclusion in MONAPipe [68], is Gamma; it can be used to evaluate classifier performance as well as inter-annotator agreement on span-based annotations. For example, Andresen et al. and Ehrmanntraut et al. [58, 88], employ it to evaluate their annotation agreement. Outside of our previously selected literature, Zehe et al. [89] employ it not only to measure inter-annotator agreements but also as a performance metric for their prediction system.

3.9 Model introspection

Some of the papers we surveyed make use of model introspection techniques. Kunilovskaya et al. [45], for example, use a linear kernel SVM to inspect feature weights and find five features that distinguish original Russian texts from translations to Russian (among them simple sentences and interrogative sentences). Steg et al. [48] take a similar approach, inspecting the importance of features (in this case individual tokens) in their Theil-Sen regressor using an existing implementation;[4] they point out differences in the importance of first and third person pronouns depending on which theoretical approach to narrativity is taken. Crucially, both of these occurrences not only validate their models' decision process but go so far as to derive literary insight by inspecting it.

3.10 Large and instruction-tuned language models

Since our survey considered works from 2022 and before, no instruction-tuned language models like ChatGPT were used. Instruction-tuned models [90] are Large Language Models (LLMs) that are tuned, based on human feedback, to provide helpful responses rather than exclusively being trained using self-supervised language modeling objectives. GPT-2, an LLM without instruction tuning, was used in our set of surveyed papers to generate poetry [49, 55]. In the near future, zero-shot methodology [91], that is to say inference using a pretrained model without training data, may simplify the annotation process for specific CLS tasks. Similarly, the few-shot capabilities of LLMs [92] could help scale annotations more quickly and simpler than traditional training approaches. That is to say, given only very few manually annotated examples, a large body of text could be automatically annotated with the given annotation schema for a specific concept allowing for corpus level analysis. Ziems et al. [93] evaluate the application of several LLMs, including ChatGPT, in social sciences while including a few tasks relevant to literary studies. In general, they find LLMs to potentially be ready for some zero-shot classification tasks in the context of social studies research. For example, ChatGPT perform particularly well on a stance detection tasks, where the objective is to identify a text's position on a given topic like *atheism* or the *legalization of abortion*. Overall, in their experiments, models perform the best on misinformation classification, stance detection, and emotion classification. The authors attribute this to the fact that in each of these tasks there is either an objective knowledge-based ground truth (as is the case for misinformation) or an annotation schema aligned with colloquial definitions. They do however also find that models perform the worst on tasks that require complex expert taxonomies (meaning tasks where the annotation guidelines are informed by domain expertise), which tend to not semantically align with much of the LLM's training data. An example of such a task is the annotation of character tropes, requiring the annotation of one of 72 tropes given a quote by a character. Aside from the large number of classes, this task is also difficult for LLMs because names of individual classes may not be easily understandable without further context, requiring expert knowledge to interpret the taxonomy. This finding seems particularly relevant in the context of the research process we outlined earlier, where theoretical work typically produces just such an annotation scheme. It remains to be seen how quickly these models are adopted in the CLS community, but we expect them to eventually see wide-spread use.

3.11 Domain-specific approaches

While some CLS methods align well with the methodology used for example in the NLP community, other methods are more specific to the field. For example, span-based text annotations are often performed in the field of NLP for tasks like named entity recognition (NER), whereas building graphs of character interactions is a technique more specific to the CLS domain.

3.11.1 Character-focused approaches

Characters here do not refer, as is usually the case in informatics contexts, to symbols, but to characters in the sense of people in the narrated world. As such, it is very specific to literary studies; while non-fictional texts may also refer to specific people, this is usually addressed by entity linking with existing knowledge bases, an approach that is not possible for characters in literary works, as they are in many cases not represented in external knowledge bases. There exist approaches for the automatic identification of characters and their occurrences; to solve this detection task in the general case, however, long document coreference resolution is required. Coreference resolution is the task of, for each described entity, resolving which spans in the text refer to it, in the case of characters, this may be names and personal pronouns, but also more general references like occupation titles that may identify individual characters. While recently, in the field of NLP, major advances have been made in terms of scores for coreference resolution systems [34, 94], they do not produce satisfactory results for

4 https://eli5.readthedocs.io/.

longer documents in the literature domain (as noted, for example, by Perri et al. [52]) although approaches specifically for this domain exist [95]. Accordingly, an alternative is needed; Perri et al. choose to rely on named mentions of characters instead, relying on exact matches of names only and thereby disregarding many references, e.g. those using only personal pronouns. Coreference resolution has recently been trained using supervised learning and transformer architectures, annotations for which are very time-consuming to build. A fairly large dataset in the domain of English literature already exists in the form of LitBank.[5]

The extraction of characters and their mentions is a prerequisite for the actual analysis step. Here, a typical approach is that of character networks, in general terms a graph where nodes are individual characters with edges representing their interaction, for example the number of interactions via an edge weight. On such graphs, a variety of established algorithms can be used for analysis, for example Konle et al. [50] find that, in all twenty novels they consider, the protagonist and their love interest rank first and second respectively in terms of a closeness-centrality measure, specifically temporal closeness centrality. In our surveyed work only Perri et al. and to a lesser extent Konle et al. use character networks as a method for literary analysis [50, 52]. Graph analysis allows literary researchers to answer a broad range of research questions, with Vauth [96], outside our surveyed papers, analyzing dramas by renowned German writer Kleist using network graphs. In general, graph analysis seems to be a very popular tool specifically for dramas, which we attribute to the rather straightforward extraction of characters and a set of datasets in multiple languages [97].

3.11.2 Narrative modeling

Two approaches in our survey are explicitly concerned with modeling narrative, i.e. modeling what happens on the story level by building up from computationally analyzing surface phenomena [48, 50]. Steg et al. [48] like Vauth et al. [98] attempt to measure the degree of narrativity in a given segment of text, attempting to quantify in different ways how much is happening at any point in the text. Konle et al. [75] on the other hand take a character-based approach to the same concept, interpreting plot and narrative as sequence of character graphs.

4 Datasets

Surveying the landscape of research data management in the German CLS community (specifically the SPP-CLS), Helling et al. [1] found XML, CSV, and plain text to be the three most prevalent data formats. XML is often used in the form of TEI,[6] a standard for representing all sorts of characteristics of texts. From an annotation point of view, many of the works we encountered built problem-specific datasets that have no immediate wider application (e.g. [35]). Others, however, annotated data intended for wider down-stream use (e.g. [46]), enabling researchers to build their work on existing annotated datasets. A whole range of work, for example, builds on the annotated drama datasets provided by the DraCor project [97].

5 Generating insights and results

The main body of this work has concerned itself with the details of how machine learning approaches are applied in CLS, at least as important to literary scholars, however, is the actual process of generating insights from data obtained in this way. We already discussed insights taken from model introspection, where feature weights can inform theoretical insights, in Section 3.9.

In most scenarios, machine learning replaces a large team of annotators, making it possible to, within the constraints of research projects, expand annotations beyond a small set of texts and onto an entire corpus. As a result corpus-level statistics, rather than information on individual texts, are subject to analysis.

6 Conclusions

In this survey, we introduced the field of computational literary studies (CLS) and provided an overview of the machine learning methods used by its practitioners. Our overview indicates that modern neural language models take a large role in the field, while still co-existing with traditional methods. CLS provides a potential test bed for NLP techniques with tasks such as coreference resolution being much more challenging in the literature domain than in the news datasets typically employed in NLP. Further, CLS is challenging as it demands a large degree of transparency

5 https://github.com/dbamman/litbank.

6 https://tei-c.org/.

from NLP methods, as black-box decisions run contrary to literary scholars' ultimate goal of interpretation. With advancements in machine learning techniques, processing longer texts will become increasingly feasible, which may give rise to new opportunities for automation in CLS. While a variety of methods are employed by literary scholars, we lined out a process that seems to be common to many works applying machine learning to the analysis of literature. Our review also highlighted two key challenges that the discipline of CLS faces, the first being the black-box nature of some machine learning models and the other being that of tying results generated by machine learning methods back to literary theory and, in turn, to gain insights. In CLS, the traditional pipeline-based approach to NLP is still alive, in part because, in conjunction with rule-sets, they allow for automation without annotations and in part because feature-based approaches often allow for inspecting the model's decision process, which may be crucial for assessing the adequacy of an approach from a literary studies point of view. We see great potential in the development of further simple-to-use tools and the case of stylometry shows that established techniques can be applied in various scenarios. Further simplifying the application of transformers may enable rapid scaling of annotations from a few examples to an entire corpus. We suspect that the few-shot and zero-shot capabilities of LLMs may constitute such a simplification in the application of models, and will be established as the standard methodology in CLS, following other fields of application.

Research ethics: Not applicable.

Author contributions: All the authors have accepted responsibility for the entire content of this submitted manuscript and approved submission.

Conflict of interest: Evelyn Gius is among the editors of the "Journal for Computational Literary Studies" which we include in our study.

Research funding: This study was funded by the DFG in the priority program Computational Literary Studies as part of the project "Evaluating Events in Narrative Theory (EvENT)" (grants BI 1544/11-1 and GI 1105/3-1).

Data availability: Not applicable.

References

[1] P. Helling, K. Jung, and S. Pielström, "Pragmatisches Forschungsdatenmanagement — qualitative und quantitative Analyse der Bedarfslandschaft in den Computational Literary Studies," in *DHd 2022 Kulturen des digitalen Gedächtnisses*, *Tagung des Verbands "Digital Humanities im deutschsprachigen Raum"*, vol. 8, 2022.

[2] C. Schöch, J. Dudar, and E. Fileva, "CLS INFRA D3.2: series of five short survey papers on methodological issues (= survey of methods in computational literary studies)," *Tech. Rep. Zenodo*, pp. 1—159, 2023.

[3] N. Z. Da, "The computational case against computational literary studies," *Crit. Inq.*, vol. 45, no. 3, pp. 601—639, 2019.

[4] T. Underwood, *Dear Humanists: Fear Not the Digital Revolution*. 2019. Available at: https://www.chronicle.com/article/dear-humanists-fear-not-the-digital-revolution/.

[5] F. Jannidis, "On the perceived complexity of literature. A response to nan Z. Da," *J. Cult. Anal.*, vol. 1, no. 1, p. 11829, 2020.

[6] F. Moretti, "The slaughterhouse of literature," *Mod. Lang. Q.*, vol. 61, no. 1, pp. 207—227, 2000.

[7] F. Moretti, *Distant Reading*, London, Verso Books, 2013.

[8] *Martin Mueller on* "Morgenstern's Spectacles or the Importance of not-reading" — *NUDHL*, 2013. Available at: https://sites.northwestern.edu/nudhl/?p=433.

[9] T. Weitin, "Scalable reading," *Z. Lit. Linguist.*, vol. 47, no. 1, pp. 1—6, 2017.

[10] E. Gius, "Algorithmen zwischen Strukturalismus und Postcolonial Studies. Zur Kritik und Entwicklung der Computationellen Literaturwissenschaft," in *Toward Undogmatic Reading. Narratology, Digital Humanities and Beyond*, Hamburg, 2021.

[11] B. Zimmer, *Language Log ≫ Rowling and "Galbraith": An Authorial Analysis*, 2013. Available at: https://languagelog.ldc.upenn.edu/nll/?p=5315.

[12] A. van Cranenburgh and E. Ketzan, "Stylometric literariness classification: the case of stephen king," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 189—197.

[13] M. L. Jockers, *Macroanalysis: Digital Methods and Literary History*, Champaign, Illinois, University of Illinois Press, 2013.

[14] C. P. Snow, *The Two Cultures and the Scientific Revolution*, New York, Cambridge University Press, 1959.

[15] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "OntoNotes: the 90% solution," in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, New York City, New York, USA, Association for Computational Linguistics, 2006, pp. 57—60.

[16] J. Wu, L. Ouyang, D. M. Ziegler, et al., "Recursively summarizing books with human feedback," 2021, arXiv: 2109.10862 [cs].

[17] T. George, "Hermeneutics," in *The Stanford Encyclopedia of Philosophy*, Winter, 2021.

[18] E. Gius and J. Jacke, "The hermeneutic profit of annotation: on preventing and fostering disagreement in literary analysis," *IJHAC*, vol. 11, no. 2, pp. 233—254, 2017.

[19] D. Malvern, B. Richards, N. Chipere, and P. Durán, "Traditional approaches to measuring lexical diversity," in *Lexical Diversity and Language Development: Quantification and Assessment*, London, Palgrave Macmillan UK, 2004, pp. 16—30.

[20] A. Pichler and N. Reiter, "Reflektierte textanalyse," in *Reflektierte algorithmische Textanalyse: Interdisziplinäre(s) Arbeiten in der CRETA-Werkstatt*, 2020, pp. 43—60.

[21] E. Gius, J. C. Meister, M. Meister, et al, *CATMA*, Zenodo, 2022. Available at: https://zenodo.org/record/1470118.

[22] A. Cooper, M. Antoniak, C. De Sa, M. Migiel, and D. Mimno, "Tecnologica cosa': modeling storyteller personalities in

boccaccio's 'decameron'," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, Punta Cana, Dominican Republic. (Online),* 2021, pp. 147—153.

[23] M. K. Schumacher, M. Flüh, and M. Lemke, "The model of choice using pure CRF- and BERT-based classifiers for gender annotation in German fantasy fiction," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[24] W. Xie, J. Lee, F. Zhan, X. Han, and C.-Y. Chow, "Unsupervised adverbial identification in modern Chinese literature," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, Punta Cana, Dominican Republic. (Online),* 2021, pp. 91—95.

[25] M. Eder, "Boosting word frequencies in authorship attribution," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 387—397.

[26] J. C. Tello and J. de la Rosa, "Evaluation of multilingual BERT in a diachronic, multilingual, and multi-genre corpus of bibles," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[27] T. Clérice, "Ground-truth free evaluation of HTR on old French and Latin medieval literary manuscripts," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 1—24.

[28] J. de la Rosa, Á. Cuéllar, and J. Lehmann, "The modernisa project: orthographic modernization of Spanish golden age dramas with Language Models," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[29] A. Karlińska, C. Rosiński, J. Wieczorek, et al., "Towards a contextualised spatial-diachronic history of literature: mapping emotional representations of the city and the country in polish fiction from 1864 to 1939," in *Proceedings of the 6th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Gyeongju, Republic of Korea, 2022, pp. 115—125.

[30] T. Schmidt, K. Dennerlein, and C. Wolff, "Emotion classification in German plays with transformer-based Language Models pretrained on historical and contemporary language," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 67—79.

[31] A. Abdibayev, Y. Igarashi, A. Riddell, and D. Rockmore, "Automating the detection of poetic features: the limerick as model organism," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, Punta Cana, Dominican Republic. (Online),* 2021, pp. 80—90.

[32] M. A. Algee-Hewitt, "A computational approach to epistemology in poetry of the long eighteenth century — a case study in objects and ideas," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[33] A. Piper and S. Bagga, "A quantitative study of fictional things," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 268—279.

[34] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "SpanBERT: improving pre-training by representing and predicting spans," *Trans. Assoc. Comput. Linguist.*, vol. 8, pp. 64—77, 2020.

[35] A. Bonch-Osmolovskaya, V. Vorobieva, A. Kriukov, and M. Podriadchikova, "Distant reading of Russian soviet diaries (prozhito database)," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, DH2022 Local Organizing Committee, 2022.

[36] J.-B. Camps, C. Chaillou, V. Mariotti, and F. Saviotti, "Textual, metrical and musical stylometry of the trouvères songs," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, DH2022 Local Organizing Committee, 2022.

[37] F. Ciotti, "Computational approaches to literary periodization: an experiment in Italian narrative of 19th and 20th century," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[38] K. Dennerlein, T. Schmidt, and C. Wolff, "Emotion courses in German historical comedies and tragedies," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[39] M. Eder and A. Šeļa, "One word to rule them all: understanding word embeddings for authorship attribution," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[40] G. Grant, "An adaptive methodology: machine learning and literary adaptation," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[41] J. B. Herrmann, J. Byszuk, and G. Grisot, "Using word embeddings for validation and enhancement of spatial entity lists," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[42] L. Ivanov, "Abstractness/concreteness as stylistic features for authorship attribution," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[43] P.-C. Langlais, J.-B. Camps, N. Baumard, and O. Morin, "From roland to conan: first results on the corpus of French literary fictions (1050-1920)," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, DH2022 Local Organizing Committee, 2022.

[44] M. K. Schumacher, "Measuring space in German novels — the spatial index (SI) as measurement for narrative space," in *Digital Humanities 2022 Combined Abstracts*, Tokyo, Japan, 2022.

[45] M. Kunilovskaya, E. Lapshinova-Koltunski, and R. Mitkov, "Translationese in Russian literary texts," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 101—112.

[46] D. Schmidt, A. Zehe, J. Lorenzen, et al., "The FairyNet corpus — character networks for German fairy tales," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 49—56.

[47] F. Schneider, B. Barz, P. Brandes, S. Marshall, and J. Denzler, "Data-driven detection of general chiasmi using lexical and semantic features," in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 96—100.

[48] M. Steg, K. Slot, and F. Pianzola, "Computational detection of narrativity: a comparison using textual features and reader response," in *Proceedings of the 6th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Gyeongju, Republic of Korea, 2022, pp. 105—114.

[49] J. Wöckener, T. Haider, T. Miller, et al., "End-to-End style-conditioned poetry generation: what does it take to learn from examples alone?" in *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, Punta Cana, Dominican Republic. (Online), 2021, pp. 57—66.

[50] L. Konle and F. Jannidis, "Modeling plots of narrative texts as temporal graphs," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 318—336.

[51] M. Parigini and M. Kestemont, "The roots of doubt. Fine-Tuning a BERT model to explore a stylistic phenomenon," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 72—91.

[52] V. Perri, L. Qarkaxhija, A. Zehe, A. Hotho, and I. Scholtes, "One graph to rule them all: using NLP and graph neural networks to analyse tolkien's legendarium," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 291—317.

[53] J. Zhang, Y. C. Ryan, I. Rastas, F. Ginter, M. Tolonen, and R. Babbar, "Detecting sequential genre change in eighteenth-century texts," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 243—255.

[54] J. J. van Zundert, M. Koolen, J. Neugarten, P. Boot, W. van Hage, and O. Mussmann, "What do we talk about when we talk about topic?," in *Proceedings of the Computational Humanities Research Conference 2022*, vol. 3290, Antwerp, Belgium, CEUR Workshop Proceedings, 2022, pp. 398—410.

[55] A. Abdibayev, Y. Igarashi, A. Riddell, and D. Rockmore, "Limericks and computational poetics: the minimal pairs framework. Computational challenges for poetic analysis and synthesis," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.117.

[56] J. Brottrager, A. Stahl, A. Arslan, U. Brandes, and T. Weitin, "Modeling and predicting literary reception. A data-rich approach to literary historical reception," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.3929/ethz-b-000596039.

[57] K. Du, J. Dudar, and C. Schöch, "Evaluation of measures of distinctiveness. Classification of literary texts on the basis of distinctive words," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.102.

[58] A. Ehrmanntraut, T. Hagen, F. Jannidis, L. Konle, M. Kröncke, and S. Winko, "Modeling and measuring short text similarities. On the multi-dimensional differences between German poetry of realism and modernism," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.116.

[59] M. Koolen, J. Neugarten, and P. Boot, "This book makes me happy and sad and I love it'. A rule-based model for extracting reading impact from English book reviews," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.104.

[60] J. Schröter and K. Du, "Validating topic modeling as a method of analyzing sujet and theme," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.91.

[61] H. Shin, "Analyzing the positive sentiment towards the term "queer" in Virginia woolf through a computational approach and close reading," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.106.

[62] Y. Völkl, S. Sarić, and M. Scholger, "Topic modeling for the identification of gender-specific discourse. Virtues and vices in French and Spanish 18th century periodicals," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.108.

[63] A. M. Weimer, F. Barth, and T. Dönicke, "The (In-)Consistency of literary concepts. Operationalising, annotating and detecting literary comment," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.108.

[64] A. Pramanick, Y. Hou, and I. Gurevych, "A diachronic analysis of the NLP research paradigm shift: when, how, and why?" 2023, arXiv: 2305.12920 [cs.CL].

[65] M. Honnibal, I. Montani S. Van Landeghem, and A. Boyd., "spaCy: Industrial-strength Natural Language Processing in Python," 2020. Available at: https://zenodo.org/record/8123552

[66] A. O. Kehinde, "Pathways to the native storyteller: a method to enable computational story understanding," Ph.D. thesis, 2020.

[67] A. Ehrmanntraut, L. Konle, and F. Jannidis, *LLpro — A Literary Language Processing Pipeline for German Narrative Texts*, 2022. Available at: https://github.com/aehrm/LLpro.

[68] T. Dönicke, F. Barth, H. Varachkina, and C. Sporleder, "MONAPipe: modes of narration and attribution pipeline for German computational literary studies and language analysis in spaCy," in *Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022)*, Potsdam, Germany, KONVENS 2022 Organizers, 2022, pp. 8—15.

[69] J. Pennington, R. Socher, and C. Manning, "GloVe: global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1532—1543.

[70] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of Word representations in vector space*, arXiv:1301.3781 [cs.CL], 2013.

[71] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 135—146, 2017.

[72] J. Bromley, J. W. Bentz, L. Bottou, et al., "Signature verification using a "siamese" time delay neural network," *Adv. Neural Inf. Process. Syst.*, vol. 6, pp. 737—744, 1993.

[73] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, USA, 2019, pp. 4171—4186.

[74] J. Bandy and N. Vincent, "Addressing "documentation debt" in machine learning: a retrospective datasheet for BookCorpus," in *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, vol. 1, 2021.

[75] L. Konle and F. Jannidis, "Domain and task adaptive pretraining for Language Models," in *Proceedings of the Workshop on Computational Humanities Research (CHR 2020)*, vol. 2723, Amsterdam, the Netherlands, CEUR Workshop Proceedings, 2020, pp. 248—256.

[76] I. Rastas, Y. Ciarán Ryan, and I. Tiihonen, "Explainable publication year prediction of eighteenth century texts with the BERT model," in *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, Dublin, Ireland, 2022, pp. 68—77.

[77] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: the long-document transformer," 2020, arXiv: 2004.05150 [cs].

[78] M. Zaheer, G. Guruganesh, K. A. Dubey, et al., "Big bird: transformers for longer sequences," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 17283—17297, 2020.
[79] X. Zhang, F. Wei, and M. Zhou, "HIBERT: document level pre-training of hierarchical bidirectional transformers for document summarization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019, pp. 5059—5069.
[80] A. Bertsch, Y. Kuratov, and M. Burtsev, "Unlimiformer: long-range transformers with unlimited length input," 2023, arXiv: 2305.01625 [cs].
[81] A. Bulatov, Y. Kuratov, and M. Burtsev, "Recurrent memory transformer," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 11079—11091, 2022.
[82] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37, Lille, France, Proceedings of Machine Learning Research, 2015, pp. 957—966.
[83] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: LIWC and computerized text analysis methods," *J. Lang. Soc. Psychol.*, vol. 29, no. 1, pp. 24—54, 2010.
[84] R. Sandhiya, A. M. Boopika, M. Akshatha, S. V. Swetha, and N. M. Hariharan, "A review of topic modeling and its application," in *Handbook of Intelligent Computing and Optimization for Sustainable Development*, 2022, pp. 305—322. Chap. 15.
[85] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993—1022, 2003.
[86] D. Angelov, "Top2Vec: distributed representations of topics," 2020, arXiv: 2008.09470 [cs, stat].
[87] S. Evert, F. Jannidis, T. Proisl, et al., "Understanding and explaining delta measures for authorship attribution," *Digit. Scholarsh. Humanit.*, vol. 32, no. 2, pp. ii4—ii16, 2017.
[88] M. Andresen, B. Krautter, J. Pagel, and N. Reiter, "Who knows what in German drama? A composite annotation scheme for knowledge transfer. Annotation, evaluation, and analysis," *J. Comput. Lit. Stud.*, vol. 1, no. 1, 2022, https://doi.org/10.48694/jcls.107.
[89] A. Zehe, L. Konle, L. K. Dümpelmann, et al., "Detecting scenes in fiction: a new segmentation task," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume.* Online, 2021, pp. 3167—3177.
[90] L. Ouyang, K. Wu, X. Jiang, et al., "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, vol. 35, New Orleans, Louisiana, USA, Curran Associates, Inc., 2022, pp. 27730—27744.
[91] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large Language models are zero-shot reasoners," in *Advances in Neural Information Processing Systems*, vol. 35, New Orleans, Louisiana, USA, Curran Associates, Inc., 2022, pp. 22199—22213.
[92] T. Brown, B. Mann, N. Ryder, et al., "Language models are few-shot learners," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 1877—1901, 2020.
[93] C. Ziems, W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang, "Can large language models transform computational social science?" 2023, arXiv: 2305.03514 [cs].
[94] V. Dobrovolskii, "Word-level coreference resolution," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic. (Online), 2021, pp. 7670—7675.
[95] S. Toshniwal, S. Wiseman, A. Ettinger, K. Livescu, and K. Gimpel, "Learning to ignore: long document coreference with bounded memory neural networks," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online, 2020, pp. 8519—8526.
[96] M. Vauth, "Figurenrede in kleists literarischem werk," in *Eine digitale Narratologie der Binnenerzählung: Untersuchungen zu den Dramen und Novellen Heinrich von Kleists*, Berlin, Heidelberg, Digitale Literaturwissenschaft, 2023, pp. 153—204.
[97] F. Fischer, I. Börner, M. Göbel, et al., "Programmable corpora: introducing DraCor, an infrastructure for the research on European drama," in *Digital Humanities 2019: "Complexities" (DH2019)*, Utrecht, Utrecht University, 2019.
[98] M. Vauth, H. O. Hatzel, E. Gius, and C. Biemann, "Automated event annotation in literary texts," in *Proceedings of the Conference on Computational Humanities Research 2021*, vol. 2989, Amsterdam, The Netherlands, CEUR Workshop Proceedings, 2021, pp. 333—345.

Bionotes

Hans Ole Hatzel
Department of Informatics, Universität Hamburg, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
hans.ole.hatzel@uni-hamburg.de

Hans Ole Hatzel studied Computer Science and is currently a Ph.D. student at the Universität Hamburg where he also got his Bachelors and Masters degrees in 2017 and 2020. He works on event and story modeling in literary texts as part of the EvENT (Evaluating Events in Narrative Theory) project, publishing in both DH and NLP venues.

Haimo Stiemer
Technical University of Darmstadt, Institute of Linguistics and Literary Studies, Residenzschloss 1, 64283 Darmstadt, Germany
stiemer@linglit.tu-darmstadt.de

Dr. Haimo Stiemer is a research associate at the fortext lab, where he supervises the EvENT project (Evaluating Events in Narrative Theory). In addition to computational narratology and digital literary studies, his research interests include the sociology of literature (especially Bourdieu's field theory), literary and cultural theory, modernist literature with a special focus on the German-language literatures of Eastern and Central Europe, and journal research.

Chris Biemann
Department of Informatics, Universität Hamburg, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
chris.biemann@uni-hamburg.de

Prof. Dr. Chris Biemann is scientific director of the House of Computing & Data Science, and the head of the Language Technology group at the Informatics department, at Universität Hamburg. He holds a doctorate from the University of Leipzig since 2007. While his PhD research was focused on unsupervised knowledge-free methods in language processing that leverage large corpora for pretraining, his research now encompasses all aspects of natural language processing, with a focus on semantics and on applications in different fields of science and the humanities.

Evelyn Gius
Technical University of Darmstadt, Institute of Linguistics and Literary Studies, Residenzschloss 1, 64283 Darmstadt, Germany
evelyn.gius@tu-darmstadt.de

Prof. Dr. Evelyn Gius is Professor of Digital Philology and Modern German Literary Studies and head of the fortext lab. She has been working in the field of Digital Humanities for about 15 years. Her research interests are mainly in the field of Computational Literary Studies and include in particular (computational) narratology, manual annotation and questions of operationalization.