

DATALAB: A Platform for Data Analysis and Intervention

Yang Xiao^{♣*}, Jinlan Fu[★], Weizhe Yuan[♣], Vijay Viswanathan[♣],
Zhoumianze Liu[♣], Yixin Liu[▲], Graham Neubig^{♣†}, Pengfei Liu[♣][†]

[♣]Carnegie Mellon University, [♣]Fudan University, [★]National University of Singapore,
[▲]Yale University, [†]Inspired Cognition

Abstract

Despite data’s crucial role in machine learning, most existing tools and research tend to focus on systems on top of existing data rather than how to interpret and manipulate data. In this paper, we propose DATALAB, a unified data-oriented platform that not only allows users to interactively analyze the characteristics of data, but also provides a standardized interface for different data processing operations. Additionally, in view of the ongoing proliferation of datasets, DATALAB has features for dataset recommendation and global vision analysis that help researchers form a better view of the data ecosystem. So far, DATALAB covers 1,715 datasets and 3,583 of its transformed version (e.g., hyponyms replacement), where 728 datasets support various analyses (e.g., with respect to gender bias) with the help of 140M samples annotated by 318 feature functions.¹ DATALAB is under active development and has been recently upgraded based on reviewers’ constructive suggestions.² We have released a wealth of resources to meet the diverse needs of researchers: *web platform*,³ open-sourced code of web platform,⁴ *web API*, open-sourced SDK,⁵ *PyPI* published package,⁶ and online documentation.⁷

1 Introduction

Datasets power modern natural language processing (NLP) systems, playing an essential

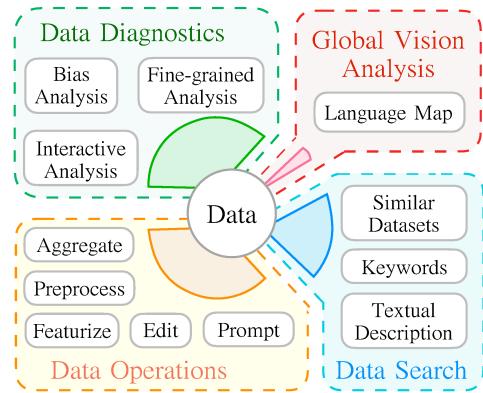


Figure 1: Overview of DATALAB functionality

role in model training, evaluation, and deployment (Paullada et al., 2021). Furthermore, methods to process data and understand have been subject to much research, including on topics such as data augmentation (Fadaee et al., 2017; Feng et al., 2021), adversarial evaluation (Jia and Liang, 2017; Ribeiro et al., 2021), bias analysis (Zhao et al., 2018a; Blodgett et al., 2020), and prompt-based learning (Liu et al., 2021b). Despite the critical role of data in NLP, the majority of open-source tooling regarding NLP has focused on methods to *build models given data*, rather than to *analyze and intervene upon the data itself*. In this paper, we present DATALAB, a unified platform that allows NLP researchers to perform a number of data-related tasks in an efficient and easy-to-use manner:

(1) **Data Diagnostics:** While a significant amount of research has focused on interpreting the outputs of machine learning systems (Lipton, 2018; Belinkov and Glass, 2019), data deserves deeper understanding as a first-class citizen of the machine learning ecosystem. DATALAB allows for analysis and understanding of data to uncover undesirable traits such as hate speech, gender bias, or label imbalance (as shown in Fig.1 and § 3.1).

(2) **Operation Standardization:** There are a num-

*Work done during a remote research collaboration with CMU

[†]Corresponding author

¹Users can also customize their favored feature functions using DATALAB SDK.

²Recent update: <https://datalab.nlpedia.ai/update>

³<http://datalab.nlpedia.ai/>

⁴https://github.com/ExpressAI/DataLab_web

⁵<https://github.com/ExpressAI/DataLab>

⁶<https://pypi.org/project/datalabs/>

⁷<https://expressai.github.io/DataLab/>

Aspects	Numbers
Tasks/Languages	142/331
Features/Prompts	318/1007
Plain/Diagnostic Datasets	1,715/3,583
Annotated Datasets	728
Annotated Samples	139,570,057

Table 1: Key statistics of DATALAB. “Diagnostic Dataset” refers to a dataset obtained by applying transformations to the original version.⁹ “Annotated” indicates datasets or samples where we compute features to obtain additional information that is not originally present in the dataset.

ber of well-designed packages for data-oriented operations such as preprocessing (Loper and Bird, 2002; Manning et al., 2014; Kudo, 2020) or editing (Ribeiro et al., 2021; Dhole et al., 2021). In practice, however, the diversity of requirements makes it necessary for users to install a variety of packages that use different data processing interfaces. This (a) reduces the efficiency of development, (b) can confuse users (e.g., not knowing what preprocessing methods are appropriate for a given dataset?), and (c) is detrimental for reproducibility (Marie et al., 2021). DATALAB provides and standardizes a large number of data processing operations, assigning each operation a unique identifier, to mitigate these issues (§ 3.2).

(3) **Data Search** An important question in practice is which datasets to use in a given scenario, given the huge proliferation of datasets in recent years.⁸ DATALAB provides a semantic dataset search tool to help identify appropriate datasets (§ 3.3).

(4) **Global Analysis** Beyond individual datasets, analyzing the entire ecosystem of existing datasets as a whole can yield insights. From a birds-eye view, we can get a clearer picture: *where we are* and *where efforts should be focused* to avoid systemic inequalities (Blodgett et al., 2020; Blasi et al., 2021). DATALAB provides tools to perform similar global analyses over a variety of datasets (§ 3.4).

With the above use cases in mind, DATALAB focuses on the following design principles:

- **Broad-coverage:** DATALAB is designed to cover the majority of NLP tasks, and imports data from a very large number of plain datasets

⁸ According to Papers With Code, the number of AI-related academic datasets has doubled in the past two years.

⁹ We collect diagnostic datasets by performing an extensive literature review and searching for existing works that released diagnostic samples from different tasks.

and diagnostic ones as shown in Table 1.¹⁰

- **Interpretable:** DATALAB has annotated statistical information for many datasets (728 datasets, 139,570,057 samples) that is not originally included in the dataset. These features can help researchers and developers better understand datasets before use, and help data creators improve data quality (e.g., removing artifacts, bias)
- **Unified:** One of the main goals of DATALAB is to unify different data analysis and processing operations into one platform and SDK. To achieve this goal, we design a generalized typology for data and operations (Figure 2).
- **Interactive:** DATALAB makes data exploration, assessment, and processing more accessible and efficient (real-time search, comparison, filtering, **generation of dataset diagnostic reports**). DATALAB can also be used as an off-the-shelf *annotation platform* where some missing yet important crowdsourcable information can be contributed by users.
- **Inspirational:** DATALAB’s global view of datasets makes it possible to inspire new research directions, e.g. by (i) finding more appropriate datasets as shown in §3.3 (ii) tracking the global status of dataset development and identifying future directions as illustrated in §3.4.

2 Related Work

Toolkits for NLP Pipelines There are a wealth of toolkits that support the processing of various NLP tasks, making it easier to build a composable NLP workflow. Typical examples are NLTK (Loper and Bird, 2002), NLPCurate (Clarke et al., 2012), Stanford CoreNLP (Manning et al., 2014), AllenNLP (Gardner et al., 2018), SpaCy (Honnibal and Montani, 2017), GluonNLP (Guo et al., 2020), Forte (Liu et al., 2021c), HuggingFace (Lhoest et al., 2021).

In contrast to these toolkits, DATALAB focuses on data analysis, bias diagnostics, and standardization of data-related operations. Moreover, besides providing the SDK, DATALAB also provides a web-based interactive platform, featuring hundreds of datasets and millions of additional annotations w.r.t. diverse features. KYD (Google, 2021) also provides a web platform for data analysis but it mainly focuses on image data. ExplainaBoard (Liu et al., 2021a) presents an analysis platform while it focuses on system diagnostics.

¹⁰ Details can be found in Appendix.

Standardization by Community Wisdom In ML in general and NLP in particular, researchers have been paying increasing attention to analyzing and improving systems from the perspective of data. In NLP, one major challenge in data processing is the diversity of data formats (e.g., *CONLL*, *BRAT*), task types (e.g., classification, generation) and design considerations (e.g., which types of preprocessing or augmentation) hinders the establishment of a unified platform. Recently, however, researchers in the field are actively trying to alleviate this problem by allowing community members to collectively contribute data-related operations on the same set of code frameworks, and eventually build a data processing platform around those operations. For example, HuggingFace (Lhoest et al., 2021) and Tensorflow (TFData, 2021) Datasets, where researchers in the community contribute data loaders for different tasks and datasets. In XL-Augmentor (Dhole et al., 2021) and Prompt Sourcing (Sanh et al., 2021) different data transformations or prompts are crowdsourced respectively.

After seeing this implicit pattern, we ask, can we have a more general platform above to unify all of these different operations? DATALAB makes a step towards this goal by not only focusing on how to unify data loader interfaces like Huggingface and Tensorflow have done, but also unifying data operations and analysis.

3 DATALAB

In this section we detail four major varieties of functionality provided by DATALAB.

3.1 Data Diagnostics

Data diagnostics aim to provide users with a comprehensive picture of data through various statistical analyses, enabling better model designs.

3.1.1 Fine-grained Analysis

Fine-grained analysis aims to answer the question: *what are the characteristics of a dataset?* Existing works have shown its advantages in better system designs (Zhong et al., 2019; Fu et al., 2020b; Tejaswin et al., 2021). Conceptually, this analysis over various dimensions can be performed over each data point (i.e. sample-level) or whole datasets (i.e. dataset-level). These are either generic (*text length* at sample-level or *the average text length* at corpus-level) or task-specific (for summarization: *summary compression* (Chen et al., 2020) or *the average of summary compression*). We detail the

features utilized for fine-grained analysis in Appendix.

One key contribution of DATALAB is that we not only design rich sample-level and dataset-level features, but also compute and store those features in a database for easy browsing. As shown in Table 1, so far, we have designed more than 300 features and computed features for 140M samples.

3.1.2 Bias Analysis

The research question to be answered by bias analysis is: *Does the dataset contain potential bias (e.g., artifacts, gender bias)?* Bias problems have been discussed extensively in NLP (Zhao et al., 2018a; Blodgett et al., 2020), and we argue that establishing a unified platform for data bias analysis can more efficiently identify or prevent (for data creators) data bias problems. For example, through the artifact analysis, users can know the shortcut provided by the dataset for model training and be inspired to design more robust systems. So far, DATALAB supports three types of bias analysis.

Artifact Identification As observed in many previous works (Gururangan et al., 2018; McCoy et al., 2019), artifacts commonly exist in datasets, which provide shortcuts for model learning and therefore reduce its robustness. DATALAB allows researchers to easily identify potential artifacts in a dataset using the features we have pre-computed for each sample. Specifically, we use PMI (Pointwise mutual information) (Bouma, 2009) to detect whether there is an association between two features (e.g. sentence length vs. label). We detail this method using an example in Appendix.¹¹

Gender Bias Analysis Gender bias is a prevalent social phenomenon. In this work, we introduce a multidimensional gender biased dictionary¹² used by Dinan et al. (2020) to measure the degree of gender bias in a dataset. Given a dictionary D_1 of female names and a dictionary D_2 of male names. Suppose a dataset A with N samples has n_1 name appearing in D_1 and n_2 in D_2 . Following Zhao et al. (2018b), we can calculate the female bias for dataset A as n_1/N ; the male bias as n_2/N .

Hate Speech Analysis Hate speech (Badjatiya et al., 2017) can lead to a "dehumanizing effect"

¹¹https://expressai.github.io/DataLab/docs/WebUI/bias_analysis_for_artifacts

¹²huggingface.gender_bias

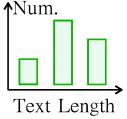
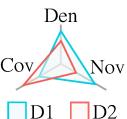
Aspect	Functionality	Input	Example Output
	Fine-grained Analysis	One dataset	
Diagnostics	Bias Analysis	One dataset	
	Interactive Analysis	Multiple datasets	
Operations	Aggregate, Edit, Featureize, Prompt, Preprocess	One dataset	
Data Search	Search	Keywords, textual descriptions, similar dataset	
Global Vision	Language Map	Multiple datasets	

Table 2: A graphical breakdown of the functionality of DATALAB.

that harms people’s mental health by undermining empathy (Tsesis, 2002). We make a first step by following Davidson et al. (2017), classifying the samples into *hate speech*, *offensive language* and *neither* categorizing by the “hatesonar” tool.¹³ Users are also allowed to customized other hate speech models using DATALAB SDK. We also averaged the offensiveness of all samples in a dataset to analyze the hate speech bias of the dataset.¹⁴

3.1.3 Interactive Analysis

Interactive analysis aims to meet users’ customized data analysis requirements in real time. Although interactivity is present in many aspects of DATALAB, we highlight here its use in three scenarios that make data analysis more accessible. (1) Users

¹³[pypi.org.hatesonar](https://pypi.org/project/hatesonar)

¹⁴Note that deciding whether a sentence contains toxic language is a complex task, which may involve the confounding effects of dialect and the social identity of a speaker Sap et al. (2019), and future iterations of DataLab may use meta-data of datasets to further perform this analysis intersectionally. We have also stored the results of hate speech detector for all samples to make the analysis process more transparent and well-grounded and users could browse them and report error cases.

can choose two datasets they are interested in and align them for comparative analysis over different dimensions, as shown in Table 2. (2) Users can upload their own datasets and DATALAB will generate diagnostic reports for comprehensive analysis and evaluation of the datasets. (3) Users can contribute some missing metadata information by directly editing in the web interface.

3.2 Data Operations

Another key feature of DATALAB is the standardization of different data operations into a unified format to satisfy different data processing requirements in one place. To this end, we devised a general typology for the concepts of data and operation as shown in Figure 2 and curated schemas for these objects. For the operation schema, we introduced (i) “operation id”: so that researchers can report them papers for easy re-implementation for follow-up research. (ii) “contributor” to credit those who contributed to the operation. Notably, user-defined operations are also supported (we give an example in Appendix).

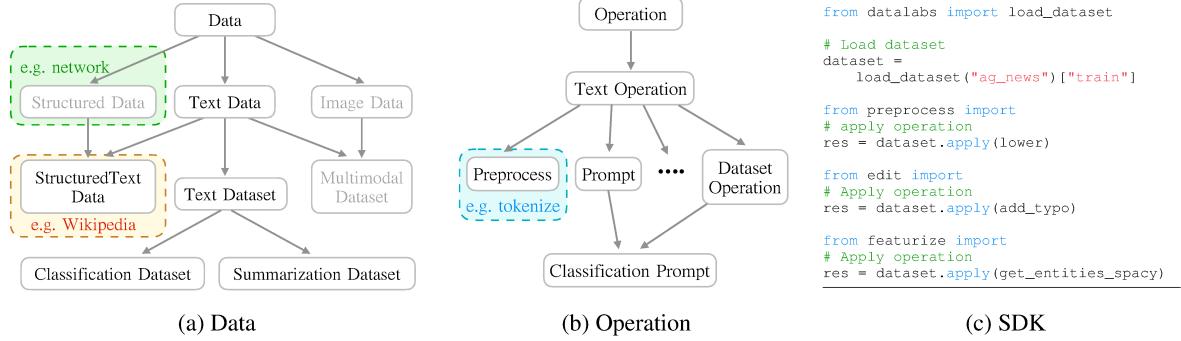


Figure 2: Typology of Data and Operations. Gray-white text (e.g., *Image Data*) indicates that the data type has been defined but we have not yet added data of that type.

Preprocessing Data preprocessing (e.g., tokenization) is an indispensable step in training deep learning and machine learning models, and the quality of the dataset directly affects the learning of models. Currently, DATALAB supports both general preprocessing functions and task-specific ones, which are built based on different sources, such as SpaCy (Honnibal and Montani, 2017), NLTK (Loper and Bird, 2002), Huggingface tokenizer.¹⁵

Editing Editing aims to apply certain transformations to a given text, which spans multiple important applications in NLP, for example (i) adversarial evaluation (Ribeiro et al., 2021; Wang et al., 2021), which usually requires diverse perturbations on test samples to test the robustness of a system. (ii) Data augmentation (Wei and Zou, 2019; Dhole et al., 2021; Feng et al., 2021). Essentially, many of the methods for constructing augmented or diagnostic datasets involve some editing operation on the original dataset (e.g., named entity replacement in diagnostic dataset construction (Ribeiro et al., 2021), token deletion in data augmentation (Wei and Zou, 2019)). DATALAB provides a unified interface for data editing and users can easily apply to edit the data they are interested in.

Featurizing This operation aims to compute sample-level features of a given text. In DATALAB, in addition to designing some general feature functions (e.g. *get_length* operation calculates the length of the text.), we also customize some feature functions for specific tasks (e.g. *get_oracle* operation for the summarization task that calculates the oracle summary of the source text.).

¹⁵[huggingface.tokenizers](https://huggingface.co/tokenizers)

Aggregating Aggregation operations are used to compute corpus-level statistics such as TF-IDF (Salton and Buckley, 1988), label distribution. Currently, DATALAB supports both generic aggregation operations applicable to any task and some customized ones for four NLP tasks (classification, summarization, extractive question answering and natural language inference).

Prompting Prompt-based learning (Liu et al., 2021b) has received considerable attention, as better utilization of pretrained language models benefits many NLP tasks. In practice, what makes a good prompt is a challenging question. We define the prompt schema as shown in fig. 3. The elements we included in a prompt cover diverse aspects including its features (e.g. length, shape, etc.), metadata (e.g. unique identifier, language, etc.), attributes (e.g. template, answers, etc.) as well as its performance w.r.t. different pre-trained language models and settings. The design can not only help researcher design prompts but also analyze what makes a good prompt.

So far, DATALAB covers 1007 prompts which can be applied to five types of tasks (topic classification, sentiment classification, sentence entailment, summarization, natural language inference), covering 309 datasets in total.

3.3 Data Search

Data search aims to answer the research question: which datasets should one use given a description of a research idea? As more datasets are proposed, there is an open question of how to choose the right dataset for a given application. DATALAB takes a step towards solving this problem by including semantic dataset search.

DATALAB data search takes a natural language

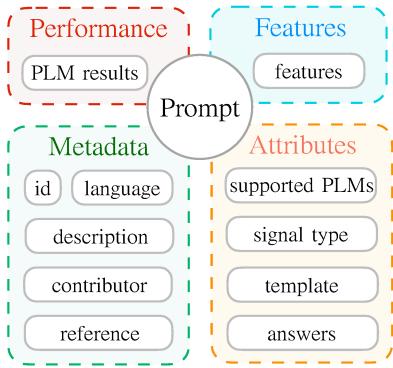


Figure 3: Prompt schema in DATALAB.

description of a research idea,¹⁶ compares it with descriptions of thousands of datasets, and displays the datasets best matching the input (a detailed example is given in Figure 4). This retrieval system goes beyond keyword search by using semantic matching. The algorithm is described in a pending paper; we provide technical details in Appendix.

3.4 Global Vision Analysis

Language Map: Which languages’ datasets get less attention? A language map is used to analyze which languages are more studied and which are less studied from a geographical view (Faisal et al., 2021), identifying potential systemic inequalities. Specifically, we first count how many datasets are available for each language. Then for each country we calculate a distribution over languages,¹⁷ where the ratio of each language represents the proportion of people who speak that language. Finally, for each country, we can get the weighted average number of datasets available for it in terms of its spoken languages (see Appendix for details).

4 Case Study

We perform three case studies to show the utility of DATALAB and put more in the appendix.

Artifacts One famous example of a dataset artifact reported by Gururangan et al. (2018) (Figure 1) is that in NLI datasets, the length of the hypothesis sentence is closely associated with the assigned label of the premise-hypothesis pair. In fact, DATALAB is able to easily re-discover this

¹⁶DATALAB also supports keyword queries as input. However we find the added context provided by natural language descriptions improves search quality.

¹⁷We refer to some official statistics from [this link](#).

artifact, and more. Fig. 4-(a) shows an analysis on the SNLI dataset (Bowman et al., 2015) between two features $\text{length}_{\text{hypothesis}}$ and label (entailment, neutral or contradiction). We can observe that, when $\text{length}_{\text{hypothesis}}$ is larger than 8.4, $\text{PMI}(\text{label}_{\text{neutral}}, \text{length}_{\text{hypothesis}}) > 0.28$, suggesting that “long hypotheses” tend to co-occur with the “neutral” label, even without consideration of the premise. Additionally, when $\text{length}_{\text{hypothesis}} \in [1, 4.7]$, $\text{PMI}(\text{label}_{\text{entailment}}, \text{length}_{\text{hypothesis}}) = 0.359$, implying that “short hypotheses” tend to co-occur with the label “entailment”. However, this is not all; **we further observed more than ten potential artifacts on SNLI and another popular dataset SST2** (Socher et al., 2013) (see Appendix), which demonstrates the ability of DATALAB to efficiently identify these artifacts.

Systemic Inequalities Fig. 4-(b) is a statistic of the degree to which languages are studied from a global (w.r.t each country in the world) perspective, with a darker red indicating more datasets studied/-constructed for the languages spoken in a given country, and darker blue indicating the opposite. Unsurprisingly, we observe that *English* is the most studied (large English-speaking countries like the US, Canada, and UK are in dark red), which also benefits those English-speaking African countries (e.g. Madagascar, Uganda, and Libya are in red.). We also observe that the languages spoken in *bm* (Mali), *ee* (Ghana), and *kr* (Niger) are rarely studied, as can be seen from our language map that these three languages have a value of 0.

Gender Bias We also showcase the gender bias analysis on SNLI as illustrate in Fig. 5. We can find that the samples in the SNLI dataset contain more male-oriented words than females ($\text{male}(0.62) > \text{female}(0.38)$).

Dataset Recommendation Fig. 4-(c) presents a case study of using our DATALAB to get recommended datasets. When a user enters a research idea “*I want to train a model that can recognize the positive and negative sentiments contained in a beer review.*”, DATALAB returns the beer review dataset BeerAdvocate (McAuley and Leskovec, 2013) first in the interface, which is a precise result since the dataset consists of beer reviews from [beeradvocate](http://www.beeradvocate.com/).¹⁸

¹⁸<http://www.beeradvocate.com/>

label / hypothesis_length (all count:50000)	1-4.7 (count:7104)	4.7-8.4 (count:28726)	8.4-12.1 (count:10893)	12.1-15.8 (count:2216)	15.8-19.5 (count:822)	19.5-23.2 (count:178)	23.2-26.9 (count:35)	26.9-30.6 (count:15)	30.6-34.3 (count:7)	34.3-38.1 (count:4)
entailment (count:16705)	0.359	0.057	-0.380	-0.548	-0.451	-0.502	-0.156	-0.513	-0.156	-0.290
neutral (count:16512)	-0.388	-0.122	0.280	0.483	0.469	0.627	0.443	0.702	0.548	0.415
contradiction (count:16783)	-0.120	0.053	-0.003	-0.212	-0.259	-0.624	-0.518	-0.923	-0.854	-0.295

(a) PMI analysis between two features: hypothesis length and label for the SNLI

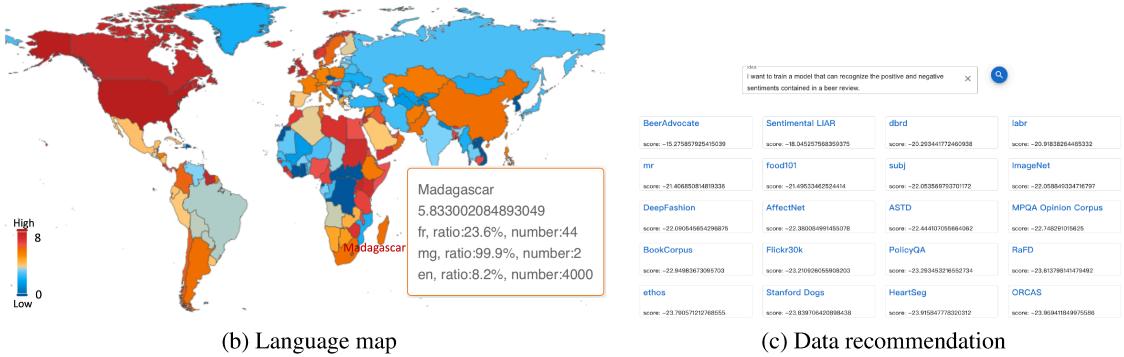


Figure 4: Case studies on artifact detection, systemic inequality, and dataset recommendation.

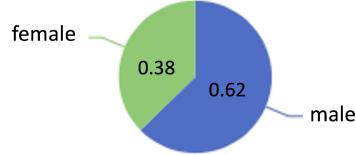


Figure 5: Gender bias analysis on SNLI.

5 Implications and Roadmap

DATALAB was born from our two visions (1) It is essential to standardize both the format of data and the interface of data-centric operations. (2) The standardization of data and operations allows more people in the community to contribute and share community wisdom. For example, in DATALAB, community researchers can easily contribute (1) new feature functions that enable us to conduct data analysis from more dimensions; (2) new datasets or the missing metadata. We hope that the unity of the platform can make it easier for *collective wisdom* to come into play.

In the future, we will extend DATALAB more broadly in terms of following perspectives: (1) index more data with different domains such as scientific and medical ones, and different modalities, such as image, video. (2) refine the current data typology over time, (3) add more built-in feature functions (e.g., labeling function (Ratner et al.,

2020)). (4) introduce more effective data management methods into DATALAB, such as FAIR.¹⁹

Acknowledgements

We thank all reviewers for their valuable comments and Antonis Anastasopoulos for sharing the mapping data between countries and languages, and thank Alissa Ostapenko, Yulia Tsvetkov, Jie Fu, Ziyun Xu, Chih-hao Wang, Lyuyang Hu, Yiran Chen, Zhengzheng Tang, Hiroaki Hayashi, Hector Liu, Xiachong Feng, Zhengfu He, Mingzhe Du, and Xiaodong Li for useful discussion and suggestions. This work was supported in part by Grant No. 2040926 of the US National Science Foundation, and the National Science Foundation of Singapore under its Industry Alignment Fund – Pre-positioning (IAF-PP) Funding Initiative. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not reflect the views of the National Research Foundation of Singapore and the US National Science Foundation.

Ethics/Broader Impact Statement

If the platform works as expected, researchers, developers, and analysts can all benefit from it. Re-

¹⁹ <https://www.go-fair.org/fair-principles/>

searchers can gain a deeper and broader understanding of the characteristics of the datasets, developers can more easily access the datasets and manipulate the data samples, and analysts can see some social insights from the datasets.

During the whole data analysis process, we tried to make it as transparent as possible, and the results of the analysis were well-grounded on sufficient evidence so that users could more reliably use it. Additionally, users are encouraged to report a case where the annotation results are not precise.

For those private datasets, we introduce three processing strategies for users: (1) users could flexibly use DATALAB SDK to deal with their data offline. (2) We introduce an account system²⁰ for users to set up a private space for their datasets. (3) We also open-source the code of DATALAB web platform²¹ so that users can host this web service locally.

As no feature analysis method will ever be perfectly reliable,²² to alleviate this issue, so far, DATALAB SDK provides several built-in feature functions for users to choose from and also make it customized by users. We are now working on benchmarking different feature functions in different domains of data so that users can have a better idea when choosing.

Currently, DATALAB only supports public datasets. In addition, knowing more about the characteristics of the test sets might make overfitting easier for model training. One possible approach is through multi-dataset evaluation, i.e., a good system should achieve good results across a series of different datasets.

DATALAB can privilege some datasets and data sources over others. To alleviate this problem, we introduce the dataset FINDER²³ functionality, where users can retrieve the dataset either from DATALAB and external resources like paperswithcode datasets²⁴ based on their requirement. Additionally, we will continue to keep the data corpus updated. For example, we recently added datasets from ACL2022.

Although DATALAB categorizes datasets into common tasks and paradigms, it still retains the

original information provided by the creators of each dataset. For example, the text to be classified in `imdb` is named as “review” while in `ag_news` is named as “sentence”. We introduce an intermediate variable²⁵ “text_column” to store the dataset-dependent naming information.

We appreciate any suggestions you have on how to make DATALAB better. Your issues are highly welcome,²⁶ and we will actively update.

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. [Deep learning for hate speech detection in tweets](#). *CoRR*, abs/1706.00188.

Yonatan Belinkov and James Glass. 2019. [Analysis methods in neural language processing: A survey](#). *Transactions of the Association for Computational Linguistics*, 7:49–72.

Damián Blasi, Antonios Anastasopoulos, and Graham Neubig. 2021. Systematic inequalities in language technology performance across the world’s languages. *arXiv preprint arXiv:2110.06733*.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. [Language \(technology\) is power: A critical survey of “bias” in NLP](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. [A large annotated corpus for learning natural language inference](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel Weld. 2020. [TLDR: Extreme summarization of scientific documents](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777, Online. Association for Computational Linguistics.

Yiran Chen, Pengfei Liu, Ming Zhong, Zi-Yi Dou, Danning Wang, Xipeng Qiu, and Xuanjing Huang. 2020. [CDEvalSumm: An empirical study of cross-dataset evaluation for neural summarization systems](#). In [https://github.com/ExpressAI/DataLab/tree/main/src/datalabs/tasks](#)

²⁰<https://datalab.nlpedia.ai/user>

²¹https://github.com/ExpressAI/DataLab_web

²²Thank reviewers for raising the discussion of this issue.

²³https://datalab.nlpedia.ai/dataset_recommendation

²⁴<https://paperswithcode.com/datasets>

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 3679–3691, Online. Association for Computational Linguistics.

James Clarke, Vivek Srikumar, Mark Sammons, and Dan Roth. 2012. [An NLP curator \(or: How I learned to stop worrying and love NLP pipelines\)](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 3276–3283, Istanbul, Turkey. European Language Resources Association (ELRA).

Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. [Automated hate speech detection and the problem of offensive language](#). In *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017, Montréal, Québec, Canada, May 15-18, 2017*, pages 512–515. AAAI Press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Kaustubh D Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahendiran, Simon Mille, Ashish Srivastava, Samson Tan, et al. 2021. [NL-augmenter: A framework for task-sensitive natural language augmentation](#). *arXiv preprint arXiv:2112.02721*.

Emily Dinan, Angela Fan, Ledell Wu, Jason Weston, Douwe Kiela, and Adina Williams. 2020. [Multi-dimensional gender bias classification](#). *CoRR*, abs/2005.00614.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. [Data augmentation for low-resource neural machine translation](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada. Association for Computational Linguistics.

Fahim Faisal, Yinkai Wang, and Antonios Anastasopoulos. 2021. [Dataset geography: Mapping language data to language users](#). *arXiv preprint arXiv:2112.03497*.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard H. Hovy. 2021. [A survey of data augmentation approaches for NLP](#). *CoRR*, abs/2105.03075.

Jinlan Fu, Pengfei Liu, and Graham Neubig. 2020a. [Interpretable multi-dataset evaluation for named entity recognition](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6058–6069. Association for Computational Linguistics.

Jinlan Fu, Pengfei Liu, Qi Zhang, and Xuanjing Huang. 2020b. [RethinkCWS: Is Chinese word segmentation a solved task?](#) In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5676–5686, Online. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. [AllenNLP: A deep semantic natural language processing platform](#). In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Google. 2021. [Explore datasets in know your data](#).

Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, et al. 2020. [Gluoncv and gluonnlp: Deep learning in computer vision and natural language processing](#). *J. Mach. Learn. Res.*, 21(23):1–7.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. [Annotation artifacts in natural language inference data](#). *CoRR*, abs/1803.02324.

Matthew Honnibal and Ines Montani. 2017. [spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing](#). To appear.

Robin Jia and Percy Liang. 2017. [Adversarial examples for evaluating reading comprehension systems](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. [Billion-scale similarity search with gpus](#). *arXiv preprint arXiv:1702.08734*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Taku Kudo. 2020. [Sentence piece](#). <https://github.com/google/sentencepiece>.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Sasko, Gunjan Chhablani, Bhavitya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid,

Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander M. Rush, and Thomas Wolf. 2021. [Datasets: A community library for natural language processing](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2021, Online and Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 175–184. Association for Computational Linguistics.

Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zachary C Lipton. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Pengfei Liu, Jinlan Fu, Yang Xiao, Weizhe Yuan, Shuaichen Chang, Junqi Dai, Yixin Liu, Zihuiwen Ye, and Graham Neubig. 2021a. [ExplainaBoard: An explainable leaderboard for NLP](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 280–289, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021b. [Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing](#). *CoRR*, abs/2107.13586.

Zhengzhong Liu, Guanxiong Ding, Avinash Bukkittu, Mansi Gupta, Pengzhi Gao, Atif Ahmed, Shikun Zhang, Xin Gao, Swapnil Singhavi, Linwei Li, Wei Wei, Zecong Hu, Haoran Shi, Xiaodan Liang, Teruko Mitamura, Eric P. Xing, and Zhiting Hu. 2021c. [A data-centric framework for composable NLP workflows](#). *CoRR*, abs/2103.01834.

Edward Loper and Steven Bird. 2002. [Nltk: The natural language toolkit](#). *arXiv preprint cs/0205028*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Benjamin Marie, Atsushi Fujita, and Raphael Rubino. 2021. [Scientific credibility of machine translation research: A meta-evaluation of 769 papers](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7297–7306, Online. Association for Computational Linguistics.

Julian John McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, pages 897–908.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. [Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Amandalynne Paullada, Inioluwa Deborah Raji, Emily M Bender, Emily Denton, and Alex Hanna. 2021. Data and its (dis) contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11):100336.

Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason A. Fries, Sen Wu, and Christopher Ré. 2020. [Snorkel: rapid training data creation with weak supervision](#). *VLDB J.*, 29(2-3):709–730.

Marco Túlio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2021. [Beyond accuracy: Behavioral testing of NLP models with checklist \(extended abstract\)](#). In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4824–4828. ijcai.org.

Brian Richards. 1987. [Type/token ratios: what do they really tell us?](#) *Journal of Child Language*, 14(2):201–209.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. [Multi-task prompted training enables zero-shot task generalization](#).

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. *The risk of racial bias in hate speech detection*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. *Recursive deep models for semantic compositionality over a sentiment treebank*. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Priyam Tejaswin, Dhruv Naik, and Pengfei Liu. 2021. *How well do you know your summarization datasets?* In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3436–3449, Online. Association for Computational Linguistics.

TFData. 2021. *Tensorflow datasets, a collection of ready-to-use datasets*.

Alexander Tsisis. 2002. *Destructive messages: How hate speech paves the way for harmful social movements*, volume 27. NYU Press.

Xiao Wang, Qin Liu, Tao Gui, Qi Zhang, Yicheng Zou, Xin Zhou, Jiacheng Ye, Yongxin Zhang, Rui Zheng, Zexiong Pang, Qinzhao Wu, Zhengyan Li, Chong Zhang, Ruotian Ma, Zichu Fei, Ruijian Cai, Jun Zhao, Xingwu Hu, Zhiheng Yan, Yiding Tan, Yuan Hu, Qiyuan Bian, Zhihua Liu, Shan Qin, Bolin Zhu, Xiaoyu Xing, Jinlan Fu, Yue Zhang, Minlong Peng, Xiaoqing Zheng, Yaqian Zhou, Zhongyu Wei, Xipeng Qiu, and Xuanjing Huang. 2021. *TextFlint: Unified multilingual robustness evaluation toolkit for natural language processing*. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 347–355, Online. Association for Computational Linguistics.

Jason W. Wei and Kai Zou. 2019. *EDA: easy data augmentation techniques for boosting performance on text classification tasks*. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6381–6387. Association for Computational Linguistics.

Lee Xiong, Chuan Hu, Chenyan Xiong, Daniel Fernando Campos, and Arnold Overwijk. 2019. Open domain web keyphrase extraction beyond language modeling. In *EMNLP*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018a. *Gender bias in coreference resolution: Evaluation and debiasing methods*. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 15–20. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018b. *Gender bias in coreference resolution: Evaluation and debiasing methods*. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 15–20. Association for Computational Linguistics.

Ming Zhong, Danding Wang, Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2019. *A closer look at data bias in neural extractive summarization models*. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 80–89, Hong Kong, China. Association for Computational Linguistics.

A Appendix

A.1 Detailed Statistics of DATALAB.

Here, we list more detailed statistics of DATALAB in [Table 3](#).

Aspect	Number
Tasks	142
Plain datasets	1,715
Diagnostics datasets	3,583
Language	331
Organization	794
Prompts	1,007
Aggregate	8
Preprocess	4
Operation	Featurize 16 Edit 23 Prompt 32
Feature	Sample level 138 Dataset level 180
Bias analysis	Hate speech datasets 240 Gender bias datasets 241 Gender bias samples 18,520,130 Hate speech samples 18,511,763
Annotated Datasets	728
Annotated samples	139,570,057
Total samples	408,460,905

Table 3: More detailed statistics of the DATALAB. “Diagnostic Dataset” refers to a dataset obtained by applying transformations to the original version.²⁷ “Annotated” indicates datasets or samples where we compute features to obtain additional information that is not originally present in the dataset.

A.2 Features

Features (e.g., sentence length) allow us to understand the characteristics of a dataset from different perspectives. Following Fu et al. (2020a), we define 318 features for 142 NLP tasks. Below, we list some core features at the sample- and dataset-level and suitable tasks.

A.2.1 Sample-level

General Features General features are task-agnostic and suitable for all NLP tasks.

- **Sentence length:** the number of tokens in a sentence.
- **Part-of-speech tags:** the part-of-speech tag for each token is automatically labeled by NLTK (Loper and Bird, 2002) Python tool.
- **Named entities:** entity names are automatically recognized by NLTK and SpaCy (Honnibal and Montani, 2017) Python tools.
- **Basic words ratio:** the proportion of words that appear in the basic English dictionary²⁸.
- **Lexical richness** (Richards, 1987): the proportion of unique words, obtained by dividing the number of unique words by the total number of words.
- **OOV density:** the proportion of words in a test sentence that do not appear in the training set.

Specialized Features In addition to general features, we also design task-specific features for some core NLP tasks. Below, we list some key task-specific features, as well as applicable tasks.

- **Span length:** the length of span. Span can be entity/answer/chunk/aspect. (NER, QA, Chunking, ABSA)
- **Label consistency of span** (Fu et al., 2020a): the visibility of a span and its label in the training set. (NER, Chunking)
- **Span frequency:** the frequency of entities in the training set. (NER, Chunking)
- **Span density:** the number of words belonging to entities in a sentence divided by the length of the sentence. (NER, Chunking)
- **Text similarity:** measures how similar two texts are. Here, we explore BLEU (Papineni et al., 2002) and ROUGE2 (Lin, 2004) for two texts. (SUMM, Match, QA)
- **Text length comparison:** measures the sentence-length relationship of sentence pairs, including

addition, subtraction, and division operation of sentence lengths. (Match, SUMM, QA)

- **Answer/span position:** measures where the answer/span starts in the text. (QA, ABSA, Chunking)
- **Coverage ratio:** measures to what extent a summary covers the content in the source text. (SUMM)
- **Copy length:** the average length of segments in a summary copied from the source document. (SUMM)

The full names of the tasks mentioned above are as follows:

- NER: Named Entity Recognition
- Chunking: Chunking
- POS: Part-of-speech Tagging
- ABSA: Aspect-Based Sentiment Analysis
- QA: Question Answering
- Matching: Text Matching
- SUMM: Text Summarization

A.2.2 Dataset-level

- **Average on dataset-level:** a sample-level feature can be converted into a dataset-level feature by averaging that feature of each sample in the dataset (e.g. the average text length, the average span length).
- **Distribution of vocabulary:** measured by the word frequency of each word in the dataset.
- **Distribution of label:** characterize the number of samples contained in each category in the dataset.
- **Sample size of different splits:** characterize the number of samples contained in different splits.
- **Hate speech ratio:** characterize the degree of hate speech bias of the dataset.
- **Spelling errors ratio:** measures the extent of spelling errors contained in a dataset with the help of a detection tool²⁹.

A.3 Bias

PMI for Sentiment Classification Taking the sentiment classification task as an example, we can use PMI to detect whether sentence length can indicate sentiment polarity. Given a sentence length sequence $L = \{l_1, l_2, \dots, l_n\}$ with n sentences, and a category sequence $C = \{c_1, c_2, \dots, c_m\}$ with m categories, the correlation measure PMI between sentence length and category can be defined

²⁸wikipedia.basic_words

²⁹spelling_error_detect_tool

Observation	Conclusion
SNLI	
$\text{len}_{\text{hp}} > 8.4$, $\text{PMI}(\text{label}_{\text{neutral}}, \text{len}_{\text{hp}}) > 0.28$; $\text{len}_{\text{hp}} \in [1, 4.7]$, $\text{PMI}(\text{label}_{\text{entailment}}, \text{len}_{\text{hp}}) = 0.359$;	Long hypotheses tend to be neutral. Short hypotheses tend to be entailment.
$\text{flesch_reading_ease}_{\text{hp}} \in [-50, 1.352]$; $\text{PMI}(\text{label}_{\text{entailment}}, \text{flesch_reading_ease}_{\text{hp}}) > 0.585$;	When the hypothesis is difficult enough to read, the sample tends to be labeled as entailment.
$\text{male}_{\text{hp}} > 2$, $\text{PMI}(\text{label}_{\text{neutral}}, \text{male}_{\text{hp}}) > 0.317$; $\text{female}_{\text{hp}} > 2$, $\text{PMI}(\text{label}_{\text{neutral}}, \text{male}_{\text{hp}}) > 0.377$;	Hypotheses with gender bias words (male/female) tend to be neutral.
$X = \text{len}_{\text{pm}} - \text{len}_{\text{hp}}$, if $X \in [8, 30]$, $\text{PMI}(\text{label}_{\text{entailment}}, \text{len}_{\text{pm}} - \text{len}_{\text{hp}}) > 0.084$; while $X \in [0, 7]$; $\text{PMI}(\text{label}_{\text{neutral}}, \text{len}_{\text{pm}} - \text{len}_{\text{hp}}) = 0.045$	When the length difference of hypothesis and premise is small enough ([0,7]), the sample tends to be entailment, and when it is large enough ([8,30]) the sample tends to be entailment.
$X = \text{len}_{\text{pm}} + \text{len}_{\text{hp}}$, if $X \in [4, 13]$, $\text{PMI}(\text{label}_{\text{entailment}}, \text{len}_{\text{pm}} + \text{len}_{\text{hp}}) = 0.259$; if $X > 22$, $\text{PMI}(\text{label}_{\text{neutral}}, \text{len}_{\text{pm}} + \text{len}_{\text{hp}}) > 0.105$;	When the sum of the lengths of hypothesis and premise is small enough, the sample tends to be entailment, and when it is large enough it tends to be neutral.
$X = \text{len}_{\text{pm}}/\text{len}_{\text{hp}}$, if $X < 2$, $\text{PMI}(\text{label}_{\text{neutral}}, \text{len}_{\text{pm}}/\text{len}_{\text{hp}}) > 0.094$; if $X > 2$, $\text{PMI}(\text{label}_{\text{entailment}}, \text{len}_{\text{pm}}/\text{len}_{\text{hp}}) > 0.141$;	When the lengths of hypothesis and premise are close enough, the samples tend to be neutral, and when their lengths are sufficiently different, samples tend to be entailment.
$\text{PMI}(\text{label}_{*}, \text{len}_{\text{pm}}) \approx 0$;	The length and gender features of the premise are irrelevance with the label.
SST2	
$\text{len}_{\text{sent}} < 7$, $\text{PMI}(\text{label}_{\text{positive}}, \text{len}_{\text{sent}}) = 0.06$ $\text{len}_{\text{sent}} > 7$, $\text{PMI}(\text{label}_{\text{negative}}, \text{len}_{\text{sent}}) > 0$	Sentences that are long enough tend to be negative, while sentences that are short enough tend to be positive.
$\text{female}_{\text{sent}} \in [4.8, 5.4]$, $\text{PMI}(\text{label}_{\text{positive}}, \text{female}_{\text{sent}}) = 0.58$ $\text{female}_{\text{sent}} < 0.6$, $\text{PMI}(\text{label}_{\text{negative}}, \text{female}_{\text{sent}}) = 0.021$ $\text{male}_{\text{sent}} < 1.2$, $\text{PMI}(\text{label}_{\text{positive}}, \text{male}_{\text{sent}}) = 0.018$ $\text{male}_{\text{sent}} > 1.2$, $\text{PMI}(\text{label}_{\text{negative}}, \text{male}_{\text{sent}}) > 0.068$	Sentences with low female bias tend to be negative, with high female bias tend to be positive; while sentences with high male bias tend to be negative.

Table 4: Observations and conclusions of bias analysis with PMI on the SNLI and GLUE-SST2 dataset. “hp” and “pm” denote *hypothesis* and *premise*, respectively. “len” is a function that computes the length of a sentence. “sent” denotes “sentence”.

as:

$$\phi_{\text{pmi}}(c_i, l_j) = \log\left(\frac{p(c_i, l_j)}{p(c_i)p(l_j)}\right), \quad (1)$$

where c_i and l_j denote the sentence length of the i -th sentence and the j -th category, respectively.

Gender Bias Given a male dictionary $K_{\text{male}} = [w_{m,1}, w_{m,2}, \dots, w_{m,k_1}]$ with k_1 words, female dictionary $K_{\text{female}} = [w_{f,1}, w_{f,2}, \dots, w_{f,k_2}]$ with k_2 words, and a dataset $D = [s_1, s_2, \dots, s_N]$ with N samples, the gender bias gb of dataset D can be defined as:

$$b_m = N_{\text{male}}/N, \quad (2)$$

$$b_f = N_{\text{female}}/N, \quad (3)$$

$$gb = b_m/b_f, \quad (4)$$

where b_m and b_f is the degree to which the dataset is biased towards men and towards women, respectively. N_{male} and N_{female} represent the number of words in the dataset D that appear in the dictionary K_{male} and the number of words in the dictionary K_{female} , respectively. N is the sample size of dataset D.

A.4 Calculation for Language Map

In language map, each country will be assigned a number that can be obtained by following steps: (1) for each country, collect the information that the languages spoken in this country and the proportion of people speaking each language. (2) for each data set, record the language of the data set (3) for each language, count the number of data set that belong to the language (4) for each language in the country,

multiply the ration of the language and the number of data set belong to the language. Finally sum the score of all languages in the country.

A.5 Customized Operation

```
from datalabs import load_dataset
from featurize import featurize

# Operation definition
@datalabs.feature
def get_length(text):
    return len(text.split(" "))

# Load dataset
dataset =
    load_dataset("ag_news") ["train"]
# Apply operation
res = dataset.apply(get_length)
```

A.6 Technical Implementation of Data Search

Our dataset search tool is designed to take as input a natural language description of a method and compare it against a search corpus of datasets.

We train our retrieval model with the Tevatron package.³⁰ The retrieval algorithm we use is effectively identical to Dense Passage Retrieval (DPR, Karpukhin et al. (2020)). Under this dual-encoder framework, the search corpus is indexed by encoding each document using the `CLS` embedding from BERT (Devlin et al., 2019). When our system receives a query, we first compute its embedding (again using the `CLS` embedding from BERT), then we rank the top documents using approximate nearest neighbor search (Johnson et al., 2017) on the shared inner product space of embeddings:

$$\text{score}(q, d) = \text{CLS}(\text{BERT}(q))^T \text{CLS}(\text{BERT}(d))$$

As a supervised learning-based retrieval method, this approach requires a large training set. To effectively generate a large training set, we adopt an automatic method for constructing annotations. We make the key observation that published AI/ML research papers reveal both a system description (contained in the abstract) as well as the datasets used to train or evaluate the system (usually found in the “Results” or “Experiments” section).

We use the abstracts of real papers as a proxy for natural language method descriptions, but we do not expect users to submit abstract-length queries into our system. Therefore, we pass these abstracts

through the “TLDR” scientific abstract summarization system (Cachola et al., 2020) to generate brief method descriptions.

We next automatically extract the datasets used by a given paper, which are used as a proxy for the relevant (positive) documents for each query during training. We extract these using a heuristic: for a given paper, if it mentions a dataset by name twice in the “Results”, “Experiments”, or “Methods” section and also cites the paper that introduces the dataset, we register this dataset as being used by the given paper. By manually inspecting 200 automatic dataset tags, we found over 90% of the tags from this method were correct.

We also support traditional keyword queries in our system. To support these queries, we duplicate each example in our training set to replace the natural language description “query” with a keyword query. To generate keyword queries, we pass the abstract through a keyphrase extraction system trained on OpenKP (Xiong et al., 2019). We then train a single retriever using a training set containing these two heterogenous types of queries.

³⁰<https://github.com/texttron/tevatron>