# LMDIFF: A Visual Diff Tool to Compare Language Models

**Hendrik Strobelt**
IBM Research
MIT-IBM Watson AI Lab
hendrik@strobelt.com

**Benjamin Hoover**
IBM Research
MIT-IBM Watson AI Lab
benjamin.hoover@ibm.com

**Arvind Satyanarayan**
MIT CSAIL
Massachusetts Institute of Technology
arvindsatya@mit.edu

**Sebastian Gehrmann**
Google Research
Harvard University
gehrmann@google.com

## Abstract

While different language models are ubiquitous in NLP, it is hard to contrast their outputs and identify which contexts one can handle better than the other. To address this question, we introduce LMDIFF, a tool that visually compares probability distributions of two models that differ, e.g., through finetuning, distillation, or simply training with different parameter sizes. LMDIFF allows the generation of hypotheses about model behavior by investigating text instances token by token and further assists in choosing these interesting text instances by identifying the most interesting phrases from large corpora. We showcase the applicability of LMDIFF for hypothesis generation across multiple case studies. A demo is available at http://lmdiff.net.

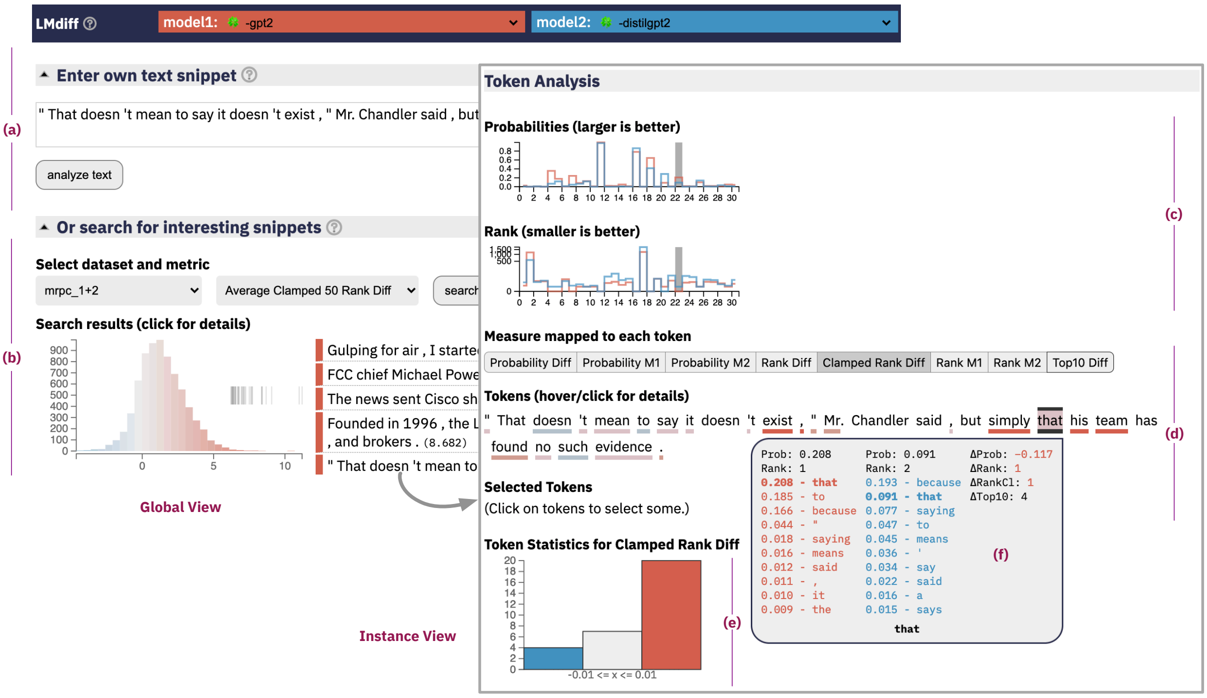
## 1 Introduction

Interactive tools play an important role when analyzing language models and other machine learning models in natural language processing (NLP) as they enable the qualitative examination of examples and help assemble anecdotal evidence that a model exhibits a particular behavior in certain contexts. This anecdotal evidence informs hypotheses that are then rigorously studied (e.g., Tenney et al., 2019; Belinkov and Glass, 2019; Rogers et al., 2020). Many such tools exist, for example to inspect attention mechanisms (Hoover et al., 2020; Vig, 2019), explain translations through nearest neighbors (Strobelt et al., 2018), investigate neuron values (Dalvi et al., 2019; Strobelt et al., 2017), and many more that focus on the outputs of models (e.g., Cabrera et al., 2019). There also exist multiple frameworks that aggregate methods employed in the initial tools to enable others to extend or combine them (Pruksachatkun et al., 2020; Wallace et al., 2019; Tenney et al., 2020).

However, notably absent from the range of available tools are those that aim to *compare* distributions produced by different models. While comparisons according to performance numbers are common practice in benchmarks (Wang et al., 2018; Hu et al., 2020; Gehrmann et al., 2021), there exists only rudimentary support in existing tools for inspecting how model outputs compare for specific tasks or documents. Yet, this problem motivates many current studies, including questions about how models handle gendered words, whether domain transfer is easy between models, what happens during finetuning, where differences lie between models of different sizes, or how multilingual and monolingual models differ.

To fill this gap, we introduce LMDIFF: an interactive tool for comparing language models by qualitatively comparing per-token likelihoods. Our design provides a *global* and a *local* view: In the global step, we operate on an entire corpus of texts, provide aggregate statistics across thousands of data points, and help users identify the most interesting examples. An interesting example can then be further analyzed in the local view. Fine-grained information about the model outputs for the chosen example is visualized, including the probability of each token and the difference in rank within each model's distribution. Similar to other visual tools, LMDIFF helps form hypotheses that can then be tested through rigorous statistical analyses. Across six case studies, we demonstrate how it enables an effective exploration of model differences and motivates future research. A deployed version of LMDIFF with six corpora and nine models is available at http://lmdiff.net/ and

*Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 96–105
November 7–11, 2021. ©2021 Association for Computational Linguistics



```
Prob: 0.208       Prob: 0.091       ΔProb: -0.117
Rank: 1           Rank: 2           ΔRank: 1
0.208 - that      0.193 - because   ΔRankCl: 1
0.185 - to        0.091 - that      ΔTop10: 4
0.166 - because   0.077 - saying
0.044 - "         0.047 - to
0.018 - saying    0.045 - means
0.016 - means     0.036 - '
0.012 - said      0.034 - say
0.011 - ,         0.022 - said
0.010 - it        0.016 - a
0.009 - the       0.015 - says
                that
```

Figure 1: LMDIFF interface. The Global View (a,b) allows finding interesting examples which are then selected for in-depth investigation in the Instance View (c-f).

its code is released at https://github.com/HendrikStrobelt/LMdiff (Apache 2.0 license) with support to easily add additional models, corpora, or evaluation metrics.

## 2 Methods

LMDIFF compares two models $m_{\{1,2\}}$ by analyzing their probability distributions at the position of each token $\hat{X}_{1:N}$ in a specific text. A correct token's probability distribution $p_{\mathrm{m_j}}(X_i = \hat{X}_i | X_{1:i-1})$ is easily influenced the scaling factor $\beta$ in the function $p = \mathrm{softmax}(\beta x)$ used to convert logits $x$ into probabilities $p$ (though two distributions are still comparable if both use the same $\beta$). For this reason, we also include the correct token's rank in $p_{\mathrm{m_j}}(X_i | X_{1:i-1})$. From the probabilities and ranks, we derive eight measures of global difference (comparison over a corpus) and eight measures of local difference (comparison over an example). The global measures are the (1) difference in rank of each token, (2) the difference in rank after clamping a rank to a maximum of 50, (3) the difference in probability of each token, and (4) the number of different tokens within the top-10 predicted tokens.[1] For each measure, we allow filtering by its average or maximum in a sequence.

To compare two models on a single example, we either directly visualize $p_{m_1}(X_i = \hat{X}_i)$, $p_{m_2}(X_i = \hat{X}_i)$, $p_{m_1}(X_i = \hat{X}_i) - p_{m_2}(X_i = \hat{X}_i)$, or the equivalent measures but focusing on the rank instead of the probability. As for the global measures, we present rank differences in both an unclamped and a clamped version. The clamped version surfaces more interesting examples; e.g., the difference between a token of rank 1 and 5 is more important than the rank difference between 44 and 60. The visual interface maps the difference to a blue-red scale (see Figure 1d) and visualizations of a single model to a gray scale.

### 2.1 Visual Interface

Figure 1 shows the LMDIFF interface. The user starts their investigation by specifying the two models $m_1$ and $m_2$ and a target text $d$. This target may either be entered into the free-text field (1a) or chosen from the list of suggested interesting text snippets (1b, see Section 2.2). Upon selection of the text, the likelihoods, ranks, and difference metrics for $m_1$ and $m_2$ for each token of $d$ are computed.

Users can compare results using the instance view, which leverages multiple visual idioms to show aspects of the models' performance. The step plots (Figure 1c) show the absolute values for likelihoods and ranks, with color indicating the model.

[1] Other metrics like the KL-Divergence were omitted from the final interface since the numbers were too hard to interpret.

| **Dataset** | **Description** |
|---|---|
| WinoBias<br>(Zhao et al., 2018) | Collection of 3,160 sentences using different resolution systems to understand gender bias issues. We include two versions: (a) just sentence, (b) sentence with addendum (e.g., “he refers to doctor”) |
| CommonsenseQA<br>(Talmor et al., 2019) | Collection of 12,102 questions with one correct answer and four distractor answers. For our use cases, we concatenate the question and the correct answer to one single string. |
| MRPC (Dolan et al., 2004) | Collection of 5,801 sentence pairs collected from newswire articles. |
| GPT2-GEN<br>(Radford et al., 2019a) | Collection of generated sentences from GPT-2 models. For each model the dataset contains 250K random samples (temperature 1, no truncation) and 250K samples generated with Top-K 40 truncation. We use the subset GPT-2-762M k40. |
| Short Jokes (Moudgil, 2017) | Collection of 231,657 short jokes provided as Kaggle challenge for humor understanding. |
| BioLang<br>(Liechti et al., 2017) | Collection of 12 million abstracts and captions from open access Europe PubMedCentral processed by the EMBO SourceData project |
| **Model** | |
| GPT-2 (Radford et al., 2019b) | The decoder of a Transformer trained on OpenWebText |
| DistilGPT-2 | A smaller Transformer trained to replicate GPT-2 output |
| GPT-2-ArXiv | GPT-2 finetuned on a large arxiv dataset |
| GPT-2-ArXiv-NLP | GPT-2 finetuned only on arxiv NLP papers |
| BERT-base-uncased<br>(Devlin et al., 2018) | Masked language model with case-insensitive tokenization. |
| DistilBERT (Sanh et al., 2019) | A smaller Transformer trained to replicate BERT output |
| DistilBERT-SST-2 | distilBERT finetuned on the SST-2 (Socher et al., 2013) dataset |
| GPT-2-German | GPT-2 trained on various German texts |
| GPT-2-German-Faust | The German GPT-2 model finetuned on Faust I & II |

Table 1: The default corpora and models found in the deployed version of LMDIFF. All models were taken from Huggingface’s model hub. Horizontal lines group tokenization-compatible models.

Upon selecting a distance metric, it is mapped onto the text (1d) using a red-white-blue diverging color scheme: white for no or minimal distance, red/blue for values in favor of a corresponding model. For instance, a token is colored blue if the rank of that token under model $m_2$ is lower than under $m_1$ or its likelihood higher. The highlighting on hover between both plots (1c+d) is synchronized, to help spot examples where the measures diverge.

The histogram (1e) indicates the distribution of measures for the text. If the centroid of the histogram leans decidedly to one side, it indicates that one model is better at reproducing the given text (observe the shift for red in Figure 1e). The token detail view (1f), shows all difference measures for a selected token and allows for a direct comparison of the top-k predictions for each model at the token position. E.g., in Figure 1f, the token “that” has rank 1 in model $m_1$ but rank 2 in $m_2$. Clicking tokens makes the detail view for those tokens stick to the bottom of the page to enable investigations of multiple tokens in the same sequence.

### 2.2 Finding Interesting Candidates

To facilitate searching for interesting texts, we extract examples from a large corpus of texts for which the two models differ the most. The corpus is prepared via an offline preprocessing step in which the differences between the models are scored according to the methods outlined above. Each example is compared using different aggregation methods, like averaging, finding the median, the upper quartile, or the top-k of differences in likelihoods, ranks, and clamped ranks. The 50 highest-ranking text snippets for each measure are considered as interesting. The interface (Figure 1b) shows a histogram of the distribution of a measure over the entire corpus and indicates through black stripes where interesting outlier samples are located fall on the histogram. That way, users can get an overview of how the two models compare across the corpus while also being able to view the most interesting samples.

## 3 Supported Data and Models

The deployed version of LMDIFF currently supports six datasets and nine models, detailed in Table 1. All pretrained models were taken from Hug-

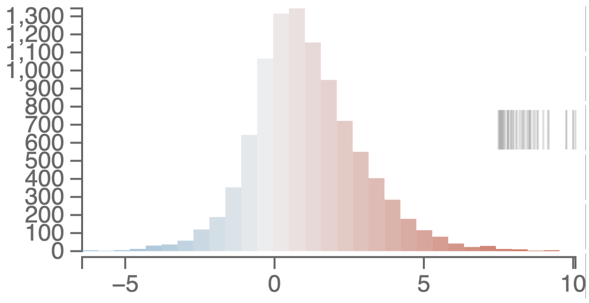


Figure 2: The global view on the CommonsenseQA dataset when comparing GPT-2 and DistilGPT-2. The histogram depicts the distribution of a specific measure (Average Clamped 50 Rank) over the reference corpus. The short black lines depict the values of the 20 highest values.

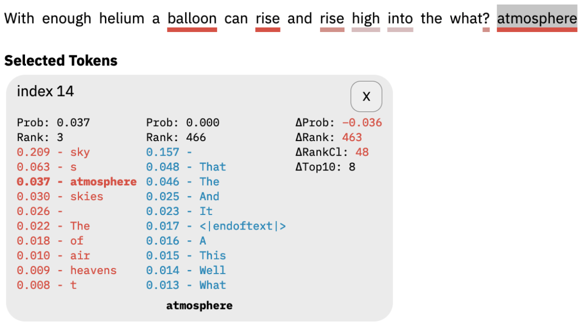
gingface's model hub[2]. Section 5 explains how to use LMDIFF with many more custom models and datasets.

## 4 Case Studies

As discussed above, this tool aims to generate hypotheses by discovering anecdotal evidence for certain model behavior. It will not be able to give definite proofs for discovered hypotheses, which should instead be explored more in-depth in follow-up studies. As such, in this section, we provide examples of new kinds of questions that LMDIFF helps investigate and explore further questions inspired by past findings.

### 4.1 Which model is better at commonsense reasoning?

Prompt-based approaches have become a popular way to test whether a model can perform a task (Brown et al., 2020). A relevant question to this is whether models can perform tasks that require memorization of commonsense knowledge (e.g., the name of the company that develops Windows, or the colors of the US flag) (Jiang et al., 2020). For our case study, we format the CommonsenseQA (Talmor et al., 2019) dataset to follow a "Question? Answer" schema, such that we can compare the probability of the answer under different models. Comparing GPT-2 (red) and its distilled variant DistilGPT-2 (blue), we can observe in Figure 2 that overall, GPT-2 performs much better on the task, commonly ranking the correct answer between 1 and 5 ranks higher in its distribution. An interesting example shown in Figure 3 paints a particularly grim picture for DistilGPT-2 — while the standard model ranks the correct answer third, the distilled variant ranks it 466th. This leads to the questions of why this bit of knowledge (and those of other outliers) was squashed in the distillation process, whether there is commonality between the forgotten knowledge, and it motivates the development of methods that prevent this from happening.

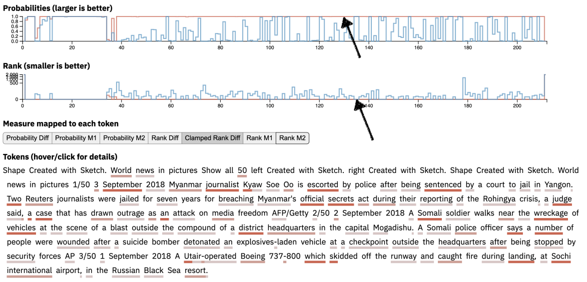

With enough helium a balloon can rise and rise high into the what? atmosphere

**Selected Tokens**

index 14

```
Prob: 0.037            Prob: 0.000                  ΔProb: -0.036
Rank: 3                Rank: 466                    ΔRank: 463
0.209 - sky            0.157 -                      ΔRankCl: 48
0.063 - s              0.048 - That                 ΔTop10: 8
0.037 - atmosphere     0.046 - The
0.030 - skies          0.025 - And
0.026 -                0.023 - It
0.022 - The            0.017 - <|endoftext|>
0.018 - of             0.016 - A
0.010 - air            0.015 - This
0.009 - heavens        0.014 - Well
0.008 - t              0.013 - What
                    atmosphere
```

Figure 3: A commonsenseQA example in which GPT-2 performs much better than DistilGPT-2. Showing Clamped Rank difference.



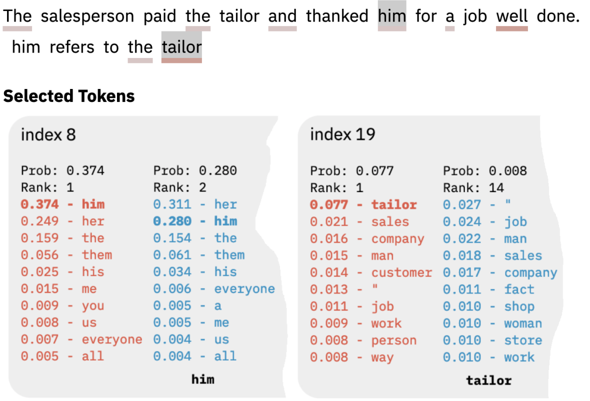
Shape Created with Sketch. World news in pictures Show all 50 left Created with Sketch. right Created with Sketch. Shape Created with Sketch. World news in pictures 1/50 3 September 2018 Myanmar journalist Kyaw Soe Oo is escorted by police after being sentenced by a court to jail in Yangon. Two Reuters journalists were jailed for seven years for breaching Myanmar's official secrets act during their reporting of the Rohingya crisis, a judge said, a case that has drawn outrage as an attack on media freedom AFP/Getty 2/50 2 September 2018 A Somali soldier walks near the wreckage of vehicles at the scene of a blast outside the compound of a district headquarters in the capital Mogadishu. A Somali police officer says a number of people were wounded after a suicide bomber detonated an explosives-laden vehicle at a checkpoint outside the headquarters after being stopped by security forces AP 3/50 1 September 2018 A Utair-operated Boeing 737-800 which skidded off the runway and caught fire during landing, at Sochi international airport, in the Russian Black Sea resort.

Figure 4: Comparing GPT-2 vs DistilGPT-2 on GPT-2 generated text shows that it is easy to spot which model produced it.

### 4.2 Which model produced a text?

Prior work has investigated different ways to detect whether a text was generated by a model or written by a human, either by training classifiers on samples from a model (Zellers et al., 2019; Brown et al., 2020) or directly using a models probability distribution (Gehrmann et al., 2019). A core insight from these works was that search algorithms (beam search, top-k sampling, etc.) tend to sample from the head of a models' distribution. That means that it is visually easy to detect if a model generated a text. With LMDIFF, we extend upon this insight to point to *which* model generated a text — if a model generated a text, the text should be consistently more likely under that model than under other similar models. While our tool does not allow us to test this hypothesis at scale, we can

[2] https://huggingface.co/models



The salesperson paid the tailor and thanked him for a job well done. him refers to the tailor

Selected Tokens

```
index 8                                index 19

Prob: 0.374     Prob: 0.280            Prob: 0.077          Prob: 0.008
Rank: 1         Rank: 2                Rank: 1              Rank: 14
0.374 - him     0.311 - her            0.077 - tailor       0.027 - "
0.249 - her     0.280 - him            0.021 - sales        0.024 - job
0.159 - the     0.154 - the            0.016 - company      0.022 - man
0.056 - them    0.061 - them           0.015 - man          0.018 - sales
0.025 - his     0.034 - his            0.014 - customer     0.017 - company
0.015 - me      0.006 - everyone       0.013 - "            0.011 - fact
0.009 - you     0.005 - a              0.011 - job          0.010 - shop
0.008 - us      0.005 - me             0.009 - work         0.010 - woman
0.007 - everyone 0.004 - us            0.008 - person       0.010 - store
0.005 - all     0.004 - all            0.008 - way          0.010 - work
                       him                                         tailor
```

Figure 5: Winobias example with addendum for GPT-2 vs DistilGPT-2 showing Clamped Rank difference. Interesting since him/her probability rank switches between models and only distil fails at the addendum task.

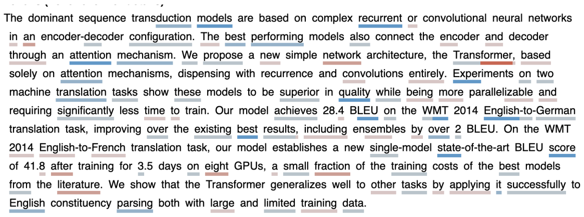
find clear anecdotal evidence shown in Figure 4. In the figure, we compare the probabilities of GPT-2 and DistilGPT-2 on a sample of GPT-2 generated text. We observe the consistent pattern that GPT-2 assigns an equal or higher likelihood to almost every token in the text.

### 4.3 Which model is more prone to be overconfident in coreference?

We next investigate whether one model has learned spurious correlations in coreference tasks, using our augmented version of the WinoBias dataset (Zhao et al., 2018). Since we are comparing language models, we modified the text to add the string "*[pronoun]* refers to the *[profession]*". We can then use the detail view to look at the probabilities of the pronoun in the original sentence and the probability of the disambiguating mention of the profession. In our example (Figure 5), we again compare GPT-2 (red) and DistilGPT-2 (blue). Curiously, the distillation process flipped the order of the predicted pronouns "him" and "her". Moreover, DistilGPT-2 fails to complete the second sentence while GPT-2 successfully predicts "Tailor" as the most probable continuation, indicating that DistilGPT-2 did not strongly associate the pronoun with the profession. This case study motivates further investigation of cases where distillation does not maintain the expected ranking of continuations. A similar effect has previously been detected in distillation processes for computer vision models (Hooker et al., 2020).

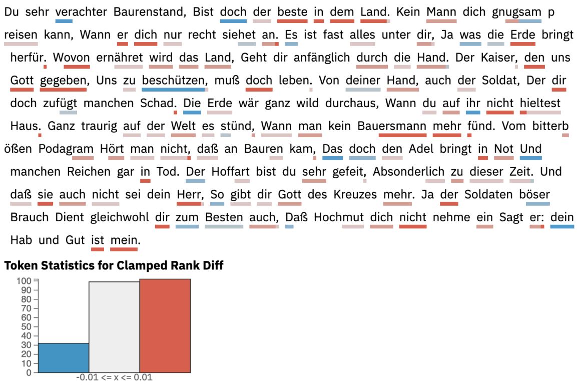
### 4.4 What predictions are affected the most by finetuning?

Other, more open-ended, qualitative comparisons that are enabled through LMDIFF aim to understand how a model changes when it is finetuned on a specific task or documents from a specific domain. The finetuning process can impact prediction both in the downstream domain and in not anticipated, unrelated other domains.



The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Figure 6: GPT-2 vs GPT-Arxiv-nlp on an abstract of an NLP paper.



Du sehr verachter Baurenstand, Bist doch der beste in dem Land. Kein Mann dich gnugsam p reisen kann, Wann er dich nur recht siehet an. Es ist fast alles unter dir, Ja was die Erde bringt herfür. Wovon ernähret wird das Land, Geht dir anfänglich durch die Hand. Der Kaiser, den uns Gott gegeben, Uns zu beschützen, muß doch leben. Von deiner Hand, auch der Soldat, Der dir doch zufügt manchen Schad. Die Erde wär ganz wild durchaus, Wann du auf ihr nicht hieltest Haus. Ganz traurig auf der Welt es stünd, Wann man kein Bauersmann mehr fünd. Vom bitterb ößen Podagram Hört man nicht, daß an Bauren kam, Das doch den Adel bringt in Not Und manchen Reichen gar in Tod. Der Hoffart bist du sehr gefeit, Absonderlich zu dieser Zeit. Und daß sie auch nicht sei dein Herr, So gibt dir Gott des Kreuzes mehr. Ja der Soldaten böser Brauch Dient gleichwohl dir zum Besten auch, Daß Hochmut dich nicht nehme ein Sagt er: dein Hab und Gut ist mein.

Figure 7: GPT2-German vs GPT2-German-Faust on a snippet from the 1668 book "Simplicius Simplicissimus" using the Clamped Rank difference.

**In Domain** In Figure 6, we show a comparison between GPT-2 and GPT-2-ArXiv-NLP on an abstract of an NLP paper, highlighting the probability difference. As expected, NLP-specific terms (WMT BLEU, model, attention, etc.) tend to be more likely under the finetuned model. But, interestingly, the name of languages and Transformer are both more likely under the original model. This finding may warrant a deeper investigation for possible causes and whether this phenomenon persists across other contexts.

**Out of Domain** Out-of-domain tests can be useful for checking whether the finetuning process led to some transfer learning, or to test for catastrophic forgetting. In our case study, we compare GPT-2-German before and after finetuning on Goethe's

Faust part I (1808) and II (1832). We hypothesized that the contemporary model would not be able to handle other works of literature from a similar time-period as well as the Faust-model, and thus tested on various snippets from books of the years 1200 to 1900. Our sample from the book Simplicius Simplicissimus (1668) (Figure 7) is representative of the consistent finding that GPT-2-German performs better than the Faust variant. This could have many reasons — the model may have overfit on the Faust-style of writing, the investigated periods of literature may differ too much, or they may differ too little from contemporary German.

### 4.5 Finding dataset errors

While not the original goal of LMDiff, we observed that in some cases the outlier detection method could also be used to find outlier *data* instead of examples where models differ significantly. One such example occurred when comparing GPT-ArXiv to GPT-2 on the BioLang dataset. It appears that GPT-2 is much better at modeling repetitive, nonsensical character sequences which were thus surfaced through the algorithm (see Appendix A).

## 5 System Description

All comparisons in LMDiff begin with three provided arguments: a dataset containing the interesting phases to analyze, and two comparable Transformer models. LMDiff wraps Huggingface Transformers (Wolf et al., 2020) and can use any of their pretrained autoregressive or masked language models from their model hub[3] or a local directory. Two models are comparable if they use the same tokenization scheme and vocabulary. This is required such that a phrase passed to either of them will have an identical encoding, with special tokens added in the same locations.

LMDiff then does the work of recording each model's predictions across the dataset into an *AnalysisCache*. Each token in each phrase of the dataset is analyzed for its "rank" and "probability". We define a token's rank as the affinity of the LM to predict the token relative to all other tokens, where a rank of 1 indicates it is the most favorable token, and the probability is computed from a direct softmax of the token's logit. Other useful information is also stored, such as the top-10 tokens (and their probabilities) that would have been predicted in that token's spot. This information can then be compared to other caches and explored in the visual interface. The interface can also be used independently of cache files to compare models on individual inputs.

The modular design separating *datasets*, *models*, and their *caches* makes it easy to compare the differences between many different models on distinct datasets. Once a cache has been made of a (model, dataset_D) pair, it can be compared to any other cache of a (comparable_model, dataset_D) pair within seconds. More information is provided in Appendix B.

**Adding models and datasets** It is easy to load additional models and datasets. First, ensure that the model can be loaded through the Huggingface `AutoModelWithLMHead` and `AutoTokenizer` function `from_pretrained(...)` which supports loading from a local directory. The following script prepares two models and a dataset for comparison:

```
python scripts/preprocess.py all \
    [OPTIONS] M1 M2 DATASET \
    --output-dir OUT
- M1 = Path (or name) of HF model 1
- M2 = Path (or name) of HF model 2
- DATASET = Path to dataset.txt
- OUT = Where to store outputs
```

The output configuration directory `OUT` can be passed directly to the LMDiff server and interface which will automatically load the new data:

```
python backend/server/main.py \
    --config DIR
- DIR = Contains preprocessed outputs
```

The interface works equally well to compare two models on individual examples without a preprocessed cache:

```
python backend/server/main.py \
    --m1 MODEL1 --m2 MODEL2
```

## 6 Discussion and Conclusion

We presented LMDiff, a tool to visually inspect qualitative differences between language models based on output distributions. We show in several use cases how finding specific text snippets and analyzing them token-by-token can lead to interesting hypotheses.

We emphasize that LMDiff by itself does not provide any definite answers to these hypotheses by itself – it cannot, for example, show which model

[3] https://huggingface.co/models

is generally better at a given task. To answer these kind of questions, statistical analysis is required.

A design limitation of LMDIFF is that it relies on compatible models. Because the tool is based on per-token model outputs and apples-to-apples comparisons of distributions, only models that use the same tokenization scheme and vocabulary can be compared in the instance view. In future work, we will work toward extending the compatibility by introducing additional tokenization-independent measures and visualizations.

Another extension of LMDIFF may probe for memorized training examples and personal information using methods proposed by Carlini et al. (2020). As shown in Sections 4.2 and 4.5, we can already identify text that was generated by a model and leverage patterns that a model has learned. Adding support to filter a corpus by measures in addition to finding outliers may help with the analysis of potentially memorized examples.

## 7 Acknowledgements

We thank Ankur Parikh and Ian Tenney for helpful comments on an earlier draft of this paper. This work was supported by the MIT-IBM Watson AI Lab. This work has been developed in part during the BigScience Summer of Language Models 2021.

## References

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

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.*

Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie Morgenstern, and Duen Horng Chau. 2019. Fairvis: Visual analytics for discovering intersectional bias in machine learning. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 46–56. IEEE.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2020. Extracting training data from large language models. *CoRR*, abs/2012.07805.

Fahim Dalvi, Avery Nortonsmith, Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, and James Glass. 2019. NeuroX: A toolkit for analyzing individual neurons in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9851–9852.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Chinenye Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Andre Niyongabo Rubungo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120, Online. Association for Computational Linguistics.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. 2020. Characterising bias in compressed models. *ICML 2020 Workshop on Human Interpretability in Machine Learning*.

Benjamin Hoover, Hendrik Strobelt, and Sebastian Gehrmann. 2020. exBERT: A Visual Analysis Tool to Explore Learned Representations in Transformer Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 187–196, Online. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multitask benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know. *Trans. Assoc. Comput. Linguistics*, 8:423–438.

Robin Liechti, Nancy George, Lou Götz, Sara El-Gebali, Anastasia Chasapi, Isaac Crespo, Ioannis Xenarios, and Thomas Lemberger. 2017. Sourcedata: a semantic platform for curating and searching figures. *Nature methods*, 14(11):1021–1022.

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

Abhinav Moudgil. 2017. Kaggle: Short jokes - collection of over 200,000 short jokes for humour research.

Yada Pruksachatkun, Phil Yeres, Haokun Liu, Jason Phang, Phu Mon Htut, Alex Wang, Ian Tenney, and Samuel R. Bowman. 2020. jiant: A software toolkit for research on general-purpose text understanding models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 109–117, Online. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, and Jeff Wu. 2019a. Gpt2 output dataset on github.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019b. Language models are unsupervised multitask learners.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

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.

Hendrik Strobelt, Sebastian Gehrmann, Michael Behrisch, Adam Perer, Hanspeter Pfister, and Alexander M Rush. 2018. Seq2Seq-Vis: A visual debugging tool for sequence-to-sequence models. *IEEE transactions on visualization and computer graphics*, 25(1):353–363.

Hendrik Strobelt, Sebastian Gehrmann, Hanspeter Pfister, and Alexander M Rush. 2017. LSTMvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks. *IEEE transactions on visualization and computer graphics*, 24(1):667–676.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Ian Tenney, James Wexler, Jasmijn Bastings, Tolga Bolukbasi, Andy Coenen, Sebastian Gehrmann, Ellen Jiang, Mahima Pushkarna, Carey Radebaugh, Emily Reif, and Ann Yuan. 2020. The language interpretability tool: Extensible, interactive visualizations and analysis for NLP models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 107–118, Online. Association for Computational Linguistics.

J. Vig. 2019. A multiscale visualization of attention in the transformer model. *ArXiv*, abs/1906.05714.

Eric Wallace, Jens Tuyls, Junlin Wang, Sanjay Subramanian, Matt Gardner, and Sameer Singh. 2019. AllenNLP interpret: A framework for explaining predictions of NLP models. 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): System Demonstrations*, pages 7–12, Hong Kong, China. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018.

GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 9051–9062.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. 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, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana. Association for Computational Linguistics.

# A Additional Case Studies

## A.1 Masked LMs break when fine-tuning on different tasks

When finetuning an autoregressive language model, the output representations are preserved since downstream tasks often make use of the language modeling objective. This is different for masked language models like BERT. Typically, the contextual embeddings are combined with a new untrained head and thus, the language modeling is ignored during finetuning. We demonstrate this in Figure 8 where we compare DistilBERT (blue) and DistilBERT-SST (red) on a recent abstract published in Science. DistilBERT performs much better, having a significantly higher probability for almost every token in the text. Since the finetuned model started with the same parameters, this is a particular instance of catastrophic forgetting (McCloskey and Cohen, 1989). While this case is somewhat obvious, LMDIFF can help identify domains that are potentially more affected by this phenomenon even for cases in which the language modeling objective is not abandoned.

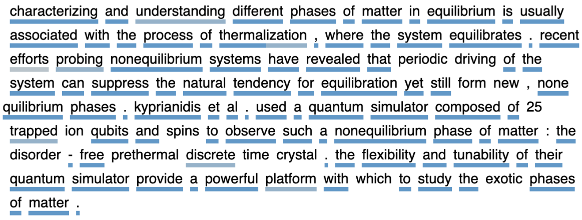
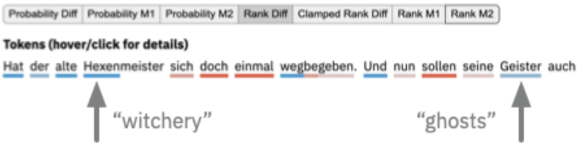

characterizing and understanding different phases of matter in equilibrium is usually associated with the process of thermalization , where the system equilibrates . recent efforts probing nonequilibrium systems have revealed that periodic driving of the system can suppress the natural tendency for equilibration yet still form new , none quilibrium phases . kyprianidis et al . used a quantum simulator composed of 25 trapped ion qubits and spins to observe such a nonequilibrium phase of matter : the disorder - free prethermal discrete time crystal . the flexibility and tunability of their quantum simulator provide a powerful platform with which to study the exotic phases of matter .

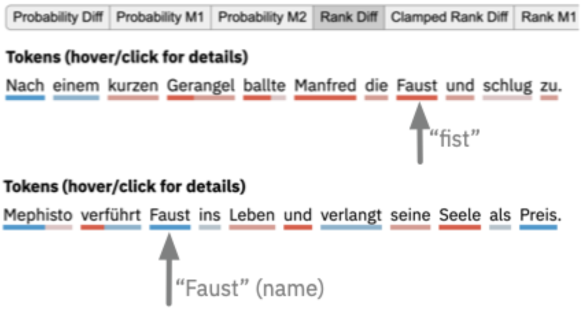
Figure 8: DistilBERT-SST vs DistilBERT on a scientific abstract.



**Tokens (hover/click for details)**

Hat der alte Hexenmeister sich doch einmal wegbegeben. Und nun sollen seine Geister auch

Figure 9: Magic characters more likely under GPT2-German-Faust



**Tokens (hover/click for details)**

Nach einem kurzen Gerangel ballte Manfred die Faust und schlug zu.

**Tokens (hover/click for details)**

Mephisto verführt Faust ins Leben und verlangt seine Seele als Preis.

Figure 10: Tokens can be more likely under different models depending on contexts.

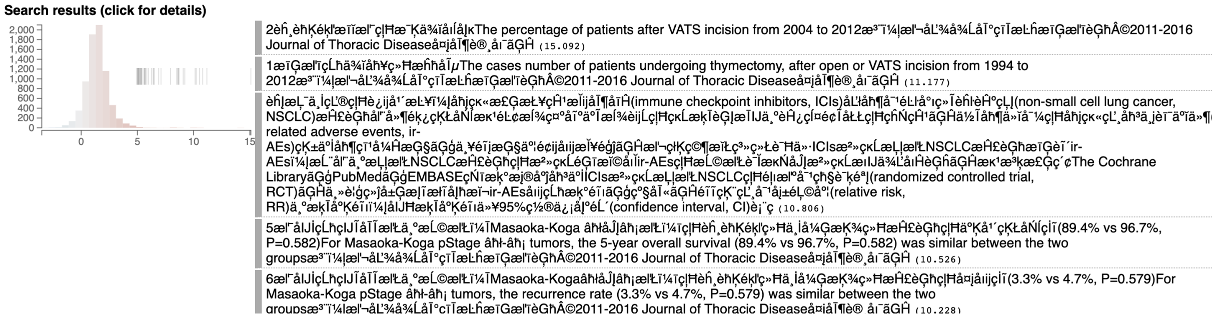
## A.2 Data Outliers

We show one example of a data outlier, described in Section 4.5, in Figure 11. The top-ranked examples in the corpus all have severe encoding errors and those examples should be removed from the corpus.

## A.3 Language specific to finetuned model

The comparison of GPT2-German and GPT2-German-Faust (see Section 4.4) also revealed more patterns that indicate that the fine-tuning of the model might have been successful. Figure 9 shows an example where tokens related to the core text of the Faust text are more likely under the fine-tuned model than the wild-type GPT2-German. Tokens like “Hexe” (witch) or “Geister” (ghosts) are core characters in the Faust text.

Another interesting observation is that even the



2èħ¸èħĶékľæĭĩæľ¯çļĦæ¯Ķä¾ĭåıĺåļĸThe percentage of patients after VATS incision from 2004 to 2012æ³¨ï¼ļæľ¬åĽ¾å¾Ĺåĩ°çĩĨæĿĥæĩĢæľĩèĢħÂ©2011-2016 Journal of Thoracic Diseaseå¤¡åĩ¶è®¸åı¯ãĢĤ (15.092)

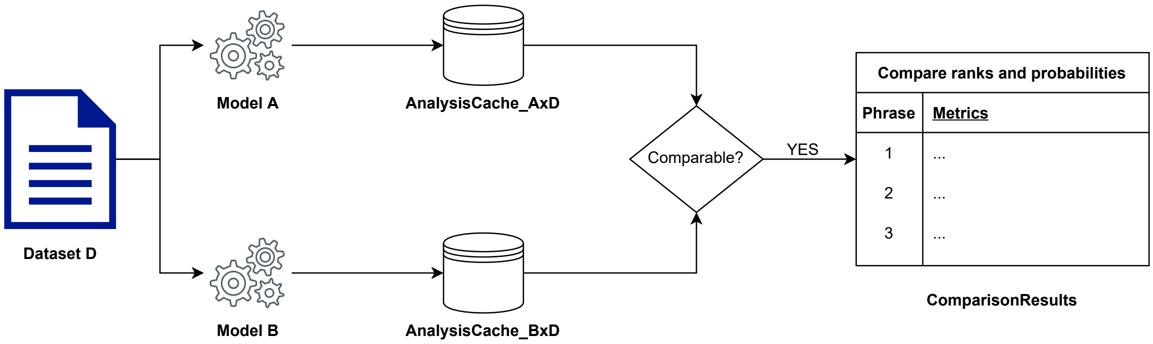
1æĩĢæľĩçĹħä¾ĭåħ¥ç»ĦæĥħåĨµThe cases number of patients undergoing thymectomy, after open or VATS incision from 1994 to 2012æ³¨ï¼ļæľ¬åĽ¾å¾Ĺåĩ°çĩĨæĿĥæĩĢæľĩèĢħÂ©2011-2016 Journal of Thoracic Diseaseå¤¡åĩ¶è®¸åı¯ãĢĤ (11.177)

èĥļæĽ¯ä¸İçĽ®çļĦè¿ijå¹´æĿ¥ï¼ļåħįçĸ«æ£ĢæŁ¥çĤ¹æĬijåĪ¶åīĤ(immune checkpoint inhibitors, ICIs)åĽłåħ¶å¯¹éĿŀå°ıç»ĨèĥŀèĥºçĻļ(non-small cell lung cancer, NSCLC)æĤ£èĢħåľ¨å»¶éķ¿çĶŁåŃĺæĸ¹éĿ¢æĺ¾ç¤ºåĩºåĩºæĺ¾èijĹçļĦçĸĹæķĪèĢļæĪIJä¸ºèĤ¿çĺ¤é¢ĨåŁŁçļĦçĥŃçĤ¹ãĢĤä½Ĩåħ¶ä»ïå¯¼çļĦåħįçĸ«çĽ¸åħ³ä¸jèī¯ä°ïä»¶(immune-related adverse events, ir-AEs)çĶ±äºĨåħ¶çī¹å¼ĤæĢ§ãĢģä¸¥éĩįæĢ§äº¦é¢ijåıijæĬ¥éģĵãĢĤæľ¬çłĶç©¶æĭŁç³»ç»Łè¯Ħä»·ICIsæ²»çĸĹæĻļæľŁNSCLCæĤ£èĢħæĩĢèĭ´ir-AEsï¼ļæĽ´åľ¨ä¸ºæĻļæľŁNSCLCæĤ£èĢħçļĦæ²»çĸĹéĢīæĭ©åıĬir-AEsçļĦæĽ©æľŁè¯ĨæĸŃåĴļæ²»çĸĹæıIJä¾ĽåıĤèĢħãĢĤæĸ¹æ³ķæ£Ģç´¢The Cochrane LibraryãĢģPubMedãĢģEMBASEçŃīæķ°æį®åºĵåħ³äºİICIsæ²»çĸĹæĻļæľŁNSCLCçļĦéļıæľºå¯¹çħ§è¯ķéª|(randomized controlled trial, RCT)ãĢĤä¸»èĭģç»Ĵå±Ģæļĩæłĩåļħæĭ¬ir-AEsåıijçĹħæķ°éĩıãĢģç°§åĨ«ãĢĤéĩĩçĶ¨çĽ¸å¯¹å¡±éĻ©åº¦(relative risk, RR)ä¸ºæķĪåºĶéĩıï¼ļåIJĦæķĪåºĶéĩıä»¥95%ç½®ä¿¡åļºéĹ´(confidence interval, CI)è¡¨ç (10.806)

5æľ¯åIJİçĹħçIJĨåĪĨæľŁä¸ºæĽ©æľŁï¼ĪMasaoka-Koga âħłåĴļâħ¡æľŁï¼īçļĦèĥ¸èħĶékľç»Ħä¸İå¼ĢæĶ¾ç»ĦæĤ£èĢħçļĦä°Ķå¹´çĶŁåŃĺçİĩ(89.4% vs 96.7%, P=0.582)For Masaoka-Koga pStage âħł-âħ¡ tumors, the 5-year overall survival (89.4% vs 96.7%, P=0.582) was similar between the two groupsæ³¨ï¼ļæľ¬åĽ¾å¾Ĺåĩ°çĩĨæĿĥæĩĢæľĩèĢħÂ©2011-2016 Journal of Thoracic Diseaseå¤¡åĩ¶è®¸åı¯ãĢĤ (10.526)

6æľ¯åIJİçĹħçIJĨåĪĨæľŁä¸ºæĽ©æľŁï¼ĪMasaoka-KogaâħłåĴļâħ¡æľŁï¼īçļĦèĥ¸èħĶékľç»Ħä¸İå¼ĢæĶ¾ç»ĦæĤ£èĢħçļĦå¤¡åıijçİĩ(3.3% vs 4.7%, P=0.579)For Masaoka-Koga pStage âħł-âħ¡ tumors, the recurrence rate (3.3% vs 4.7%, P=0.579) was similar between the two groupsæ³¨ï¼ļæľ¬åĽ¾å¾Ĺåĩ°çĩĨæĿĥæĩĢæľĩèĢħÂ©2011-2016 Journal of Thoracic Diseaseå¤¡åĩ¶è®¸åı¯ãĢĤ (10.228)

Figure 11: BioLang with GPT-2 vs the GPT-2-ArXiv. GPT-2 is much better at modeling repeated patterns which helps identify malformed examples.



| Compare ranks and probabilities | |
|---|---|
| Phrase | Metrics |
| 1 | ... |
| 2 | ... |
| 3 | ... |

Figure 12: System diagram of the LMDIFF backend.

same tokens in different contexts can be more likely under different models. The token "Faust" can refer to the name of the main character in the story or be the common German translation for "fist". Figure 10 shows how the word is more likely under the general language model if embedded in a fighting context versus being embedded in a one-sentence summary of the Faust story.

## B System diagram for corpus analyses

Figure 12 describes how LMDIFF identifies compatibility between models and precomputed corpora. The **Dataset** is a text file where each new line contains a phrase to analyze. It also contains a YAML header containing necessary information like its name and a unique hash of the contents. This *dataset* is processed by different Huggingface Transformer **Models** that receive the contents of the dataset as input and make predictions at every token. The tokenizations and predictions for each of the phrases are stored in the **AnalysisCache**, which takes the form of an HDF5 file. Finally, any two *AnalysisCaches* can be checked for comparability. If they are comparable, the difference between them can be summarized in a **ComparisonResults** table and presented through the aforementioned interface for inspection and exploration by the user.