What's in a (dataset's) name? The case of BigPatent

Silvia Casola
Università degli Studi di Padua
Fondazione Bruno Kessler, Trento
scasola@fbk.eu

Alberto Lavelli
Fondazione Bruno Kessler, Trento
lavelli@fbk.eu

Horacio Saggion
Universitat Pompeu Fabra, Barcelona
horacio.saggion@upf.edu

Abstract

Sharing datasets and benchmarks has been crucial for rapidly improving Natural Language Processing models and systems. Documenting datasets' characteristics (and any modification introduced over time) is equally important to avoid confusion and make comparisons reliable.

Here, we describe the case of BigPatent, a dataset for patent summarization that exists in at least two rather different versions under the same name. While previous literature has not clearly distinguished among versions, their differences not only lay on a surface level but also modify the dataset's core nature and, thus, the complexity of the summarization task.

While this paper describes a specific case, we aim to shed light on new challenges that might emerge in resource sharing and advocate for comprehensive documentation of datasets and models.

1 Introduction

Sharing models and datasets is essential for Natural Language Processing (NLP). With the rise of transfer learning in the last few years, releasing large pre-trained models has become standard practice. Consequently, several libraries have provided APIs to access and work with those models efficiently. Datasets have followed a similar trend: they are often shared by their authors and stored in hubs that expose APIs. Two notable examples of this trend are the TensorFlow Datasets collection[1] and the Hugging Face dataset library[2] (Lhoest et al., 2021). These libraries allow accessing published data, often with just a few lines of code. They drastically ease the experimentation loop, and allow users to download, experiment with, and probe existing resources. There is, however, another side to the coin: the dataset documentation is sometimes insufficient, which might lead to inconsistencies when performing experiments and comparing results to previous work.

This paper analyzes a somewhat extreme case: the BigPatent dataset (Sharma et al., 2019). BigPatent is a dataset for patent summarization, first published in 2019. Patents have many peculiar characteristics that might be challenging for standard NLP systems: they span multiple pages, have very long sentences, contain a mix of legal and technical vocabulary, and are built out of noun phrases instead of clauses, with a long lexical chain (Casola and Lavelli, 2022). Thus, the dataset has also become popular as a general benchmark for summarization.

We show that the two popular TensorFlow and Hugging Face dataset hubs expose different versions of BigPatent. These differences are not only superficial (e.g., casing, tokenization) but regard the very content of the source documents.

We first briefly describe this difference and its impact on the dataset features (Section 2); then, we examine previous work and show it hardly ever clarifies the version of the dataset used in experiments (Section 3); finally, we show how the difference substantially impacts models' performance (Section 4).

While strongly advocating for resource sharing and infrastructure that make them easier to use, we hope that the discussion of this extreme case can shed light on the importance of careful resource documentation.

2 The BigPatent dataset

BigPatent is a dataset for the automatic summarization of patent documents.
Patents award inventors the exclusive right to use, make, and sell their inventions for a specific time and geographical area. Patents are structured legal documents containing several sections. The De-

[1] https://www.tensorflow.org/datasets (Last accessed: September 2022)

[2] https://huggingface.co/docs/datasets/ (Last accessed: September 2022)

Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM), pages 399 - 404
December 7, 2022 ©2022 Association for Computational Linguistics

scription section reports the technical characteristics of the invention and its preferred embodiments so that a person skilled in the art can understand and reproduce it. The Description can be further divided into subsections (e.g., Background, Field of the Invention, Summary of the invention, Detailed Description, Description of the Drawings, etc.). The patent document also contains a human-written Abstract. It is thus somewhat natural to construct a summarization dataset using the Descriptions (or part of them) as the source texts and the Abstracts as the gold-standard summaries.

The dataset is not only interesting for a niche of patent mining researchers: in fact, patent documents show several interesting linguistic characteristics worth investigating (e.g., long sentences, unusual vocabulary, specific syntactic structure). Moreover, since many popular large-scale summarization datasets are in the news domain (Nallapati et al., 2016; Narayan et al., 2018; Fabbri et al., 2019), gathering data from different sources opens new challenges for NLP systems. For example, patent documents are very long, and their Abstract is not very extractive with respect to the Detailed Description, as the original dataset shows (Sharma et al., 2019).

In its original version, published by BigPatent's authors and accessible on GitHub[3], only part of the Description (typically the Detailed Description) is included in the input document, and the source does not contain any of the other subsections. The published dataset is also cased and tokenized. The Hugging Face dataset library exposes this version of the dataset (described in the related paper)[4].

With the advent of sequence-to-sequence transformer models for summarization (e.g., BART (Lewis et al., 2020) or Pegasus (Zhang et al., 2020))), however, using a strongly preprocessed dataset is not ideal. It is common practice to process the raw text with a model-specific tokenizer. This is likely why the TensorFlow Datasets collection contains a different version of the dataset that is cased and untokenized, with limited preprocessing over the original raw text[5].

However, a deeper look at the data reveals another difference: the TensorFlow source documents contain a superset of the text contained in the original version. All subsections in the patent Description are included. Thus, the input not only contains the Detailed Description but often also the Background, the Field of the invention, etc., and, interestingly, a Summary of the invention[6,7]. Table 1 shows the first tokens of the input of some entries in the corpus.

In the following, we compute some statistics on the two dataset versions (we call the original version $BigPatent_{Original}$ and the subsequent modified cased version $BigPatent_{New}$) and their different characteristics.

The dataset is divided into several subsets, following the Cooperative Patent Classification (CPC) codes. Due to the large dataset size (over 1.3 million examples), we restrict our analysis to its G (Physics) subset: it includes patents of information systems devices and processes, for which the authors of this paper might be considered skilled in the art. However, our considerations are general.

2.1 Dataset characteristics

Table 2 reports some statistics[8] over BigPatent/G. Note that the dataset split is identical in the two versions (i.e., the train, validation, and test splits contain the same documents). While the summaries characteristics are very similar between the original and the new version (we attribute the difference to errors in the tokenization, since $BigPatent_{Original}$ is pre-tokenized, while $BigPatent_{New}$ is not), $BigPatent_{New}$ clearly contains more text than the original version (38% more tokens, on average, in the training set), and more sentences (68% more, on average, in the training set). The compression ratio (i.e., the ratio between the number of tokens in the source and the number of tokens in the Abstract) is also higher in $BigPatent_{New}$.

To get a closer look at the datasets' abstractiveness, we compute their coverage and density, following Grusky et al. (2018).
Given a document $D = \langle d_1, d_2, \ldots, d_n \rangle$ where d_i is a token of D and a summary $S = \langle s_1, s_2, \ldots, s_m \rangle$, with $m \leq n$, where s_j is a token in the summary, $F(D, S)$ is the set of their shared

[3] https://evasharma.github.io/bigpatent/
(Last accessed: September 2022)

[4] https://huggingface.co/datasets/big_patent
(Last accessed: September 2022)

[5] https://www.tensorflow.org/datasets/catalog/big_patent
(Last accessed: September 2022)

[6] We will refer to this summary included in the document (input) as Summary of the Invention and to the dataset gold-standard as Abstract or gold standard.

[7] Note that this difference is not explicitly discussed on the dataset page.

[8] we use NLTK for sentence and word tokenization.

publication_number	$\text{Description}_{Original}$	Description_{New}
US-2007088503-A1	referring now to fig1 and 2 , a service technician visiting a customer service location is provided with a technician input device 2 for receiving and transmitting information related to a disruption or interruption of service at the service location . the input device 2 can be a wireless pc , for example , a laptop , a personal digital assistant (pda), a wireless pager or any other device suitable for receiving and transmitting data associated with providing service at the customer service location . [+2858 tokens]	This is a continuation of application Ser. No. 10/445,861 filed May 27, 2003, which is a continuation of application Ser. No. 10/032,853 filed Oct. 25, 2001 and now U.S. Pat. No. 6,772,064. The present methods and systems generally relate to processing and transmitting information to facilitate providing service in a telecommunications network. [+986 tokens] Referring now to FIG1 and 2 , a service technician visiting a customer service location is provided with a technician input device 2 for receiving and transmitting information related to a disruption or interruption of service at the service location. [+2427 tokens]
US-2011144953-A1	in the following , the invention is described in more detail referring to the attached figures by means of exemplary embodiments , wherein same reference signs refer to same components . fig1 schematically shows the system for compensating electromagnetic interfering fields . an object 2 to be protected against effects of the interfering field 1 is permeated by the interfering field 1 . here , the interfering field 1 is assumed to be a gradient field . the amplitude of the interfering field 1 is measured by two real magnetic field sensors 3 , and 4 . the first real sensor 3 provides an output signal right arrow over (s) 1 =[x 1 (t), y 1 (t), z 1 (t)], and the second real sensor 4 provides an output signal right arrow over (s) 2 =[x 2 (t), y 2 (t), z 2 (t)]. [+1855 tokens]	This application claims benefit under 35 U.S.C. (a) of German Patent Application No. 10 2009 024 826.9-32, filed Jun. 13, 2009, the entire contents of which are incorporated herein by reference.The invention relates generally to a system for compensating electromagnetic interfering fields, and in particular to a system for magnetic field compensation having two sensors and a digital processor. [+16010 tokens] In the following, the invention is described in more detail referring to the attached figures by means of exemplary embodiments, wherein same reference signs refer to same components.FIG1 schematically shows the system for compensating electromagnetic interfering fields. [+1427 tokens]
US-4830479-A	referring now to fig1 of the drawings , there is depicted a ray 12 entering the paper plane perpendicularly along an axis z orthogonal to axes x and y . ray 12 is deflected into the paper plane by a mirror 16 which is located at the origin and is oriented upwardly at a forty five degree angle from the paper plane . mirror 16 rotates with an angular velocity ω around axis z which is in line with the arriving ray 12 . [+1579 tokens]	The invention described herein may be manufactured and used by or for the Government for governmental purposes without the payment of any royalty thereon.At radio frequencies, superheterodyne receivers typically have sensitivities that are orders of magnitude higher than those of direct detection receivers. [+1044 tokens] Referring now to FIG1 of the drawings, there is depicted a ray 12 entering the paper plane perpendicularly along an axis Z orthogonal to axes X and Y. Ray 12 is deflected into the paper plane by a mirror 16 which is located at the origin and is oriented upwardly at a forty five degree angle from the paper plane. [+1380 tokens]

Table 1: Some examples from the two versions of the dataset. We report the first tokens from the input in the original version, and the first tokens in the new version of the dataset. Note that the new version might contain many paragraphs before the content of the original input.

		$BigPatent_{Original}$	$BigPatent_{New}$
# docs (train, val, test)		258,935	258,935
		14,385	14,385
		14,386	14,386
Summary	# tokens (avg)	123.9	121
		123.7	120.9
		124.1	121.2
	# sents (avg)	3.7	3.6
		3.6	3.6
		3.7	3.7
	sent len (avg)	44.3	43.4
		44.2	43.3
		44.5	43.7
Source	# tokens (avg)	3,959.2	5,488.3
		3,953.3	5,517.5
		3,976.8	5,501.9
	# sents (avg)	105.6	177.6
		105.5	178.4
		106.3	178.3
	sent length (avg)	42.6	31.8
		42.6	31.8
		42.5	31.8
compression ratio		36.1	51.2
		36.0	51.5
		35.8	50.9

Table 2: Length statistics on the two BigPatent versions. The number of tokens, sentences, tokens per sentence, and the compression ratio are computed per document and then averaged. The compression ratio is the ratio between the number of tokens in the source and the number of tokens in the Abstract.

	$BigPatent_{Original}$	$BigPatent_{New}$
Coverage (avg)	0.87	0.95
Density (avg)	2.40	20.8

Table 3: The extractive fragment coverage and the density for the two versions of the dataset. Measures are computed per document and then averaged.

fragments (shared sequences of tokens). The extractive fragment coverage measures the proportion of tokens in the summary belonging to an extractive fragment and qualitatively describes how much a summary vocabulary is derivative of a text.

$$Coverage(D, S) = \frac{1}{|S|} \sum_{f \in F(D,S)} |f|$$

where $|S|$ is the number of tokens in the summary. The density also takes into account the length of the extractive fragments: the higher the density, the more a summary can be described as a series of extractions.

$$Density(D, S) = \frac{1}{|S|} \sum_{f \in F(D,S)} |f|^2$$

Table 3 shows the measures computed for the two versions of the dataset, while Table 4 shows their percentage of novel n-grams. Note that both datasets have relatively high coverage (the increase in $BigPatent_{New}$ might be partially motivated by the increased length of the source). However, the extractive density is an order of magnitude higher in $BigPatent_{New}$, suggesting that the reference summaries are significantly more extractive than the original version.

$BigPatent_{New}$ also has a lower number of novel n-grams in the summary (and the difference with $BigPatent_{Original}$ stays high even when accounting for the length of the source). We attribute this difference to the presence of sections such as the Summary of the invention, the Background, and the Field of the invention in the input; these sections already abstract the core features of the claimed invention.

To investigate how similar the Abstract is to each subsection in $BigPatent_{New}$, we compute their ROUGE scores (Lin, 2004) with the summary[9]. We report both ROUGE f1 and recall since we want to quantify how much "information in the Abstract" each section contains. $BigPatent_{New}$ does not include the name of the patent subsections (an uppercase short header in the raw text). In fact, short sentences (including subsection names) are removed during the preprocessing. To divide the text into subsections, we regenerate the dataset using the original TensorFlow script and remove the portion of the code that gets rid of short sentences. We use a regular expression to divide the text into subsections and extract their headers. Since the headers do not have normalized names (e.g., the Background's header might be indicated as "Background", "Background of the invention", etc.), we use a simple key-based method to classify them into 9 groups. Note that not all patents include all subsection types. Table 5 reports the obtained ROUGE score, the subsection average length, and the percentage of patents that include each subsection type. Note that the Summary of the invention (in 94% of the inputs in $BigPatent_{New}$) has the highest scores; compared to the Detailed Description, the Summary of the Invention has a higher ROUGE-recall even though it is much shorter.

In a nutshell, our analysis shows that the additional text in $BigPatent_{New}$ decreases the need for an abstractive model for the task. The additional Description subsections – in some cases, already a summary of the rest of the patent – contain the most information in the patent Abstract.

	$BigPatent_{Original}$	$BigPatent_{New}$
Novel 1-grams (avg)	10.9%	4.21%
Novel 2-grams (avg)	46.9%	23.46%
Novel 3-grams (avg)	74.0%	42.25%
Novel 4-grams (avg)	87.1%	53.58%

Table 4: Percentage of new n-grams in the summary in the two datasets. All percentages are computed per document and then averaged.

3 How to compare to the previous literature?

While the two versions of the dataset have different characteristics, the vast majority of previous literature using BigPatent does not explicitly mention the version used.

Zhang et al. (2020) mention they "updated the BIGPATENT dataset to preserve casing, some format cleanings are also changed"; this operation might have led to the creation of the new dataset version now exposed by TensorFlow (whose differences with the original version are, however, not limited to casing and minor format cleaning). Some previous work (He et al., 2020) noticed a substantial performance gap between models trained with the original version and Pegasus and speculated this difference might be due to the different preprocessing (and, we add, possibly to the additional content); these findings are compatible with our experiments in the next section.

In the vast majority of cases, the reported statistics are directly taken from the original publication and not recomputed; in a few cases, the values computed (e.g., in terms of document lengths) are compatible with the use of the cased version (e.g., in Guo et al. (2022)).

BigPatent is widely used when testing systems, often as an example of a dataset with a very long source. The dataset was cited 115 times, according to Google Scholar[10]. Since the used dataset version is unknown, and authors are unaware of the two different versions, it is impossible to understand if comparing results to previous work is fair. Since the Tensorflow version was updated on the 31st Jan 2020[11], papers published after that date could potentially use the new version of the dataset, with likely better results. In fact, a simple BART model results in a very different performance on the two versions of the dataset, as shown in the next section.

[9] All ROUGE scores are computed using the Hugging Face version of the metric, with stemming.

[10] Checked on 27/10/2022

[11] See this github commit: a708d506748870237eafa2bbb659dc64cd7cf04a

	ROUGE-1		ROUGE-2		ROUGE-L		#Tokens	%
	R	f1	R	f1	R	f1		patents
SUMMARY	84.68	35.97	60.76	25.97	69.07	29.36	744.56	93.79%
FIELD	23.62	28.66	10.17	11.92	16.14	19.44	73.73	38.27%
BACKGROUND	66.04	24.45	25.38	8.60	41.42	14.70	710.04	94.85%
DRAWINGS	38.96	28.36	10.35	7.39	24.52	17.55	243.43	97.6%
EMBODIMENTS	81.39	8.58	42.44	4.14	59.21	5.92	3168.25	53.07%
REFERENCES	10.82	11.40	1.48	1.35	07.38	7.94	92.10	28.18%
RELATED ART	52.47	20.33	18.48	6.36	32.13	12.04	644.27	4.12%
OBJECTIVE	44.35	32.31	16.05	10.93	27.49	19.58	256.95	2.09%
DESCRIPTION	84.39	8.27	4.10	4.08	61.90	5.78	3404.91	55.23%

Table 5: The ROUGE score (recall (R), f1) between the different subsections of the patents and the patent Abstract. The subsections are obtained from the $BigPatent_{New}$ raw data. The scores are computed per document and normalized by the number of documents that contain each subsection. The average length of each subsection and the percentage of patents that contain the subsection are also reported.

4 Experiments

To understand if the version of the dataset impacts models' performance, we fine-tuned a pre-trained BART (Lewis et al., 2020) base model on the two versions of the dataset. We train using the Hugging Face library with early stopping on the evaluation loss (patience: 5) and the following hyperparameters: max source length: 1000; max target length: 150; number of beams: 5; eval steps: 10k; max steps: 500M. We leave all other parameters to their default values. Table 6 reports the results. Note how results on $BigPatent_{New}$ are more than 11 points of ROUGE-L over $BigPatent_{Original}$.

To corroborate the idea that the Summary of the invention in the input improves the performance on $BigPatent_{New}$, we trained a model using, as input, only the text in the Summary of the Invention subsection. In the few cases in which the patent did not include the Summary subsection, we used the Detailed Description or the Description of the embodiments. As described in Section 2, we resorted to the raw data to extract the text in the Summary of the Invention subsection. This setting further improves the performance, with an increase of almost 16 and almost 5 points of ROUGE-L with respect to the original and the new version; note, however, that since $BigPatent_{New}$ does not contain the subsection headers, it is not directly possible to train models using the Summary of the Invention only as input.

5 Conclusions

We have discussed the case of BigPatent, a dataset that exists in two very different versions. We have shown that the updated version of the dataset lacks some of the original characteristics (e.g., the high level of abstraction in the reference summaries and their high percentage of novel n-grams) and leads to much higher results with a simple transformer.

To our best knowledge, this difference is not reported elsewhere, either in published research or in the dataset's online documentation. In fact, previous work tends to ignore the difference between the original and the new version, making it virtually impossible to understand experimental results, reproduce, and compare them.

We believe BigPatent is an extreme case in which the lack of clear documentation has led to confusion – with two datasets so distant in their characteristics that they might be considered two different ones, used interchangeably. We always advise reporting the dataset version and characteristics when using BigPatent (and being aware of the possible problems with the comparison with previous work).

We hope that the analysis of this case underlines the importance of clearly documenting datasets' characteristics and any possible modifications introduced over time.

Limitations and ethical impact statement

The dataset we analyzed is public and derives from public patent data. We are not aware of any ethical concerns related to the dataset.

In this paper, we have only analyzed a subset of the dataset, but our considerations are general. We have done so for computational concerns, including trying to limit the requirement for energy resources.

	$BigPatent_{Original}$			$BigPatent_{New}$		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
Lead-3	29.54	7.95	18.15	23.15	7.27	15.42
Summary Lead-3	-	-	-	48.11	30.16	36.66
BART-base	42.25	15.99	27.58	50.18	29.46	38.64
BART-base (Summary)	-	-	-	55.16	34.85	43.56

Table 6: Results (test set) on the two dataset versions for a BART-base model. The Lead-3 baseline considers the first three sentences of the input text as a proxy for the generated summary. Summary Lead-3 uses the first 3 sentences of the Summary of the invention (obtained from the Summary of the invention as described in Section 2.1). We also trained a BART model that only uses the Summary of the Invention as input. The split is identical, i.e., the train, validation, and test splits contain the same documents in both versions.

References

Silvia Casola and Alberto Lavelli. 2022. Summarization, simplification, and generation: The case of patents. *Expert Systems with Applications*, 205:117627.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2020. Ctrlsum: Towards generic controllable text summarization. *ArXiv*, abs/2012.04281.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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 Šaško, Gunjan Chhablani, Bhavitvya 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 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*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the workshop on Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.