

Project PIAF: Building a Native French Question-Answering Dataset

Rachel Keraron^{≡,★}, Guillaume Lancrenon^{≡,‡}, Mathilde Bras^{≡,‡},
Frédéric Allary[★], Gilles Moyses[★], Thomas Scialom^{★◇},
Edmundo-Pavel Soriano-Morales[‡], Jacopo Staiano[★]

≡ equal contribution

★ reciTAL, Paris (France)

‡Etalab, DINUM, Prime Minister’s Office, Paris (France)

◇ Sorbonne Université, CNRS, LIP6, F-75005 Paris, France

{rachel, frederic, gilles, thomas, jacopo}@recital.ai

{guillaume.lancrenon, mathilde.bras, pavel.soriano}@data.gouv.fr

Abstract

Motivated by the lack of data for non-English languages, in particular for the evaluation of downstream tasks such as Question Answering, we present a participatory effort to collect a native French Question Answering Dataset. Furthermore, we describe and publicly release the annotation tool developed for our collection effort, along with the data obtained and preliminary baselines.

Keywords: Question Answering, Annotation, Crowdsourcing

1. Introduction

Along with the availability of massive amounts of data, the increase in computational power has in recent years allowed the development of Deep Learning techniques, leading to significant advancements in the fields of Computer Vision (CV), and Natural Language Processing (NLP), among others. Visual information can, to some extent, be considered to generalize across cultures in many real world applications; in contrast, having to deal with languages, NLP applications are naturally bound to language specificities.

Over the years, the NLP community has produced several resources to tackle tasks we call, for simplicity, *upstream* (such as Part-Of-Speech tagging, Dependency Parsing, etc.), targeting multiple languages and enabling the construction of effective automated systems. Still, for tasks we refer to as *downstream*, i.e. those which enable the development of value-added end products such as Question Answering (QA) or Conversational Agents, the current state-of-the-art approaches require massive amounts of annotated data which are almost exclusively available in English. Notable exceptions include Machine Translation, for which abundant parallel corpora have been built from resources such as the European parliamentary proceedings, or Language Modeling – which can be tackled in a self-supervised manner, hence only requiring massive amounts of text in the target language(s). Such asymmetry has recently been acknowledged by a resolution of the EU Parliament.¹

In this work, we focus on collecting Question Answering data in French in a participatory setup: we describe the data collection protocol adopted, report descriptive statistics and

baselines, and provide details on the implementation of the open source annotation tool we developed. Such tool allows volunteers to participate in crowdsourced QA dataset collection campaigns.

In summary, we make the following contributions:

1. we develop and release a novel annotation tool to collect large-scale QA data in a participatory scenario;
2. we release a native French QA dataset;
3. we provide baselines using state-of-the-art methodologies.

2. Related Work

Several datasets for QA have been recently produced. The Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) consists of 100,000+ questions posed by crowdworkers on a set of Wikipedia articles. The answer to each question is a segment of text from the corresponding article. Its latest version includes circa 50k additional unanswerable questions (Rajpurkar et al., 2018).

Targeting open-domain QA, WikiQA (Yang et al., 2015) consists of Bing queries and corresponding answer sentences taken from Wikipedia articles. CoQA (Reddy et al., 2018) focuses on conversational aspects, compounding to 127,000+ questions with answers collected from 8000+ conversations. Each conversation is collected by pairing two crowdworkers to chat about a passage in the form of questions and answers. Based on CNN news articles, NewsQA (Trischler et al., 2017) consists of 120k Q&A pairs, including unanswerable questions, collected using a 3-stage, siloed process; crowdworkers are split into 3 groups: Questioners, who see only an article’s headline and highlights to produce a question; Answerers, who see the question and the full article, then select an answer passage; and Validators, who see the article, the question, and a set of answers that they rank.

¹European Parliament resolution of 11 September 2018 on language equality in the digital age. <http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//TEXT+TA+P8-TA-2018-0332+0+DOC+XML+V0//EN>

Recognizing that several pieces of information often jointly imply another fact, the two datasets provided by QAngaroo (Welbl et al., 2018) target Multi-Hop (or Multi-Step) QA, i.e. the goal is to answer text understanding queries by combining multiple facts that are spread across different documents. The provided datasets, WikiHop and MedHop target, respectively, open-domain and domain-specific QA, with the latter focusing on interactions between pairs of drugs in the medical domain. Also targeting Multi-Hop reasoning, HotpotQA (Yang et al., 2018) includes 113k Wikipedia-based question-answer pairs where the questions require finding and reasoning over multiple supporting documents to answer.

TriviaQA (Joshi et al., 2017) includes over 650k question-answer-evidence triples, with questions originating from trivia enthusiasts independent of the evidence documents, automatically gathered and hence noisy. In addition, it provides a clean, human-annotated subset of 1975 question-document-answer triples whose documents are certified to contain all facts required to answer the questions.

The Yahoo! Answers datasets, available for English and French under restrictive non-commercial licenses², include question-answer pairs obtained through the Yahoo! Answers service. No supporting passage for an answer is provided, although links to relevant web pages might be included in the more elaborate answers. Available also under a non-commercial license, InsuranceQA (Feng et al., 2015) is a domain-specific QA dataset targeting the insurance domain.

Do We Need Native Non-English Data?

In short, *yes*.

From the above mentioned data collection efforts, we see that with the exception of Yahoo! Answers, which does not provide supporting evidence for the answers, no large-scale QA dataset is available for languages other than English. Current approaches based on transfer learning and multilingual model pretraining allow to build models able to deal with non-English. In other words, one can effectively fine-tune a pretrained multilingual language model (e.g. (Devlin et al., 2019)) on English QA data. Nonetheless, as observed by (Lewis et al., 2019), native *evaluation* data for the targeted language is a *must-have* in order to measure progress on the task for a given language. (Lewis et al., 2019) thus released to the community an aligned multilingual evaluation corpus containing QA data in 7 languages (English, Arabic, German, Spanish, Hindi, Vietnamese and Simplified Chinese).

Recent research (Liu et al., 2019) on Chinese language (arguably, the other high-resource language³ along with English) indicates that, while translation-based method and multilingual approaches can obtain reasonable performances, there exists a large margin for improvements. To allow the research community to tackle those issues, it is thus desirable to obtain comparable data for the task, natively for the language of choice. Efforts in this direction

have focused on Chinese (Cui et al., 2018), Korean (Lee et al., 2018; Lim et al., 2018).

Hence, project PIAF⁴ (*For a French-language AI* or *Pour une IA Francophone* in French) focuses on filling this gap for French, starting from the Question-Answering use case. In a first stage of the collection effort, we prioritize quality over quantity: we conducted several in-place *annotathons* wherein volunteers are accompanied by the PIAF team. As showing up to the meetings was usually motivated by curiosity towards AI topics, the presence of the PIAF team served to increase engagement and allowed to provide participants with basic knowledge on how AI models can be built. We provide more details on *annotathons* in Section 5. Concurrently with our public effort, another team was collecting native French QA data, unbeknownst to us: we provide a comparison with FQUAD (d’Hoffschmidt et al., 2020) in Section 6.

3. Protocol

Consistently with the recent works targeting QA data collections in non-English languages, we focus on *extractive* Question Answering and adapt the protocol used for SQuAD: we collect samples in the form of triplets $\{P, Q, A\}$, in which the answer A to a question Q is contained in the paragraph P .⁵

The great majority ($\sim 95\%$) of the Wikipedia articles used to build the SQuAD dataset have a version on the French Wikipedia, making an article-by-article replication of the collection process feasible. Still, such approach suffers from several downsides: for instance, cultural differences can cause significant divergence between the English and French versions of the same Wikipedia article, both in terms of data availability and reliability. Taking the Wikipedia article on the 2016 edition of the Super Bowl (present in the SQuAD dataset) as an example, one can easily notice how the contents significantly differ between its English and French versions.

Thus, we select relevant articles the French version of Wikipedia, using the same relevance metric as used in SQuAD (i.e. PageRank); we segment those articles in smaller paragraphs, and collect sets of question/answer pairs corresponding to those paragraphs.

Two collection settings are envisioned: the first, *certified*, restricted to volunteers participating in live *annotathons* (see Section 5); the second, *open*, wherein we will keep instances of the annotation platform open to the public web and engage the community to contribute to the collection, using a similar approach as that of Mozilla Common Voice.⁶

The collected data is publicly released under CC-BY-SA license.

Source Articles Selection

For SQuAD, the authors used the highest-ranking, in terms of PageRank, 10k articles from the English Wikipedia. Nonetheless, when applying the same threshold on the

²<https://webscope.sandbox.yahoo.com/catalog.php?datatype=1>

³In terms of publicly available text and corpora.

⁴<https://piaf.etalab.studio/>

⁵The presence of *unanswerable* question marks the difference between SQuADv1.1. and SQuADv2.0.

⁶<https://voice.mozilla.org>

French version, we noticed significant differences in terms of structural properties between French and English Wikipedia. For instance, we observed the massive presence of pages referring to *years* on the French version, a characteristic that the top-10k subsample of the English Wikipedia does not exhibit. After manual inspection, the editing practices seem to differ between the French and English communities: for instance, while the latter do not link all *year* mentions to the actual page dedicated to that year, the French Wikipedia editors seem to systematically do so — a fact that boosts the PageRank score of such articles and therefore explains their abundant presence in the French top-10k subsample.

To account for such practices, an extensive manual inspection step on random samples from the French Wikipedia allowed us to identify the following filtering criteria.

At section level, we discard those with the following titles: *Voir aussi (See also)*, *Articles connexes (Related Articles)*, *Liens externes (External links)*, *Notes et références (References)*. Those sections most often contain only bullet lists with structured rather than textual information.

At article level, we discard those containing a section titled *Événements (Events)*: those articles are about years, as in the aforementioned example, their structure is not apt to extractive QA since they mostly consist of lists of events (a date, followed by a very short text). Further, for concerns on data quality we discard articles falling in two Wikipedia categories: *Catégorie:Wikipédia:ébauche (Draft)* and *Catégorie:Homonymie (Disambiguation)*.

To summarize, we operate as follows:

- gather the top-25k articles in terms of PageRank;⁷
- discard the articles matching the above criteria;
- set a min-max char limit on the paragraph length (*min* = 500; *max* = 1000);
- filter out articles with less than 5 paragraphs.

Compared to the English SQuAD, we hence obtain annotated QA data on more articles, with less paragraphs per article, and a comparable length.

Category Distribution

We associated each article to the main category it belongs to. To do so, we first analysed the category/project trees associated to each Wikipedia article. We deemed such information as not directly exploitable, given the presence of several ambiguities and inconsistencies. Nonetheless, such analysis allowed us to shortlist a subset of categories which were most represented. We then proceeded to a manual annotation step to associate each article to the most relevant category. The category information would be instrumental as a factor for engagement, in our participatory scenario: the volunteers have control over the general category they want to contribute to. In Table 1 we show the distribution of articles per category.

⁷We adapted the code from Project Nayuki <https://www.nayuki.io/page/computing-wikipedias-internal-pageranks>

Continuous Evaluation

It is of utmost importance to drive the users to produce challenging questions. In the SQuAD collection interface, the user was reminded to avoid using the same words/phrases as the paragraph while writing a question, via a text message displayed on screen. In our collection interface, we provide annotators with examples of good and bad questions.

In order to check the quality of the collected data, we execute both manual and automatic evaluations throughout the data collection process, on a rolling basis (*i.e.* as the collected dataset grows), as in the original SQuAD paper.

For manual evaluation, we analyzed 191 questions from the collected *certified* data (*i.e.* one randomly sampled triplet per article), and assign scores according to the dimensions defined in the SQuAD paper – see Table 3. Conversely, we compute automatically, on the ensemble of the data and on a rolling basis, scores for *syntactic divergence* and *lexical variation*. Manual and automatic evaluation results for *PIAFv1.0* are discussed in Section 6.

Further, we incrementally measure the performances obtained by state-of-the-art multilingual QA systems, under several experimental setups.

4. Annotation Platform

Effective annotation frameworks are essential for building language- and/or domain-specific NLP corpora. While multiple QA datasets have recently been produced (see Section 2), there is still a lack of annotation frameworks which are open and accessible to the community. The main reasons lie in the usage of proprietary software, the deep link with crowd-sourcing marketplaces (e.g. Mechanical Turk), and to the lack of minimal features enabling QA data collection. We thus present PIAFAnno, a browser-based, mobile-friendly, QA annotation tool. PIAFAnno was created to meet the following constraints:

1. *Web-based crowd-sourcing platform*: allowing for a distributed, large-scale contribution.
2. *User and contribution management*: supporting different roles among the annotators, as well as keeping control of the progress and the quality of the contributions.
3. *Modern interface*: our collection protocol is centered on voluntary participation. Therefore, the workflow should be pleasing and engaging for the annotators.
4. *SQuAD compatible*: to make the data quickly actionable by the community, the input and output formats follow the SQuAD format. The SQuAD annotation flow is respected (*i.e.* creating a number of questions for each paragraph of a single document).
5. *Permissive and open-source license*: making our tool reusable by the community as a part of our commitment to open source and data.

	Arts	Geography	History	Religion	Sciences	Society/Misc.	Sport	Total
Certified	52	52	57	16	52	36	26	291
<i>PIAFv1.0</i>	30	32	38	12	36	25	18	191
Open	228	155	233	85	190	167	97	1155
Total	280	207	290	101	242	203	123	1446

Table 1: Category distribution of source articles in *Certified* (a subset of which is released as *PIAFv1.0*) and *Open* splits.

Features	Daemo	cdQA	QA-Turk	PIAFAnno
Multiple users	✓			✓
Modern interface				✓
Open source	✓	✓	✓	✓
SQuAD compatible	✓	✓	✓	✓
Actively developed		✓	✓	✓

Table 2: QA annotation tools and their supported features.

4.1. Existing Platforms

Before launching our development effort, we reviewed existing open-source platforms that could fit for our scenario. Unfortunately, while there are several sequence-annotation tools, such as *brat* (Stenetorp et al., 2012), *WebAnno* (Yimam et al., 2013), or *Doccano*⁸ (Nakayama et al., 2018), we identified, to the best of our efforts, only three candidates as open-source, web-based, QA annotation platforms (shown in Table 2). We aim to fill this gap with PIAFAnno. The crowd-sourcing platform used in the original SQuAD paper (Rajpurkar et al., 2016), *Daemo* (Gaikwad et al., 2015), is publicly available and open-source.⁹ Nonetheless, its development seems to have stalled and the project appears not actively maintained.

Turning to the specific QA annotation platforms, we surveyed two tools: *cdQA-annotator* (Mikaelian et al., 2019) and *QA-Turk* (Fisch, 2018). The former is a part of a larger web-based QA suite. The annotation workflow in *cdQA-annotator* allows for direct selection of the answer span, which facilitates the task. Furthermore, it uses the SQuADv1.1 file format for input and output files. Nonetheless, it includes neither contribution nor user management capabilities. Conversely, *QA-Turk*, based on *ieturk* (Quach, 2018), is an add-on allowing to create question-answers pairs. By default, it uses Amazon Mechanical Turk as a crowd-sourcing back-end, but can be used with a local-based alternative. While also tailored for crowd-sourcing, it does not support different roles for the contributors. We note that, while these two platforms do not satisfy our requirements, they include valuable features such as Mechanical Turk integration for *QA-Turk* or the QA tools included in the *cdQA* suite (model training, visualization, exploration). Finally, those tools do not seem to allow collection of additional answers for a subset of the data (as mentioned above, this allows to both make the evaluation data more robust, and to compute human performance).

⁸We actually began working on implementing a QA module for Doccano but due to a lack of time for pull-requests reviews and other technical details we finally decided to roll out our own solution from scratch.

⁹<https://github.com/crowdresearch/daemo>

PIAFAnno aims to overcome these limitations, specifically in two areas: dealing with multiple, role-diverse contributors and making it as easy as possible for them to create annotated samples. As said before, due to the constraints of our protocol, having two types of users (certified and open, see Section 3) and keeping volunteering contributors engaged is essential for the success of our approach. These two features can be easily be extended to other annotation efforts, making PIAFAnno a valuable resource for the creation of custom QA datasets. Source code and installation/deployment instructions for PIAFAnno can be found at <https://github.com/etalab/piaf>. In the following section, we provide a general description of the inner workings of our annotation platform.

4.2. PIAFAnno System Architecture

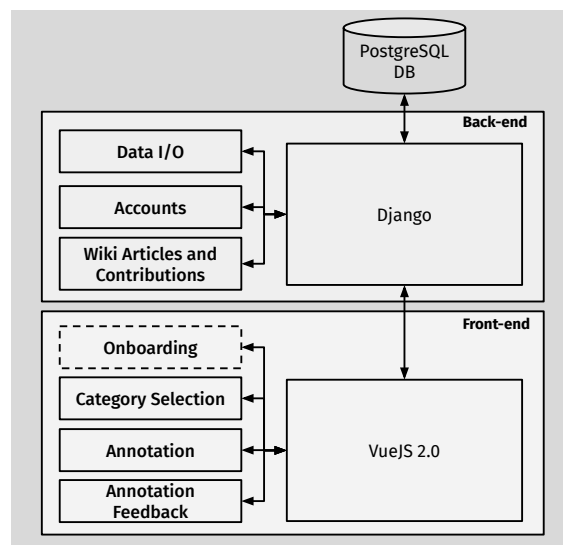


Figure 1: PIAFAnno system architecture.

The architecture of PIAFAnno is presented in Figure 1. The system consists of seven modules attached to either the back-end or the front-end. All the functionalities described in this section are accessible through a web interface. Back-

end administration is handled via the Python Django framework.¹⁰ The front-end annotation modules were developed using the Vuetify¹¹ framework on top of VueJS 2.0¹².

PIAFAAnno is built in a modular fashion, with a focus on simplicity and ease-of-use, and allows for ad-hoc modifications and extensions. Furthermore, the platform is dockerized via `docker-compose` allowing for quick and straight-forward installation.

Below we describe the characteristics of each of the seven modules. We cover first the back-end and then move to the front-end modules.

Data I/O The data I/O interface allows for import/export of documents (in our case, French Wikipedia articles and their paragraphs) to be used during the QA collection. Both tasks require and produce JSON files that follow the SquADv1.1 file format.

Accounts This interface allows the platform administrators to manage the contributors' accounts. These include creation date, user type (either regular, admin or super-admin¹³), their role (certified or open). Admin users can modify the certified status of the contributors of the platform. Via this module, administrators can access the scores obtained by the contributors, and are thus able to fine-tune the process for future onboardings or to change a contributor status accordingly.

Once an account is created for a contributor, they receive a validation link via email, allowing them to login into the platform. Standard facilities such as password reset are also managed.

Wikipedia Articles and Contributions This module allows to explore the source texts¹⁴ as well as to monitor the contributions. Details on the created samples (timestamp, author, content, etc.) are available for inspection. This monitoring interface allows for real-time monitoring of the data collection effort.

Similarly, it allows fine-granularity access to the source data, providing article-level details such as category (or topic), and the contributor role required for annotation. At paragraph-level, besides the inherited information, contents and annotation status (e.g. whether question/answer pairs have already been generated on it) are made available.

Below, we move to describe the interfaces the contributors directly interact with, during the annotation.

Onboarding The onboarding module aims to facilitate the comprehension of the platform. During preliminary user tests, we found that contributors need a set of guidelines to begin the annotation process according to our quality requirements (e.g., question complexity, answer span length, and so on). It also include a short assessment to evaluate the level of understanding of the annotation guidelines for the onboarding user. The procedure thus consists

of two steps: i) the explanation of the guidelines, and ii) the user assessment via simple evaluation questions. This allows us to limit the number of bots, trolls, and contributors who misunderstand the guidelines of the annotation process. This module is still under active development.

Category Selection As described in Section 3, we manually categorized the source Wikipedia articles into seven topics. This interface allows users to select the category they would like to contribute samples on.

Annotation Users are required to come up with five question-answer pairs per paragraph. The paragraph is shown according to the original order in their respective Wikipedia article (see Figure 2). The article shown, and thus its paragraphs, belong to the category previously chosen by the contributor.

The annotation procedure consists of three steps:

1. *Paragraph reading*: the main area of the page is dedicated to the paragraph from where contributors will create their annotations. The text formatting (e.g. font, spacing, line height, color contrast) has been reviewed by a design specialist. A progress bar allows contributors to know where they are in the current annotation task.
2. *Question writing*: the question input field is placed below the reading area. The input is limited to 200 characters.
3. *Answer selection*: within the paragraph reading area, the user can select the answer to question they previously wrote. We found that web browsers' traditional text span selection lacks speediness, usability, and regularly shows buggy mobile interaction. Specifically, during our preliminary tests, we found several cases of highlighting of incomplete words, and thus incomplete answers, which is a situation we needed to avoid. To address this problem we developed a custom highlighting component. In this component, the highlighting is done at word-level, rather than at character-level. This ensures complete answers and increases the annotation efficiency. The answer selection works as follows: with the first click, the user selects the initial word of the answer, while with the second click they will select the last word of the answer span. Finally, the complete answer text span is automatically inferred.

The question-answer pairs can always be edited by the contributor. Once five question-answer pairs are generated for a given paragraph, they are sent to the back-end and stored in the database. The user is then redirected to the annotation feedback page described below.

Annotation Feedback The annotation feedback page is shown after a paragraph is completed by the annotator. It congratulates the user and shows them general statistics as a contributor, such as the running number of data samples contributed. This is part of our effort increase and sustain engagement for our contributors.

¹⁰<https://www.djangoproject.com/>

¹¹<https://vuetifyjs.com/>

¹²<https://vuejs.org/>

¹³The difference between an admin and a super-admin is that the admin is restricted on most of the deletion, modification, and insertion database operations.

¹⁴In this study, these are Wikipedia articles. Other types of text sources may be used as input.

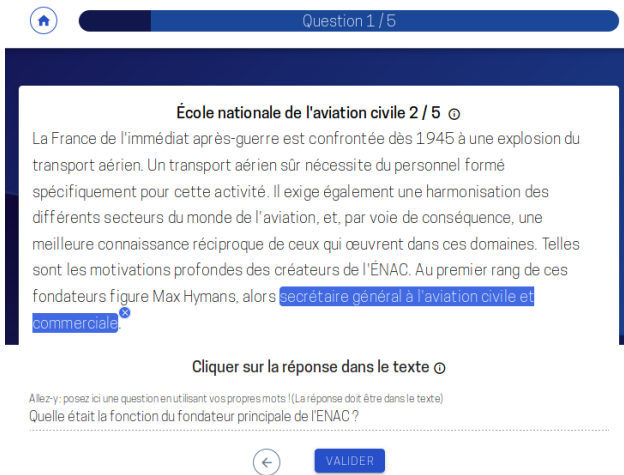


Figure 2: PIAFAnno’s annotation page.

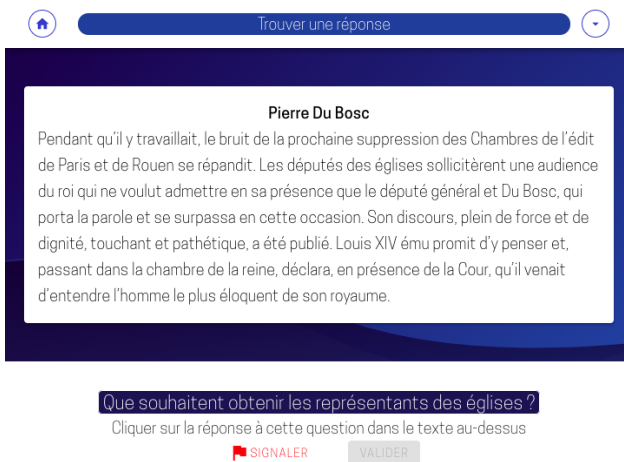


Figure 3: PIAFAnno’s additional answer page.

Additional answers The certified users have access to an additional annotation mode. It is similar to the original annotation procedure, but in this case the question is already written and the task is to select the corresponding answer. This procedure is composed of two steps (as in Annotation mode):

1. *Paragraph and Question reading,*
2. *Answer selection.*

The question comes from the pool of question-answer pairs already created by other contributors. We consider a sample as complete when two additional answers are selected. In the end, every question will have three answers from three different contributors. Additionally, the user can if needed flag questions (e.g. as unanswerable, ambiguous or offensive questions) using a dedicated button. A screenshot of this mode can be seen in Figure 3. Additional answers are, at the time of this writing, under collection, and will be included in the upcoming 1.1 release of the dataset.

5. Participatory Approach

We base our approach on the protocol described previously (see Section 3). We introduce necessary modifications due

to the constraints we set to work with: notably, not using Mechanical Turk or similar platforms, thus exploring a method to source voluntary contributions while using a dedicated annotation platform.

The reasons for not using Mechanical Turk (or similar platforms) derive from the participatory approach we adopted. Given its experimental nature, we want to have full control on the entire process and to keep it as fair as possible to the volunteering contributors. Full ownership of the process and the associated software tools allows us to quickly and easily modify the annotation effort methodology “on-the-fly”: the required changes can be implemented and deployed as transparently as possible for the contributors, safeguarding a pleasant annotation experience.

The creation of quality question-answer pairs is a complex task that, as noticed in our first annotation events, requires a considerable amount of concentration and engagement. Hence, the annotation procedure needs to keep the users motivated and engaged to discover and continue to use the platform. At the same time, we need to maximize the number of contributions by rallying volunteer contributors. As discussed in the previous sections, the developed software includes features aiming at facilitating the annotation task and at engaging participation. On the other hand, to entice the participation of contributors, as a first approach we explore a voluntary crowdsourced acquisition method, inspired by the Mozilla CommonVoice project.

Specifically, we organize weekly annotation events, or *annotathons*, where contributors are invited to come and participate in the creation of the French QA dataset. These sessions are guided by the PIAF team in order to prepare the annotators to use the annotation tool and to generate high-quality annotations. Since participants receive in-house training into the flow and quality expectations of the annotation process, we consider them as certified annotators (as explained before). The first contributors for these sessions were mainly drawn from the public sector.¹⁵

The layers of contributions we are currently testing are the following:

1. *Neighboring:* we promote the weekly annotation event across our neighboring ecosystem network,
2. *Traveling:* we go and present our annotation project to local natural language processing, artificial intelligence or open data related events,
3. *Peripheral:* we build links with and among the external scientific and/or activist communities to get feedback and new avenues of collaboration in order to increase the project’s relevance,
4. *International:* as this project is a French-language effort, and not only French (from France), we seek contributions with other governments, research laboratories, institutions, and other communities beyond the French borders.

¹⁵We leverage Etalab’s network and expertise on the organization of annotathons. Etalab is the French government Open Data task force.

Our voluntary contribution approach is currently being tested. Other means of contribution may be explored. We also note that during all the annotation events we carry out, we put emphasis on enhancing artificial intelligence literacy through short, educational presentations. Furthermore, since these crowdsourcing events are innovative within the French public-sector context, we are working with sociology specialists in order to observe and provide feedback on the dynamics and development of the project.

6. Dataset Analysis

PIAFv1.0 compounds to 3835 question/answer pairs collected on 761 paragraphs under a *certified* setup: volunteers gathered for *annotathons* animated by the PIAF team. 285 volunteers contributed annotated samples to *PIAFv1.0*.

Consistently with SQuAD, we report in Table 3 the human assessment scores for 191 triplets (one per article, randomly sampled). Conversely, Figure 4 shows the distributions for lexical variation and syntactic divergence among all triplets.

The PIAF project went public on October 3rd, 2019. As mentioned in Section 2, a similar effort (d’Hoffschmidt et al., 2020) was concurrently being conducted, unbeknownst to us, producing a quantitatively larger French QA dataset named FQUAD. Therefore, we include a comparative analysis of *PIAFv1.0* and FQUAD.

For some analyses below, we use the automatic French translation of the English SQuAD (v1.1) obtained via Google Translate APIs,¹⁶ referred to as *SQuAD-Fr*. It compounds to 74308 and 9455 samples for training and development sets, respectively, after filtering out bad samples (e.g. those for which the translated answer could not be recovered from the input).

In (d’Hoffschmidt et al., 2020), the authors rely on CamemBERT (Martin et al., 2019) for their evaluations, but do not report the hyper-parameters used. For all our experiments, we use *batch_size* = 8, *learning_rate* = $3e^{-5}$, *n_epochs* = 2, *max_seq_len* = 384, *doc_stride* = 128.

In Table 4, we report the results obtained by fine-tuning CamemBERT¹⁷ on different combinations of the datasets available. First, when evaluated on *PIAFv1.0* data, the performances of all models are significantly lower than on the FQUAD development set. Second, a model trained on the automatic French translation of SQuAD obtains, on the FQUAD development set, the same performance as using the FQUAD training set.

Turning to using *PIAFv1.0* samples for training, rather than for evaluation, we observe that a model trained on SQuAD-Fr, and augmented with the *PIAFv1.0* samples, obtains comparable results (about half a F1 point improvement) w.r.t. using the FQUAD training set. Note that our CamemBERT finetuned on FQUAD training samples, and evaluated on FQUAD development data, obtains a performance significantly lower (10 absolute F1 points) than that reported by (d’Hoffschmidt et al., 2020) on the FQUAD hid-

den test set.¹⁸ This can be explained by a number of factors, including: hyperparameters’ setup (not reported in the FQUAD paper); the use of additional answers for computing evaluation scores on the hidden test set (although, for reference, this factor only justifies 3-4 points of difference on the SQuAD dev set).

As shown by (Geva et al., 2019), employing a small number of annotators (for FQUAD, $N = 18$) can result in annotator bias: if unaccounted for, e.g. by creating training/evaluation splits based also on annotator identifiers, models might fail to generalize to samples produced by annotators that did not contribute to the training set. In *PIAFv1.0* this risk is mitigated by the large number of volunteering contributors ($N = 258$). Finally, having a larger pool of annotators, we also ensure higher diversity in the textual samples collected: this seems to have a direct impact for training (Martin et al., 2019) and for evaluation robustness. The results shown in Table 4 indicate that our participatory approach allows to obtain relatively more challenging evaluation samples.

Furthermore, taking as reference a model trained on SQuAD-Fr only, we observe that the addition of samples from *PIAFv1.0* during training obtains a slightly larger improvement on FQUAD (dev) than adding a comparable subsample (FQUAD_{sub}) from the FQUAD training set. Since in the former case there is no risk of annotator bias (no overlap between FQUAD and PIAF annotators), this shows that *PIAFv1.0* samples can also effectively be used for training data augmentation.

7. Conclusion

Motivated by the scarcity of non-English data, we described our ongoing effort towards gathering native QA samples for the French language, using a participatory approach. Rather than a transactional approach to data collection, as usually adopted in crowd-sourcing efforts, we experiment with a comparatively slower and more engaging process, focusing on quality over quantity. Amongst desirable side-effects of our approach, we highlight the educational aspects, e.g. introducing a wider audience to AI concepts and methodologies during our *annotathons*.

The analyses reported indicate that the samples collected in *PIAFv1.0* can effectively be exploited for robust evaluation or training.

Furthermore, we presented a novel, open-sourced, and language-agnostic data collection platform for Question Answering tasks, developed within the context of Project PIAF.

Both the PIAFAnno platform and the data collected are released under permissive and open-source licenses, with the goal of engaging a diverse and wide community of practitioners. We leave the web platform open to the public, in order to collect additional samples from online contributors, which will be included in future releases of the dataset.

¹⁶translate.google.com

¹⁷We use the implementation provided at <https://github.com/huggingface/transformers>.

¹⁸The same CamemBERT model, evaluated on the SQuAD-Fr dev set, obtains better results (F1: 73.28, EM: 59.18) than those reported by (d’Hoffschmidt et al., 2020) (F1: 70.7, EM: 56.9).

Reasoning	Description	Example	Percentage
Synonymy	Major correspondences between the question and the answer sentence are synonyms.	Q: Combien de Polonais vivent sur Terre? Sentence: Avec une population de <u>38 millions d'habitants</u> , la Pologne [...]	41.36 %
World Knowledge	Major correspondences between the question and the answer sentence require world knowledge to resolve.	Q: Quelle maladie a terrassé le prédécesseur du général de Rochambeau? Sen.: Le général Leclerc meurt de <u>la fièvre jaune</u> .	26.18 %
Syntactic variation	After the question is paraphrased into declarative form, its syntactic dependency structure does not match that of the answer sentence even after local modifications.	Q: Quelle est la profession d'Alain Testart? Sen.: Mais les différents anthropologues qui se qualifient d'évolutionnistes de nos jours, tel qu'Alain Testart et Christophe Darmangeat, proposent [...]	67.54 %
Multi sentence reasoning	There is anaphora, or higher-level fusion of multiple sentences is required.	Q: Les îles grecques sont elles toutes habitées? Sen.: La Grèce , d'une superficie de [...] et partage des frontières maritimes avec [...]. La mer Ionienne à l'ouest et [...], encadrent le pays dont le cinquième du territoire est constitué de plus de 9 000 îles et îlots dont près de 200 sont habités .	12.04 %
Ambiguous	We don't agree with the annotator's answer, or the question does not have a unique answer.	Q: Quel a été le résultat en demi-finale? Sen.: En demi-finale à Wembley, Arsenal du se débarrasser des tenant du titre, Wigan Athletic. Le match se termine sur un score d'1-1 . [...] Arsenal se qualifia aux tirs au but, 4-2 , avec [...]	9.95 %

Table 3: $N = 191$ randomly sampled triplets were manually assigned into one or more of the above categories. Words relevant to the corresponding reasoning type are in bold, and the annotated answer is underlined.

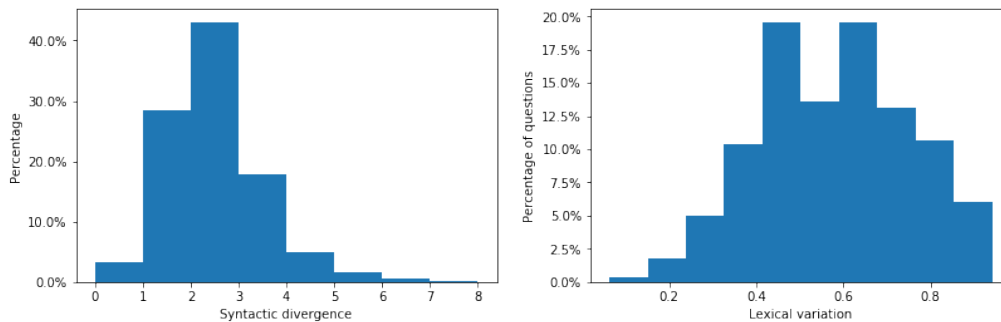


Figure 4: Distributions of syntactic divergence (left) and lexical variation (right) on question/sentence pairs for *PIAFv1.0*.

\downarrow train	eval \rightarrow	FQUAD	<i>PIAFv1.0</i>
SQuAD-Fr		78.39	68.90
FQUAD		78.96	66.30
SQuAD-Fr + FQUAD		81.09	71.11
SQuAD-Fr + FQUAD _{sub}		79.12	69
SQuAD-Fr + <i>PIAFv1.0</i>		79.48	-
FQUAD + <i>PIAFv1.0</i>		79.64	-
<i>all</i>		82.01	-

Table 4: F1 scores obtained after fine-tuning CamemBERT for QA on different training datasets. Models are evaluated on FQUAD (dev) and *PIAFv1.0* (when applicable). *all* refers to the union (shuffled) of FQUAD (train), SQuAD-Fr (train), and *PIAFv1.0*.

8. Acknowledgements

The PIAF project is supported by Etalab and receives funding from the Investing for the Future Program, led by the General Secretariat for Investment and the Deposits and Consignments Fund.

Our gratitude goes to all participants and previous/current/future contributors to the PIAF campaigns; to Djamé Seddah, Miriam Redi, and to all fellow researchers who provided valuable inputs.

We thank Project Nayuki for granting us permission to adapt and re-use their code, and the Mozilla Common Voice team for their precious feedback regarding the annotation tool and methodology.

9. Bibliographical References

- Cui, Y., Liu, T., Xiao, L., Chen, Z., Ma, W., Che, W., Wang, S., and Hu, G. (2018). A span-extraction dataset for chinese machine reading comprehension. *arXiv preprint arXiv:1810.07366*.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- d’Hoffschmidt, M., Vidal, M., Belblidia, W., and Brendlé, T. (2020). Fquad: French question answering dataset. *arXiv preprint arXiv:2002.06071*.
- Feng, M., Xiang, B., Glass, M. R., Wang, L., and Zhou, B. (2015). Applying deep learning to answer selection: A study and an open task. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 813–820. IEEE.
- Fisch, A. (2018). QA-Turk:intuitive annotation tool for extractive question answering. Software available from <https://github.com/ajfish/QA-Turk>.
- Gaikwad, S. N., Morina, D., Nistala, R., Agarwal, M., Cossette, A., Bhanu, R., Savage, S., Narwal, V., Rajpal, K., Regino, J., Mithal, A., Ginzberg, A., Nath, A., Ziulkoski, K. R., Cossette, T., Gamage, D., Richmond-Fuller, A., Suzuki, R., Herrejón, J., Le, K., Flores-Saviaga, C., Thilakarathne, H., Gupta, K., Dai, W., Sastry, A., Goyal, S., Rajapaksh, T., Abolhassani, N., Xie, A., Reyes, A., Ingle, S., Jaramillo, V., Godínez, M., Ángel, W., Toxtli, C., Flores, J., Gupta, A., Sethia, V., Padilla, D., Milland, K., Setyadi, K., Wajirasena, N., Batagoda, M., Cruz, R., Damon, J., Nekkanti, D., Sarma, T., Saleh, M., Gongora-Svartzman, G., Bateni, S., Toledo Barrera, G., Peña, A., Compton, R., Aariff, D., Palacios, L., Ritter, M. P., K.K., N., Kay, A., Uhrmeister, J., Nistala, S., Esfahani, M., Bakiu, E., Diemert, C., Matsumoto, L., Singh, M., Patel, K., Krishna, R., Kovacs, G., Vaish, R., and Bernstein, M. (2015). Daemo: A self-governed crowdsourcing marketplace. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST ’15 Adjunct*, pages 101–102, New York, NY, USA. ACM.
- Geva, M., Goldberg, Y., and Berant, J. (2019). Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. 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)*, pages 1161–1166.
- Joshi, M., Choi, E., Weld, D., and Zettlemoyer, L. (2017). Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1601–1611.
- Lee, K., Yoon, K., Park, S., and Hwang, S.-w. (2018). Semi-supervised training data generation for multilingual question answering. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan, May. European Languages Resources Association (ELRA).
- Lewis, P., Oğuz, B., Rinott, R., Riedel, S., and Schwenk, H. (2019). Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*.
- Lim, S., Kim, M., and Lee, J. (2018). Korquad: Korean qa dataset for machine comprehension.
- Liu, P., Deng, Y., Zhu, C., and Hu, H. (2019). Xcmrc: Evaluating cross-lingual machine reading comprehension. *arXiv preprint arXiv:1908.05416*.
- Martin, L., Muller, B., Ortiz Suárez, P. J., Dupont, Y., Romary, L., Villemonte de la Clergerie, É., Seddah, D., and Sagot, B. (2019). CamemBERT: a Tasty French Language Model. *arXiv e-prints*, page arXiv:1911.03894, Nov.
- Mikaelian, F., Farias, A., Amrouche, M., Nazon, T., and Sans, O. (2019). cdQA: An end-to-end closed domain question answering system. Software available from <https://github.com/cdqa-suite/cdQA-annotator>.
- Nakayama, H., Kubo, T., Kamura, J., Taniguchi, Y., and Liang, X. (2018). doccano: Text annotation tool for human. Software available from <https://github.com/chakki-works/doccano>.
- Quach, V. (2018). ieturk: Intuitive annotation tool for information extraction / named entity recognition using localturk / amazon mechanical turk. Software available from <https://github.com/Varal7/ieturk>.
- Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.
- Rajpurkar, P., Jia, R., and Liang, P. (2018). Know what you don’t know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.
- Reddy, S., Chen, D., and Manning, C. D. (2018). Coqa: A conversational question answering challenge. *arXiv preprint arXiv:1808.07042*.
- Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: A web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, EACL ’12*, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Trischler, A., Wang, T., Yuan, X., Harris, J., Sordani, A., Bachman, P., and Suleman, K. (2017). Newsqa: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200.
- Welbl, J., Stenetorp, P., and Riedel, S. (2018). Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association of Computational Linguistics*, 6:287–302.
- Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In

- Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018.
- Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W., Salakhutdinov, R., and Manning, C. D. (2018). Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.
- Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). Webanno: A flexible, web-based and visually supported system for distributed annotations. In *ACL (Conference System Demonstrations)*, pages 1–6. The Association for Computer Linguistics.