Interspeech 2018
2-6 September 2018, Hyderabad

Machine Learning powered Data Platform for High-Quality Speech and NLP workflows

João Dinis Freitas[1], *Jorge Ribeiro*[1], *Daan Baldewijns*[1], *Sara Oliveira*[1], *Daniela Braga*[1]

[1] DefinedCrowd Corporation

joao@definedcrowd.com, jorge@definedcrowd.com, daan@definedcrowd.com, sara@definedcrowd.com, daniela@definedcrowd.com

Abstract

Machine learning (ML) models - like deep neural networks - require substantial amounts of training data. Also, the training dataset should be properly annotated to obtain satisfactory results. This paper describes a platform designed to create high-quality datasets. By using data workflows adapted for speech technologies and natural language processing systems, the user can collect and enrich speech and text data. Depending on the end goal, the data is passed through multiple processing steps based on human input and ML services. To guarantee data quality, the platform combines several mechanisms like language tests, real-time audits, and user behavior into several ML models that act as quality gateways.

Index Terms: high-quality data, human input, data workflows, quality assessment.

1. Introduction

Data-driven applications and intelligent systems, such as personal assistants or autonomous vehicles, require large amounts of structured data. To enrich and structure the data (like images, audio or text) these systems require, we often need human input. This necessity has led to the recent growth of crowdsourcing platforms. These platforms allow to distribute micro-tasks across large numbers of people in exchange for a reward. However, depending on the task, the volume of low-quality contributions may be considerable, and manually reviewing micro-tasks may take as much or more time and effort than performing them. Also, many supervised learning techniques are highly dependent on data quality. Hence, when using methods like deep learning that require large amounts of data, it becomes fundamental to be able to build high-quality datasets in a scalable manner [1].

Crowdsourcing is a strategic model to attract a motivated crowd of individuals. These individuals, henceforth referred as contributors, can perform micro-tasks that take anywhere from a few seconds to several minutes to complete. The most appealing aspects of this approach, such as high throughput, low transaction costs, and human input on subjective tasks, also make it susceptible to quality control issues [2]. The required input in a micro-task may be subjective or even ambiguous, making quality control even harder. The reward behind each micro-task can also cause contributors to minimize their effort, rush the work, or even attempting to cheat the system to get the reward without any effort.

2. Platform description

The main goal of the platform built by DefinedCrowd[1] (DC) and described in this paper is to collect, enrich and structure large datasets in multiple languages. The platform is designed to serve data workflows that address the need to build diverse types of models in two domains: Natural Language Processing (henceforth NLP) and Speech Technologies (henceforth ST). As examples, we consider workflows that collect and process data to build acoustic models, as well as workflows to build named-entity recognizers and/or language models. A data workflow can informally be seen as a processing template with one or more steps, each with a specific input and output. For example, record speech, tag a named-entity in a sentence, etc. In the first step of any workflow the input can be either text or audio. The input of the remaining steps is the output of the previous step. These steps can be performed by humans or accomplished by automatic services. For example, a workflow to transcribe speech into text can use a step where an automatic speech recognition (ASR) service provides a baseline transcription before the human input is required, minimizing the effort of the contributors. Following the same idea, it is possible to configure any machine learning (ML) service to pre-annotate the data, transforming the next step (done by humans) into a validation and correction of the ML service output.

Every time a task is executed by a human, several processes take place: 1) add metadata to the collection and annotation results (e.g. speech time duration when the result is an audio file); 2) quality assessment (see section 4 below); 3) transformation of results into custom formats.

3. Data workflows

The data workflows on the DC platform can have one or more steps. In the sections below, we will present some of the possible steps (which can be viewed as a one-step workflow) and then illustrate how they can be combined into a more complex workflow.

3.1. NLP workflows

In NLP workflows, the input for a workflow is always a text corpus. This corpus may consist of situations for contributors to write about in a text-variant collection task or sentences and documents to be annotated. Given this input, the following types of tasks are available:

[1] https://www.definedcrowd.com/

Text Variant Collection: This workflow is used to collect free-text from contributors. The input is a list of scenarios, and the output generated by the contributors are sentences that exemplify those scenarios. To make the input unique across contributors, an option can be enabled to prevent the insertion of duplicate sentences as depicted in Figure 1.

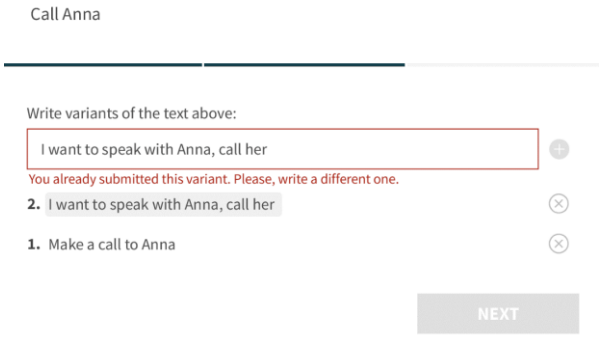

Figure 1: *Example of a contributor user interface for text-variant collection with duplication and spell checking activated.*

Domain and intent annotation: The goal of this workflow is to annotate the input text with the domain (i.e. what topic is the input about?) and the intent (i.e. what did the user try to accomplish?). This sort of data categorization is useful for building text-based interfaces (e.g. chatbots) and for those cases when annotated data is needed to create ML models to automatically classify the user data so that the system can respond appropriately to the user query. The ontology, i.e. the domains and intents, as well as which domains belong to which intents, is specified by the user.

Named-entity annotation: Depending on the application and area, the entities to be annotated in a given text range from person names, company names, dates, times, phone numbers, products, etcetera. The goal of this workflow is to have these entities annotated. Named-entity annotation works on the word or phrasal level. Hence, the input are sentences, but the output are the start and end character index of the entity, its category and the entity itself.

Sentiment annotation: Just as important as the content of user data is the way users perceive the sentiment of a given text. To cater to this need, the platform provides a workflow in which input data (be it text or audio) is annotated regarding whether the sentiment is positive, neutral, negative or uncertain.

3.2. ST workflows

In ST workflows the input can be a set of text prompts (e.g. speech data collection), audio (e.g. transcription) or both (e.g. text-audio correction).

Scripted Speech Data Collections: The input data for this workflow consists of phrases to be read by the contributors. The expected output are audio files aligned with the input phrases**.** However, the recorded speech often does not exactly match the prompt. Hence, to guarantee the text-audio alignment a validation step is often placed after this type of task. In data collections it is also possible to set demographic requirements (e.g. age and gender distributions), set audio parameters (e.g. sample rate and bit depth) or choose between mobile or desktop recordings.

Text-audio validation & correction: The input for this workflow is audio and text. The output is the corrected audio/transcription pair. Optionally, non-speech events like coughs or mispronunciations can be annotated.

3.3. Combining multiple steps

The one-step workflows mentioned before can be concatenated into a single workflow according to the user needs. An example of a workflow to create data to build a personal assistant is to combine the following steps: 1) text-variant collection; 2) speech data collection; 3) text-audio validation and correction; 4) domain and intent annotation; 5) named-entity annotation.

More complex workflows can also be configured. In these cases, instead of having linear data transfers, multiple steps are executed in parallel and their output transferred to a final step.

Each step can also be extended with custom controls that generate additional metadata (e.g. mark the speaker gender in a transcription task).

4. Quality Assessment

Finding the differences between low and high-quality work in human input is not straightforward. Depending on the type of task, it may mean looking into descriptiveness and quantity of information provided, and/or factors such as the member's fluency in the task's target language. Plus, the quality criteria will vary greatly across the various types of tasks [2]. As such, multiple quality gateways are used before, during, and after a contributor completes the task.

Before the contributor starts working on a task, passing a language test may be required. After the contributor applies for work, he/she will need to pass a qualification test with tasks similar to the actual tasks. During the execution of 'real' tasks, the contributor also has a (configurable) probability of getting a 'gold task' – a task to which the answer is known. Expected or acceptable behavior will change as the type of required input changes. A micro-task may consist of clicks, may require the generation of text, or it may require the recording of audio. As such, the platform stores the contributor's behavior and actions during the execution of tasks (e.g. keystrokes, mouse movement, or how long it took to read instructions).

When the data input is submitted, we analyze both the input and the behavior. For the input we look at the level of agreement with other contributors and consistency of the result. For example, in text collection tasks agreement is improbable, thus we use metrics directly related to the task, such as out-of-vocabulary occurrences and the probability of a sequence of words. As for behavior, we built multiple ML models trained with behavioral data of both malicious users and users with good intentions.

The combination of these quality gateways enables us to derive a score at both task and contributor level, which is used to assess the quality of a dataset at the end of a workflow.

5. References

[1] I. Goodfellow, Y. Bengio, A. Courville, Deep learning. Vol. 1. Cambridge: MIT press, 2016.

[2] J. M. Rzeszotarski and A. Kittur. "Instrumenting the crowd: using implicit behavioral measures to predict task performance." In Proceedings of the 24th annual ACM symposium on User interface software and technology, pp. 13-22. ACM, 2011.