EMBL2checklists: A Python package to facilitate the user-friendly submission of plant DNA barcoding sequences to ENA

Michael Gruenstaeudl¹*, Yannick Hartmaring²

1 Institut für Biologie, Freie Universität Berlin, 14195 Berlin, Germany
2 Institut für Bioinformatik, Freie Universität Berlin, 14195 Berlin, Germany

* m.gruenstaeudl@fu-berlin.de

Abstract

Background. The submission of DNA sequences to public sequence databases is an essential, but insufficiently automated step in the process of generating and disseminating novel DNA sequence data. Despite the centrality of database submissions to biological research, the range of available software tools that facilitate the preparation of sequence data for database submissions is low, especially for sequences generated via plant DNA barcoding. Current submission procedures can be complex and prohibitively time expensive for any but a small number of input sequences. A user-friendly software tool is needed that streamlines the file preparation for database submissions of DNA sequences that are commonly generated in plant DNA barcoding.

Methods. A Python package was developed that converts DNA sequences from the common EMBL and GenBank flat file formats to submission-ready, tab-delimited spreadsheets (so-called ”checklists”) for a subsequent upload to the public sequence database of the European Nucleotide Archive (ENA). The software tool, titled ”EMBL2checklists”, automatically converts DNA sequences, their annotation features, and associated metadata into the idiosyncratic format of marker-specific ENA checklists and, thus, generates output that can be uploaded via the interactive Webin submission system of ENA.

Results. EMBL2checklists provides a simple, platform-independent tool that automates the conversion of common plant DNA barcoding sequences into easily editable spreadsheets that require no further processing but their upload to ENA via the interactive Webin submission system. The software is equipped with an intuitive graphical as well as an efficient command-line interface for its operation. The utility of the software is illustrated by its application in the submission of DNA sequences of two recent plant phylogenetic investigations and one fungal metagenomic study.

Discussion. EMBL2checklists bridges the gap between common software suites for DNA sequence assembly and annotation and the interactive data submission process of ENA. It represents an easy-to-use solution for plant biologists without bioinformatics expertise to generate submission-ready checklists from common plant DNA sequence data. It allows the post-processing of checklists as well as work-sharing during the submission process and solves a critical bottleneck in the effort to increase participation in public data sharing.
Introduction

Only a few software tools assist in the preparation of DNA sequence data for submission to public sequence databases, despite the centrality of this process for generating and disseminating novel biological data. Contemporary biological research depends on the preservation, curation, and reproducibility of the data under study, and the submission of analyzed data to publicly accessible databases constitutes one of the most important best-practices in biology, particularly in the era of big data. DNA sequences generated to identify and characterize novel organisms or unchartered biodiversity must typically be submitted to public sequence databases before publication of the research is granted. Compliance with this prerequisite remains mixed. Several large nucleotide sequence repositories accept DNA sequence submissions, including GenBank, the European Nucleotide Archive or the DNA Data Bank of Japan. These repositories coordinate their policies and operations under the umbrella of the International Nucleotide Sequence Database Collaboration (INSDC), but each database employs custom procedures for sequence upload and data submission. ENA, for example, channels the submission of annotated DNA sequences through the Webin submission framework (https://www.ebi.ac.uk/ena/submit/sra), which, in its interactive version, operates with pre-formatted, tab-delimited spreadsheets (called "annotation checklists" or "templates") that are filled out by the user and then uploaded for submission. In order to account for different types of annotated DNA sequences (e.g., coding vs. non-coding, nuclear vs. organellar origin), a series of pre-tailored spreadsheets (hereafter "checklists") was developed by ENA, each with its idiosyncratic, tab-delimited fields of information. Users of ENA must choose the correct spreadsheet for their data submission, and different types of DNA sequences must be submitted via separate data uploads. Since June 2017, the submission process through Webin has been automated and now includes automatic validation procedures for annotation features, taxonomic metadata, and sequence integrity. Despite the centrality of data sharing to biological research, the range of user-friendly software tools that assist in data preparation for database submission is perceived as low. Indeed, very few contemporary, user-friendly software tools exist (e.g., Geneious) that facilitate the preparatory steps of annotated DNA sequences prior to uploading them to public sequence repositories. The number of software tools that assist with, and are specifically customized for, the preparation of common plant DNA barcoding sequences is particularly sparse.

Unlike sequence submissions to GenBank, the preparation of DNA sequence data for submission to ENA is insufficiently facilitated, highlighting the demand for software that converts annotated DNA sequences into submission-ready checklists. Upon DNA sequencing, researchers often utilize user-friendly software suites such as Artemis, Geneious or PhyDE for the assembly and annotation of DNA sequences. Some of these suits (e.g., Artemis, Geneious) enable the conversion of annotated DNA sequences to file formats that are easily submittable to GenBank, either by producing files in a direct submission format (i.e., the Sequin format) or through the processing with additional tools that convert GenBank-formatted flat files into the Sequin file format (e.g., GB2sequin). While there are similar software suites that generate EMBL-formatted flat files (e.g., Artemis, the "seqret" tool of EMBOSS), they do not produce output files that are suitable for a direct upload and submission via the interactive Webin submission system of ENA. To the best of our knowledge, no conversion tool currently exists that automatically converts annotated DNA sequences and associated metadata from the EMBL flat file format into submission-ready Webin checklists. The Webin submission system is the default gateway for DNA sequence submission to ENA and offers two routes for data upload: an interactive route, in which
Fig 1. Structural overview of the six DNA markers and their corresponding Webin checklists as implemented in EMBL2checklists. The overview displays the structural characteristics of each DNA marker and the idiosyncratic column names and numbers of the corresponding checklists. The DNA sequences displayed represent dummy sequences and are identical to the sequences of the example test files co-supplied with the software.
checklists are uploaded or generated online; and a programmatic route, in which both checklists and pre-formatted flat files can be submitted to the ENA server [12]. Flat files in EMBL file format are not currently accepted through the interactive, but only through the programmatic Webin submission route. However, using the programmatic route requires bioinformatics expertise and is, thus, inaccessible to many users. In absence of a more suitable solution, many users are compelled to undergo the tedious process of parsing information from EMBL-formatted flat files and copying them into Webin checklists manually, or to submit their data to repositories with a richer toolkit for submission preparation (e.g., GenBank). Some users also employ the guided web interface of the Webin submission platform, in which users click their way through an elaborate data entry interface. For any but a small number of input sequences, the use of the guided web interface is prohibitively time-expensive. Thus, there is a strong demand for an easy-to-use, platform-independent software tool that converts DNA sequences in EMBL flat file format including their annotation features and associated metadata into submission-ready checklists for upload via the Webin submission system.

Plant DNA barcoding is a key method to identify and characterize botanical specimens, and the resulting DNA sequences are well suited for a streamlined and automated conversion to marker-specific Webin checklists. The identification and characterization of plant specimens via the sequencing of specific genome regions ("plant DNA barcodes"; [22]) have become a key method in botanical research [23]. Thousands of DNA sequences have been generated in investigations on suitable plant DNA barcoding markers [24-26], and plant DNA barcoding is now routinely applied across evolutionary, ecological and conservation research [22,23,27], even in regional studies [25,30]. Future investigations applying plant DNA barcoding will invariably require the submission of novel DNA sequences to public sequence repositories [7,27], and a user-friendly, streamlined data submission process can be instrumental to their data sharing process [3]. In fact, plant DNA barcoding sequences lend themselves for the application of software tools that streamline and automate their submission process, because common barcoding markers display (a) a general homogeneity in sequence length and gene synteny, at least within most target lineages [31,32], (b) a general absence of structural inversions or strong secondary structure [32,33], and (c) the presence of conserved regions, preferentially in the flanking parts of the marker, that allow the anchoring of universal PCR primers and, thus, bidirectional sequencing [31,33]. Unfortunately, the preparation process for submissions of DNA sequences that are commonly generated in plant DNA barcoding experiments is insufficiently automated, especially to the sequence database of ENA. A software tool is needed that streamlines and automates the preparation of such barcoding sequences, primarily for submissions via the interactive Webin submission system of ENA.

The plant sciences community would benefit from a software tool that streamlines the generation of marker-specific Webin checklists, which is tedious to conduct by hand and difficult to code dynamically due to the idiosyncrasies of the different checklists (Fig. 1). Specifically, it would be desirable to automate the conversion of EMBL- or GenBank-formatted flat files, which can be generated via various software suites for DNA sequence assembly and annotation (e.g., Geneious, Artemis), into correctly formatted Webin checklists that require nothing more but their upload to ENA via the interactive Webin submission system (Table 1). The software should hereby be available as an open-source tool that can be customized and expanded by other researchers. In the present investigation, we present such a tool. We report about the development and application of a Python package, entitled "EMBL2checklists", that takes annotated DNA sequences of common plant DNA barcoding regions and associated metadata as input and returns properly-formatted checklists that are ready for data upload to ENA via the interactive Webin submission system (Fig. 2).
Table 1. Overview of the bioinformatic steps involved in submitting novel DNA sequence data to ENA while using EMBL2checklists, starting from assembled DNA sequences.

| Bioinformatic step       | Details of step                                                                 | Manual processing | Available software                  |
|--------------------------|----------------------------------------------------------------------------------|-------------------|-------------------------------------|
| 1 Sequence annotation    | Sequence features and feature qualifiers using INSDC-compatible keywords are added to each DNA sequence. | high              | Geneious¹, Artemis¹,²                |
| 2 Saving as flat file    | Multiple annotated DNA sequences are saved as a single multi-sequence EMBL- or GenBank-formatted flat file. | minimal           | Geneious¹, Artemis¹,²                |
| 3 Flat file validation   | flat file format, feature table syntax or taxonomic status of organism, among other aspects, are validated. | minimal           | EMBL flat file validator¹ (cit.), SBOL validator² (cit.) |
| 4 Conversion to checklist| The EMBL- or GenBank-formatted flat file is converted into the Webin checklist file format. | minimal           | EMBL2checklists¹,²                  |
| 5 Upload to ENA          | The checklist is uploaded to ENA via the interactive route of the Webin submission system. | minimal           | Any web browser                     |

Table notes. ¹For EMBL-formatted flat files; ²For GenBank-formatted flat files
Materials and methods

Implemented checklist types

The software EMBL2checklists was designed to convert annotated DNA sequences and associated metadata into six different Webin checklist types. Sequence submission via the Webin submission system is primarily conducted via marker-specific checklists [15]. These checklists are pre-tailored to the idiosyncrasies of different genome regions, with different checklists displaying marker-specific customizations in order to capture the distinct information of the genomic regions under study (Fig. 1). A conversion of annotated DNA sequences into Webin checklists, thus, needs to take these marker-specific customizations into account. For example, a checklist that contains sequence information on the plastid trnK/matK region will be more complex than a checklist on the nuclear ribosomal external transcribed spacer (ETS) due to the location of the gene matK inside the group II intron of the tRNA gene for Lysine (trnK-UUU; [34]). Thus, the Webin checklist for trnK/matK comprises eight mandatory columns, whereas the Webin checklist for the ETS comprises only five (Table 2). To enable submissions of a wide range of genomic regions to ENA, numerous marker-specific checklists have been implemented in Webin (https://www.ebi.ac.uk/ena/submit/annotation-checklists), but only a small number of these are relevant to plant DNA barcoding. The software EMBL2checklists is designed to convert annotated DNA sequences and associated metadata into one of six different Webin checklists. Given an EMBL- or GenBank-formatted flat file with the correct annotation features and feature qualifiers as input (Table 2), EMBL2checklists can generate marker-specific checklists for a series of DNA markers that are commonly employed in plant DNA barcoding. These markers are: (i) a common gene intron (e.g., trnL intron; [35]); (ii) a common intergenic spacer (IGS; e.g., trnH-psbA; [36]); (iii) the plastid trnK/matK region [37]; (iv) the nuclear ribosomal rRNA-encoding rDNA genes (e.g., 18S rDNA; [38]); (v) the nuclear ribosomal internal transcribed spacer (ITS; [24]); and (vi) the nuclear ribosomal ETS ([39]).

Conversion of multiple sequence records

EMBL2checklists is able to convert multiple sequence records contained in input flat file into a single Webin checklist. EMBL- or GenBank-formatted flat files may contain multiple sequence records, each with a specific set of annotation features and sequence metadata. EMBL2checklists accepts such a user-selected flat file as input, parses each sequence record individually, and writes the parsed information to the output file. Specifically, the software converts the sequence information contained in each sequence record into a separate line of the resulting checklist. Programmatically, EMBL2checklists parses the flat file via the BioPython library [40] and then iterates through the sequence records, processing one record at a time (Fig. 3). During each iteration, the DNA sequence of a record, its annotation features, and its associated metadata are extracted. The extracted information is formatted to a pre-tailored, tab-delimited spreadsheet, the precise type of which had been selected by the user during program initiation. Eventually, EMBL2checklists collects the information of all successfully processed records and appends it to the output file so that the number of lines written to the checklist equals the number of sequence records successfully parsed from the input. This record-by-record processing of the input file allows the parsing algorithm to evaluate the sequence records individually and to skip specific records in the event of an error.
Fig 2. Overview of the conversion process from an EMBL-formatted flat file to a submission-ready Webin checklist via the application of EMBL2checklists. Name and content of the input and output files displayed are identical to the corresponding example test files co-supplied with the software.
Table 2. Overview of the mandatory column name, annotation feature and feature qualifier specifications of the six checklist types implemented in EMBL2checklists.

| Checklist name and description          | Column name | Column info | Flat file feature | Feature qualifier |
|-----------------------------------------|-------------|-------------|-------------------|-------------------|
| gene_intron - any gene intron           | GENE        | str         | gene              | note              |
|                                         | 5'_PARTIAL* | yes/no      | intron            |                   |
|                                         | 3'_PARTIAL* | yes/no      | intron            |                   |
|                                         | 5'_INTRON*  | int         | intron            |                   |
|                                         | 3'_INTRON*  | int         | intron            |                   |
|                                         | NUMBER      | int         | intron            | number            |
| IGS - any intergenic spacer             | GENE1       | str         | misc_feature      | product           |
|                                         | G1PRESENT   | yes/no      | gene              | note              |
|                                         | GENE2       | str         | misc_feature      | product           |
|                                         | G2PRESENT   | yes/no      | gene              | note              |
| trnK_matK - plastid trnK/matK gene region | 5'_CDS      | yes/no      | gene              | note              |
|                                         | 3'_CDS      | yes/no      | gene              | note              |
|                                         | 5'_PARTIAL  | yes/no      | gene              | note              |
|                                         | 3'_PARTIAL  | yes/no      | gene              | note              |
|                                         | INTRON*     | yes/no      | intron, tRNA     | gene              |
| rRNA - 18S/28S/5.8S nr rDNA gene        | SEDIMENT    | 18S/28S/5.8S| rRNA             | product           |
| ITS - nr internal transcribed spacer    | ISOLATION   | str         | source            | isolation         |
|                                         | _SOURCE*    |             |                   |                   |
|                                         | 18S         | yes/no      | rRNA              | gene              |
|                                         | ITS1*       | yes/no      | misc_rRNA        | note              |
|                                         | ITS2*       | yes/no      | misc_rRNA        | note              |
|                                         | 28S         | yes/no      | rRNA              | gene              |
| ETS - nr external transcribed spacer    | 18S*        | yes/no      | rRNA              | gene              |
|                                         | ETS_TYPE*   | 5'/3'       | misc_rRNA        | note              |
|                                         | 28S*        | yes/no      | rRNA              | gene              |

Table notes. Checklist and column names are displayed as defined by the Webin submission interface. The mandatory checklist columns "ORGANISM_NAME", "ENV_SAMPLE" and "SEQUENCE" are present in all checklist types and, thus, not displayed. The number of mandatory and optional checklist columns sums up to the total number of implemented columns for each checklist type. The application of special parsing rules (see main text) is indicated by an asterisk. For checklist type "rRNA", any of the following sediment types are permitted: 5S, 5.8S, 12S, 16S, 18S, 23S, 25S, 26S, 28S. Abbreviations used: nr = nuclear ribosomal; str = string; int = integer.
EMBL2checklists conducts a series of data parsing steps that are executed sequentially for each sequence record and that include evaluations of feature prerequisites and the coherence between input data checklist selection. The structure of EMBL2checklists consists of two main processes that are executed sequentially for each sequence record: input audit and data processing (Fig. 3). Upon initialization, EMBL2checklists evaluates the presence, integrity, and syntax of the input flat file via the data parser of BioPython. Then, it proceeds to input audit and data processing. During input audit, two different checks are conducted on a given sequence record. These checks are implemented as separate functions and comprise the evaluation if (a) the DNA marker abbreviations found among the annotation features and their qualifiers are coherent with the user-selected checklist type ("parse marker abbreviations" in Fig. 3), and (b) the sequence record contains the minimally necessary annotation features to generate a functional checklist ("feature prerequisites"). Thus, the input audit evaluates if a given sequence record contains the necessary annotation features, feature qualifiers and qualifier values for the chosen checklist type. For example, a sequence record of a ribosomal DNA region that does not contain at least one feature of class "rRNA" in the feature table and in which that feature does not contain at least one qualifier of class "product" is flagged as an error because information on the mandatory checklist qualifier "SEDIMENT" cannot be parsed from such a sequence record (Table 2). Sequence records that fail the evaluation of minimal feature prerequisites are skipped, whereas those that fail the parsing of correct marker abbreviations terminate the entire software execution (Fig. 3) because the latter error is indicative of an incorrect checklist selection by the user. Upon successful input audit, data processing of the sequence record is started. During data processing, the annotation features and feature qualifiers of a sequence record relevant to the chosen checklist type are parsed. First, only those qualifiers are parsed that a valid checklist for the selected checklist type could not be generated without (i.e., "mandatory qualifiers"; Fig. 3). Then, qualifiers are parsed, which are permissible but not required for the selected checklist type (i.e., "optional qualifiers"). During each parsing step, the spelling of feature and qualifier names is evaluated against an INSDC-compliant dictionary of feature definitions, which serves two purposes. First, it ensures that feature and qualifier names are spelled and formatted in compliance with the definitions of the INSDC [14]. Second, it ensures the correct transfer of information from the annotation features into the columns of the resulting Webin checklists. If an error is encountered during data processing, the processing of that sequence record fails and raises an exception while communicating a short error message to the user. Upon data processing, the information of the current sequence record is saved into an output handle and the data audit for the next record initiated. Upon processing all sequence records, the writer function of EMBL2checklists appends the parsed information of each sequence record as independent lines to the output file.

Special parsing rules

EMBL2checklists applies a series of special parsing rules in order to accommodate the idiosyncratic structure and information content of the different checklist types. Data processing is not homogeneous across all checklist types but includes the application of special parsing rules for select checklists. For the checklist type "gene_intron", for example, the start and end position ("5'_INTRON" and "3'_INTRON" in Table 2) and the completeness of the intron ("5'PARTIAL" and "3'PARTIAL") is determined by the intron location information, not by its qualifier values. For checklist type "trnK_matK", either an intron feature or a tRNA feature for gene trnK-UUU must be
Fig 3. Overview of the internal structure of EMBL2checklists. The overview illustrates the two main processes that are executed sequentially for each sequence record: input audit and data processing. Rhomboid fields indicate the start and end points of the loop across sequence records. Field with dotted outlines indicate states when standard data processing fails.
present in the sequence record. For checklist type "ITS", two special rules apply: (a) if environmental samples are processed, the otherwise optional checklist column "ISOLATION_SOURCE" becomes mandatory; and (b) the completeness of the rDNA gene 5.8S is inferred based on the presence of ITS1 and ITS2. For checklist type "ETS", either a rRNA feature for the 18S and the 28S rDNA gene, or a misc_rRNA feature with the info "5' ETS" or "3' ETS" must be present in the sequence record. To maintain an accurate implementation of these parsing rules across different development stages, customary checks via the Python "unittest" framework [11] were added to the software.

Input parameters and output specifications

Upon initializing the software, users of EMBL2checklists must specify four input parameters (Fig. 2): (a) the name of the input flat file; (b) the name of the checklist output file; (c) the type of checklist selected by the user, and (d) a statement if the sequence records classify as environmental samples. Input parameter (a) must contain the name of, and the file path to, an EMBL- or GenBank-formatted flat file that comprises one or more sequence records. The precise file format of the input flat file is automatically identified based on the file ending (i.e., "embl" or ".gb"). The structure of each sequence record must be compliant with the identified flat file format and consist of a multi-line feature table, followed by the interleaved DNA sequence. The source feature, which pertains to the sequence as a whole and often contains sequence metadata, must be located at the top of the feature table. Annotation features, which indicate the type and boundaries of localized sequence features, must be located below the source feature. Input parameter (b) must contain the name of, and the file path to, the output checklist. The checklists generated by EMBL2checklists are human-readable and can be edited by any common spreadsheet editor such as Microsoft Excel (Microsoft Corporation, Redmond, WA, USA) or LibreOffice (The Document Foundation, Berlin, Germany). If all sequence records of the input file are processed correctly, the output checklist displays as many lines as the number of sequence records in the input file (disregarding the title line of the checklist). Input parameter (c) must be the name of one of six checklist types implemented in EMBL2checklists and specifies the rule set applied during input audit and data parsing of the sequence records. Input parameter (d) must be either "yes" or "no" and specifies the classification of all sequence records of an input file as environmental samples. This parameter is implemented in EMBL2checklists because its information is mandatory for certain Webin checklist types. It is typically answered with "yes" if the DNA sequences under study were generated as part of a metabarcoding experiment. Upon specifying each of these parameters, EMBL2checklists begins to process the input file.

Commandline and graphical user interface

EMBL2checklists was developed for classical plant biologists and bioinformaticians alike. Thus, the software is equipped with a graphical user interface (GUI) as well as a command-line interface (CLI) for its operation (Fig. 2). The GUI is based on the Python library "Tkinter" [42] and designed to provide an intuitive and easy-to-use interface that allows users with little or no bioinformatics knowledge to operate the software. To execute EMBL2checklists via the GUI, users enter the four input parameters via the available input fields and drop-down menus. To receive help and detailed explanations via the GUI, users may hover their mouse pointer over a field of interest, which initiates a help bubble next to the pointer. Moreover, details of individual checklist types are automatically displayed upon selecting one of the implemented checklists from the drop-down menu. If EMBL2checklists raises an exception while operating under the GUI, an error message is printed to a pop-up
window of the GUI. The GUI can be accessed through file "EMBL2checklists_GUI.py" of the scripts folder. More information on the design and functionality of the GUI of EMBL2checklists is available in [43]. The CLI employs functions of the Python library "argparse" [44] and allows more experienced users to execute the software via the command-line and to integrate the software into larger bioinformatic workflows. To execute EMBL2checklists via the GUI, users specify the name of the input flat file via command-line argument "-i", the name of the checklist output file via argument "-o", the type of checklist via argument "-c", and the classification of the sequence records as environmental samples via argument "-e". To receive help and detailed explanations, CLI users may invoke command-line argument "-h" (i.e., "python2 script/EMBL2checklists_CLI.py -h"). If EMBL2checklists raises an exception while operating under the CLI, an error message is printed to the standard output stream. The CLI can be accessed through file "EMBL2checklists_CLI.py" of the scripts folder.

Release, installation and operation

EMBL2checklists was written in Python 2.7 [45] and is, thus, platform independent. It can be executed on any system equipped with a Python 2 compiler and after the installation of the necessary Python dependencies. The software uses three separate Python packages as dependencies: Biopython [40], argparse [44] and Tkinter [42]. EMBL2checklists is open source and released under the BSD 3-Clause license (https://opensource.org/licenses/BSD-3-Clause). Other bioinformaticians are, thus, able to expand and customize the software to fit their own data submission projects. EMBL2checklists is available via the Python Package Index (https://pypi.org/project/EMBL2checklists/) and can be installed via any PyPI-compatible package management system such as pip (https://pip.pypa.io) or setuptools (https://pypi.org/project/setuptools/). For example, users may type the following command to install EMBL2checklists on their system:

$ pip2 install EMBL2checklists

EMBL2checklists was successfully tested on a Windows (Microsoft Windows 7), a MacOS (MacOS 10.12.6 - Sierra) and two different Linux environments (Arch Linux 4.18.9 and Debian 9.0). For a typical execution of EMBL2checklists via the CLI, a user may type the following command:

$ python2 scripts/EMBL2checklists_CLI.py \
   -i example/example_trnKmatK.embl \
   -o example/example_trnKmatK.tsv \
   -c trnK_matK \
   -e no

Results and Discussion

Application of software on empirical data

The utility of EMBL2checklists to plant biology is illustrated by its application in the submission process of DNA sequences to ENA by three recent investigations. Specifically, EMBL2checklists was employed for the preparation of hundreds of DNA sequences used in two plant phylogenetic and one fungal metagenomic investigation [46–48]. The two plant phylogenetic investigations utilized common plant DNA barcoding markers to infer the phylogenetic history of select plant lineages [46,47]. The fungal metagenomic investigation utilized nrDNA barcodes to characterize arbuscular mycorrhizal soil fungi [48]. In each case, EMBL2checklists was used to
convert flat files in GenBank format that were generated from sets of assembled and annotated sequences via the software suite Geneious. Upon conversion to checklists, the sequence data was uploaded to ENA via the interactive Webin submission system, and accession numbers were received from ENA by email within less than 48 hours of submission.

Post-processing of checklists and work-sharing

Due to the data format of Webin checklists and the structure of the interactive Webin submission process, the use of EMBL2checklists for sequence submissions to ENA displays two special advantages: it allows the simple post-processing of checklists, and it enables work-sharing during the submission process. First, the data format of Webin checklists is human-readable and, thus, allows the manual addition of column information, should users wish to augment the checklist prior to submission. Specifically, the easily-accessible checklist data structure allows users to modify or append column content, as long as the name and order of existing columns remain unchanged. For example, users who wish to add information about the geographic location of one or more sequences after having processed the sequence data with EMBL2checklists can simply add a column entitled "LOCALITY" before column "SEQUENCE" of the checklist and add geographic location information for one or more sequences. Likewise, users who wish to combine multiple checklists can do so, as long as the number and order of columns are identical (and the title line of the second checklist is removed).

Second, EMBL2checklists allows the implementation of a work-sharing strategy during the preparation of sequence submissions because the information contained within the identification, description and reference lines of a sequence record (such as author name or author institution) is not saved as part of the checklist output. The software only processes the feature table and the DNA sequence of a sequence record. Ancillary information of a dataset must be associated with the sequence data during the interactive submission process. Specifically, personal and institutional information of the submitter is associated with the data through the Webin submission service prior to data upload, irrespective of the checklist type. Hence, EMBL2checklists does not need to be executed by the same person that conducts the data upload or has generated the sequence but allows a work-sharing strategy in which one person (or section of a workflow) conducts the data conversion via EMBL2checklists, while another person (or section of a workflow) conducts the data submission. Work-sharing may be helpful if the sequence submission process is centralized within a lab or academic institution, allowing those researchers that prepare the data for submission to ENA to be different from those that actually conduct the data upload.

Data converters and other ENA submission strategies

The paucity of file formats acceptable for data submission to public sequence databases is one of the main bottlenecks in the effort to increase participation in public data sharing and has spurred the recent development of various data converters. The software EMBL2checklists is one of several current projects that aim to provide automated data conversion between the EMBL or GenBank flat file format and data formats that are commonly parsed by biological software and databases [20, 49, 50]. The underlying aim of many of these projects is to simplify the conversion process of sequence data into file formats that are accepted during submissions to public sequence databases [20, 49]. Given the custom validation criteria and the idiosyncratic submission procedures employed by many of these databases, such data converters represent an important means to enable user-friendly data submissions, at least until such time as submission procedures across INSDC databases are standardized [51]. EMBL2checklists
was specifically designed to bridge the gap between common software suites for DNA sequence assembly and annotation (e.g., Artemis, Geneious) and the interactive data submission process of ENA. Compared to most other recent data converters, EMBL2checklists supports the conversion of flat files that contain multiple sequence records. At the same time, EMBL2checklists aims to fulfill a seemingly counterintuitive task: The software converts EMBL- or GenBank-formatted flat files into Webin checklists, which the receiving database of ENA eventually converts back into the flat file format. Theoretically, it would be easier to upload annotated DNA sequences in EMBL flat file format to ENA directly, but presently only the Entry Upload service (https://ena-docs.readthedocs.io/en/latest/prog_12.html) and, to lesser extent, the Webin command-line submission interface (https://github.com/enasequence/webin-cli) support flat file submissions of this data type. In practice, both of these programmatic services exclude a considerable number of regular users aiming to upload annotated DNA sequences to ENA due to the bioinformatics expertise required. Until methods are in place that allow a more user-friendly upload of annotated DNA sequences in flat file format to ENA, a back-conversion to checklists as conducted by our software remains the most practical solution. Thus, EMBL2checklists currently represents the most user-friendly option for plant biologists without bioinformatics experience to automatically generate submission-ready checklists from common plant DNA sequence data.

Conclusion

The lack in automated conversion between EMBL- or GenBank formatted flat files and submission-ready Webin checklists represented a gap that compelled many researchers to conduct manual data processing before submitting data to the public sequence database ENA. By developing the software EMBL2checklists, we have filled this gap. EMBL2checklists is designed as an easy-to-use software application that bridges the gap between common software suites for DNA sequence assembly and annotation and the interactive data submission process of ENA. The software converts annotated DNA sequences plus associated metadata into properly-formatted Webin checklists for submission to ENA. Specifically, the software takes the idiosyncrasies of marker-specific checklist types into account and generates submission-ready checklists for specifically those DNA markers that are commonly employed in plant DNA barcoding. Thus, EMBL2checklists can be employed to prepare the most common plant DNA barcoding marker sequences for upload and submission to ENA via the interactive Webin submission system. Users may generate input files for our software through any of several common sequence analysis environments such as Artemis or Geneious and then employ the GUI of EMBL2checklists to prepare their sequence submissions. Upon processing with EMBL2checklists, the user receives a checklist that can be directly uploaded to ENA or further edited with a common spreadsheet editor. The utility of EMBL2checklists is best illustrated by its application during the submission preparation of hundreds of DNA sequences submitted to ENA during the publication process of several recent investigations [46–48]. With the development of EMBL2checklists, we hope to provide a useful software tool to plant biologists and bioinformaticians alike, increase the amount of sequence data deposited to public sequence databases [7] and advance the idea of publicly-shared research data [9,10]. By extension, we believe that EMBL2checklists may play an important role in future data management and data stewardship of plant DNA sequence data under the FAIR data principle [52,53].
Acknowledgments

The authors would like to acknowledge the high-performance computing service of the ZEDAT of the Freie Universität Berlin for providing allocations of computing time. The development of a GUI for the presented software constitutes part of a thesis by YH toward a bachelor of science degree.

Data availability statement

EMBL2checklists is available free of charge via the Python Package Index (https://pypi.org/) under https://pypi.org/project/EMBL2checklists/ and can be installed via any PyPI-compatible package management system such as pip (https://pip.pypa.io) or setuptools (https://pypi.org/project/setuptools/). The source code of EMBL2checklists is available via the GitHub page of MG under https://github.com/michaelgruenstaeudl/EMBL2checklists.

Funding statement

This investigation was supported by a start-up grant (Initiativmittel der Forschungskommission) of the Freie Universität Berlin to MG.

Author contributions

MG conceived the software, wrote its core functions and generated an alpha version of the software; YH added additional functions to the software and wrote its GUI; MG applied the software on the empirical data; MG wrote the manuscript.

References

1. Tenopir C, Allard S, Douglass K, Aydinoglu AU, Wu L, Read E, et al. Data sharing by scientists: Practices and perceptions. PLoS ONE. 2011;6:1–21.
2. Drew BT, Gazis R, Cabezas P, Swithers KS, Deng J, Rodriguez R, et al. Lost branches on the tree of life. PLoS Biology. 2013;11:1–5.
3. Huang X, Hawkins BA, Lei F, Miller GL, Favret C, Zhang R, et al. Willing or unwilling to share primary biodiversity data: Results and implications of an international survey. Conservation Letters. 2012;5:399–406.
4. Whitlock MC. Data archiving in ecology and evolution: best practices. Trends in Ecology & Evolution. 2011;26:61–65.
5. Hampton SE, Strasser CA, Tewksbury JJ, Gram WK, Budden AE, Batcheller AL, et al. Big data and the future of ecology. Frontiers in Ecology and the Environment. 2013;11:156–162.
6. Fairbairn DJ. The advent of mandatory data archiving. Evolution. 2011;65:1–2.
7. Blaxter M, Danchin A, Fukami-kobayashi K, Sugano S, Richard J, Salzberg SL, et al. Reminder to deposit DNA sequences. Science. 2016;352:780.
8. Noor MAF, Zimmerman KJ, Teeter KC. Data sharing: How much doesn’t get submitted to GenBank? PLoS Biology. 2006;4:1113–1114.
9. Roche DG, Lanfear R, Binning SA, Haff TM, Schwanz LE, Cain KE, et al. Troubleshooting public data archiving: Suggestions to increase participation. PLoS Biology. 2014;12.

10. Roche DG, Kruuk LEB, Lanfear R, Binning SA. Public data archiving in ecology and evolution: How well are we doing? PLoS Biology. 2015;13:1–12.

11. Benson DA, Cavanaugh M, Clark K, Karsch-Mizrachi I, Ostell J, Pruitt KD, et al. GenBank. Nucleic Acids Research. 2018;46:D41–D47.

12. Silvester N, Alako B, Amid C, Cerdeño-Tárraga A, Clarke L, Cleland I, et al. The European Nucleotide Archive in 2017. Nucleic Acids Research. 2018;46:D36—D40.

13. Kodama Y, Mashima J, Kosuge T, Kaminuma E, Ogasawara O, Okubo K, et al. DNA Data Bank of Japan: 30th anniversary. Nucleic Acids Research. 2018;46:D30–D35.

14. Karsch-Mizrachi I, Takagi T, Cochrane G. The international nucleotide sequence database collaboration. Nucleic Acids Research. 2018;46:D48–D51.

15. Amid C, Birney E, Bower L, Cerdeño-Tárraga A, Cheng Y, Cleland I, et al. Major submissions tool developments at the European nucleotide archive. Nucleic Acids Research. 2012;40:43–47.

16. Kearse M, Moir R, Wilson A, Stones-Havas S, Cheung M, Sturrock S, et al. Geneious Basic: An integrated and extendable desktop software platform for the organization and analysis of sequence data. Bioinformatics. 2012;28:1647–1649.

17. Rutherford K, Parkhill J, Crook J, Horsnell T, Rice P, Rajandream MA, et al. Artemis: Sequence visualization and annotation. Bioinformatics. 2000;16:944–945.

18. Müller J, Müller K, Neinhuis C, Quandt D. PhyDE: Phylogenetic Data Editor. Available from: http://www.phyde.de/

19. Benson DA, Boguski MS, Lipman DJ, Ostell J, Ouellette BF. GenBank. Nucleic Acids Res. 1998;26:1–7.

20. Lehwaark P, Greiner S. GB2sequin - A file converter preparing custom GenBank files for database submission. Genomics. 2018;in press:

21. Rice P, Longden I, Bleasby A. EMBOSS: The European Molecular Biology Open Software Suite. Trends in Genetics. 2000;16:276–277.

22. Kress WJ. Plant DNA barcodes: Applications today and in the future. Journal of Systematics and Evolution. 2017;55:291–307.

23. Hollingsworth PM, Li DZ, Van Der Bank M, Twyford AD. Telling plant species apart with DNA: From barcodes to genomes. Philosophical Transactions of the Royal Society B: Biological Sciences. 2016;371.

24. Li DZ, Gao LM, Li HT, Wang H, Ge XJ, Liu JQ, et al. Comparative analysis of a large dataset indicates that internal transcribed spacer (ITS) should be incorporated into the core barcode for seed plants. Proceedings of the National Academy of Sciences. 2011;108:19641–19646.

25. Liu J, Shi L, Han J, Li G, Lu H, Hou J, et al. Identification of species in the angiosperm family Apiaceae using DNA barcodes. Molecular Ecology Resources. 2014;14:1231–1238.
26. Pei N, Erickson DL, Chen B, Ge X, Mi X, Swenson NG, et al. Closely-related taxa influence woody species discrimination via DNA barcoding: Evidence from global forest dynamics plots. Scientific Reports. 2015;5:1–9.

27. Joly S, Davies TJ, Archambault A, Bruneau A, Derry A, Kembel SW, et al. Ecology in the age of DNA barcoding: The resource, the promise and the challenges ahead. Molecular Ecology Resources. 2014;14:221–232.

28. Heckenhauer J, Abu Salim K, Chase MW, Dexter KG, Pennington RT, Tan S, et al. Plant DNA barcodes and assessment of phylogenetic community structure of a tropical mixed dipterocarp forest in Brunei Darussalam (Borneo). PLoS ONE. 2017;12:1–24.

29. Tan Sl, Hollingsworth YhLPM, Li KXDz, Gao Lm, Burgess KS. DNA barcoding herbaceous and woody plant species at a subalpine forest dynamics plot in Southwest China. Ecology and Evolution. 2018;in press.

30. Dean GH, Asmarayani R, Ardiyani M, Santika Y, Triono T, Mathews S, et al. Generating DNA sequence data with limited resources for molecular biology: Lessons from a barcoding project in Indonesia. Applications in Plant Sciences. 2018:6:1–12.

31. CBOL-Plant-Working-Group. A DNA barcode for land plants. Proceedings of the National Academy of Sciences 2009;106:12794–12797.

32. Li X, Yang Y, Henry RJ, Rossetto M, Wang Y, Chen S. Plant DNA barcoding: from gene to genome. Biological reviews of the Cambridge Philosophical Society. 2015;90:157–166.

33. Hollingsworth PM, Graham SW, Little DP. Choosing and using a plant DNA barcode. PLoS ONE. 2011;6.

34. Neuhaus H, Link G. The chloroplast tRNAlys(UUU) gene from mustard (Sinapis alba) contains a class II intron potentially coding for a matrerase-related polypeptide. Current Genetics. 1987;07:251–257.

35. Taberlet P, Coissac E, Ponapanon F, Gielly L, Miquel C, Valentini A, et al. Power and limitations of the chloroplast trnL (UAA) intron for plant DNA barcoding. Nucleic Acids Research. 2007;35.

36. Kress WJ, Erickson DL. A two-locus global DNA barcode for land plants: The coding rbcL gene complements the non-coding trnH-psbA spacer region. PLoS ONE. 2007;2.

37. Wicke S, Quandt D. Universal primers for the amplification of the plastid trnK/matK region in land plants. Anales Jard Bot Madrid. 2009;66:285–288.

38. Soltis DE, Soltis PS, Nickrent DL, Johnson LA, William J, Hoot SB, et al. Angiosperm phylogeny inferred from 18S ribosomal DNA sequences. Annals of the Missouri Botanical Garden. 1997;84:1–49.

39. Baldwin BG, Markos S. Phylogenetic utility of the external transcribed spacer (ETS) of 18S-26S rDNA: congruence of ETS and ITS trees of Calycadenia (Compositae). Molecular Phylogenetics and Evolution. 1998;10:449–463.

40. Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, et al. Biopython: Freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics. 2009;25:1422–1423.
41. Python-Software-Foundation. Unit testing framework. Available from: https://docs.python.org/2/library/unittest.html

42. Shipman JW. Tkinter 8.5 reference: a GUI for Python. New Mexico Tech Computer Center. 2013.

43. Hartmaring Y. Ausbau eines Software-Scripts für die Konvertierung von EMBL flatfiles zu ENA checklists, inklusive Entwicklung eines GUI und beispielhafter Anwendung an empirischen Daten. Bachelor thesis, Freie Universität Berlin, 2018.

44. Waldmann T. argparse. Available from: https://pypi.python.org/pypi/argparse

45. Python-Software-Foundation. Python Language Reference. Available from: https://www.python.org/

46. Korotkova N, Parolly G, Khachatryan A, Ghulikyan L, Sargsyan H, Akopian J, et al. Towards resolving the evolutionary history of Caucasian pears (Pyrus, Rosaceae)—Phylogenetic relationships, divergence times and leaf trait evolution. Journal of Systematics and Evolution. 2018;56:35–47.

47. Canal D, Köster N, Jones KE, Korotkova N, Croat TB, Borsch T. Phylogeny and diversification history of the large Neotropical genus Philodendron (Araceae): Accelerated speciation in a lineage dominated by epiphytes. American Journal of Botany. 2018;105:1035–1052.

48. Roy J, Reichel R, Bruggemann N, Hempel S, Rillig MC. Succession of arbuscular mycorrhizal fungi along a 52-year agricultural recultivation chronosequence. FEMS Microbiology Ecology. 2017;93:1–13.

49. Page AJ, Steinbiss S, Taylor B, Seemann T, Keane JA. GFF3toEMBL: Preparing annotated assemblies for submission to EMBL. The Journal of Open Source Software. 2016;1:8–9.

50. Norling M, Jareborg N, Dainat J. EMBLmyGFF3: a converter facilitating genome annotation submission to European Nucleotide Archive. BMC Research Notes. 2018;11:584.

51. Pirovano W, Boetzer M, Derks MFL, Smit S. NCBI-compliant genome submissions: Tips and tricks to save time and money. Briefings in Bioinformatics. 2017;18:179–182.

52. Wilkinson MD, Dumontier M, Aalbersberg IJ, Appleton G, Axton M, Baak A, et al. The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data. 2016;3:160018.

53. Mons B, Neylon C, Velterop J, Dumontier M, Da Silva Santos LOB, Wilkinson MD. Cloudy, increasingly FAIR: Revisiting the FAIR Data guiding principles for the European Open Science Cloud. Information Services and Use. 2017;37:49–56.