TopClassGBM
(Topic Classification Gradient Boosting Machine)

Submission for EPO-CF22

Goals

What we want to achieve.

Goals

The objective is to train a classifier for determining whether or not a patent or non-patent document belongs to the topic "green plastics".

<u>Our Goals:</u>

1) <u>High sample efficiency</u> (that is, a low number of labeled examples is required) because labeling is a time consuming and tedious process.
2) <u>Unbiased validation metrics</u> to properly estimate generalization capabilities.
3) <u>Maximize specificity at acceptable recall</u> (selectivity) because of the low prevalence of positive examples. Otherwise, false positive examples may easily outnumber the true positive examples. *

* For example, assuming that 2% of documents belong to “green plastics” and that recall and specificity of a model are 90% and 98%, the number of false positive examples (about 200) would be larger than the number of true positive examples (about 180). Such a model would not be very useful in practice.

Creativity and Innovation

How we achieve our goals and provide an accurate and robust classifier which requires only few labeled samples.

Creativity and Innovation – Conventional Approach

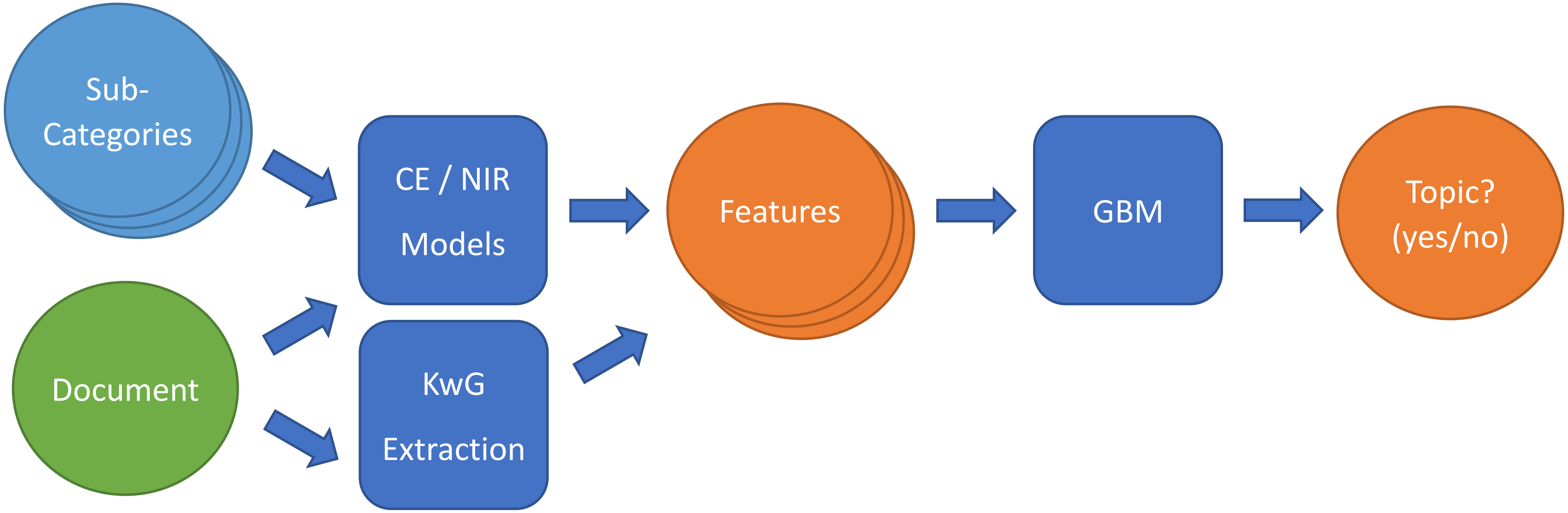
BERT-like binary classification model (CLS) takes a document as input and predicts a label (yes / no) indicating whether the document belongs to the topic.

<u>Disadvantages:</u>

- No readily available training data.
- All training data needs to be labeled and labels are highly specific to the topic.
- Any change in labeling rules requires every example to be reviewed.

Creativity and Innovation – Our Approach

- Define sub-categories of the topic and label examples as belonging to one of the subcategories or to none of the sub-categories.
- Use neural models (CE / NIR) to compute scores how well a document matches a query and use definitions of sub-categories as queries.
- Use scores and engineered features (KwG) as input for a decision-tree based gradient boosting machine (GBM) as the final binary classifier.



Creativity and Innovation – Pre-Training for Sample Efficiency

We use neural query-document-models:

- A Cross-Encoder (CE) which takes concatenated query and document as input and scores how well they match. CE is accurate but tends to overfit.
- A Neural Information Retriever (NIR) which encodes query and document separately in embedding vectors and computes the cosine similarity indicating how well they match. NIR is faster and less prone to overfitting but also less accurate.

Data for training query-document models is readily available in patent / scientific literature:

- Use title as query and abstract as document.
- Use CPC titles as query and abstract + title as document.

→ Use this data for pre-training the CE / NIR models.

→ Due to pretraining less labeled training data is required **(Goal 1)**.

Creativity and Innovation – New Fine-Tuning Method for NIR

CE is fine-tuned in the usual manner with the objective that the document matches the correct query (corresponding to the example's label) and does not match all other queries.

The NIR model computes embedding vectors of query and document separately.
→ We cannot only use similarity / dissimilarity among document and categories but also between documents as the training objective for fine-tuning.
→ Further doc-vs-doc objective: Documents in the same sub-category (including "none" category) are similar to each other but dissimilar to documents in other sub-categories.

Results of Training NIR/CE:

- Pre-training and fine-tuning are both effective in improving the performance of NIR and CE.
- Remarkably, the doc-vs-doc objective is highly effective even more than pre-training of NIR.
- Note that AUROC* is only a proxy for the model's actual performance. CE features are still superior to NIR features (see Effectiveness - Results).

NIR Training	AUROC* [%]
Our approach	**97.99**
w/o pre-training	97.83
w/o doc-vs-doc objective	97.08
w/o fine-tuning	92.53

CE Training	AUROC* [%]
Our approach	**97.39**
w/o pre-training	83.74
w/o fine-tuning	93.21

* AUROC: Area under receiver operating characteristic, a metric for judging the overall performance of a classifier (100 % is best).

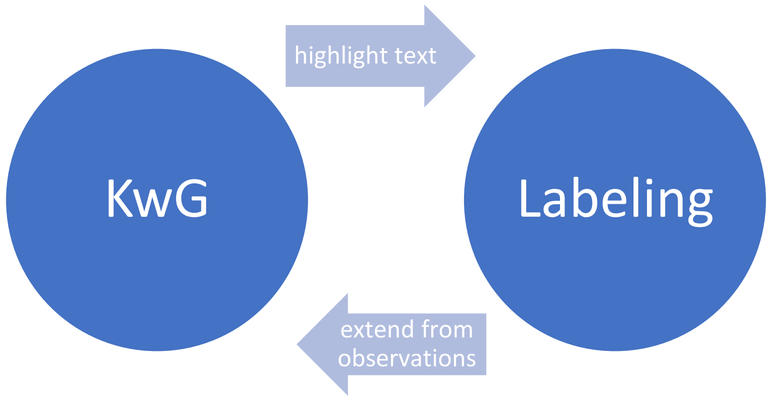
Creativity and Innovation – Keyword Group (KwG) Features

In addition to features from neural models, we use engineered features to improve robustness:

- Keyword Groups (KwG): Keyword groups contain keywords (and key phrases). Per group, the count of distinct keywords in the document is used as a feature for GBM.
- We found that distinguishing between keyword counts in title + abstract and the full-text (title + abstract + beginning of description) led to the best results.

<u>Synergy between labeling and keyword groups</u>

Defining keyword groups may appear to be extra work. But in practice, text highlighting based on the keyword groups is extremely helpful for labeling (see Design and Usability). And Keyword Groups can be extended based on recurrent terms observed during labeling.



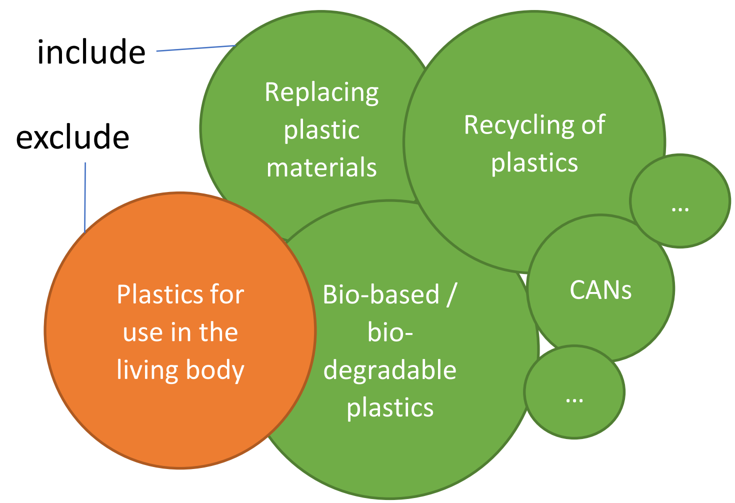
Creativity and Innovation – Further Advantages

GBMs can handle unbalanced data very well.

With GBMs, we can combine accuracy of neural models with robustness of engineered features in different feature combinations.

Examples labeled with the sub-categories are easier to maintain:

- When definition of sub-category changes, only examples of that category need to be reviewed.
- Sub-categories can be easily switched between positive and negative.
- Moreover, negative sub-categories to be excluded from the topic can be defined explicitly.



Completeness

How to find appropriate examples for training and testing so that the topic “green plastics” is actually covered.

Completeness – The Sampling Problem

Number of negative examples (not in the topic “green plastics”) in patent / non-patent literature is much larger than number of positive examples (within the topic “green plastics”).

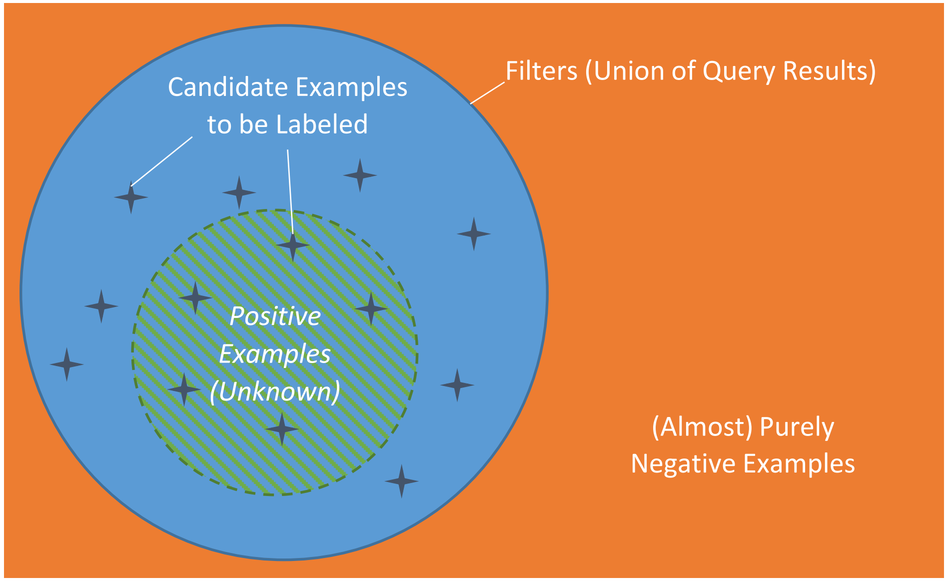
→ Naïve sampling from all documents would return only very few positive examples such that a vast number of documents would need to be labeled in order to have enough labeled positive examples.

→ Not feasible.

Completeness – Our Sampling Method

Filter:

Documents are filtered based on keywords, key phrases, and CPC codes to identify candidate examples using an Apache Lucene index over title, abstract, beginning of the description, CPC codes, and keywords if available.



The filter (union of query results) separates candidate examples for labeling which may be positive or negative from almost purely negative examples.

Higher rate of positive examples in candidate examples (40% in our data) makes labeling feasible.

Purely negative examples are also used for training (“none” category is assumed).

Completeness – Testing Generalization Capability

Goal 1 is to provide a sample efficient method in order to reduce time-consuming and tedious labeling work.

When relatively few labeled examples are used (about 1500 in our experiments), a central question is: <u>How well does the model generalize?</u>

To obtain an <u>unbiased estimate of the generalization capability</u>, we split the labeled data into three disjoined sets of approximately equal quantity:

- A training set used for training model parameters.
- An evaluation set used for tuning hyper-parameters and model selection.
- A test set <u>exclusively</u> used for computing final validation metrics.
  → Because models are trained and selected completely without the knowledge of the examples in the test set, the validation metrics are unbiased **(Goal 2)**.

Completeness – Data Sources

As the source for patent literature (published patent applications), we use:

- USPTO front-page data (title, abstract, CPC codes) from 2014 to 2022 for pre-training of CE and NIR
- USPTO full-text data of 2021 and 2022 for fine-tuning of NIR and CE and for training GBM

As the source for non-patent literature, we use:

- Articles (title, abstract, and keywords) from DOAJ (Directory of Open Access Journals) data dump for pre-training and fine-tuning of CE and NIR and for training GBM.

Why not EPO data?

- EP full-text data for text analytics contains title, abstract, and description but does not seem to contain CPC codes.
- Rate limits of OPS too low for downloading a large number of records.
- Maybe, the EPO data science team could consider to include CPC codes in future releases of the EP full-text data for text analytics?

Effectiveness

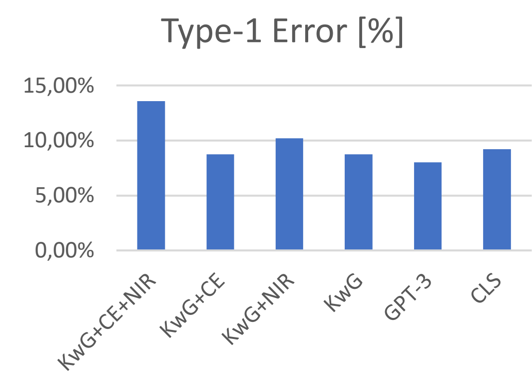
How well our GBM model with CE / NIR and KwG features performs in comparison to other approaches.

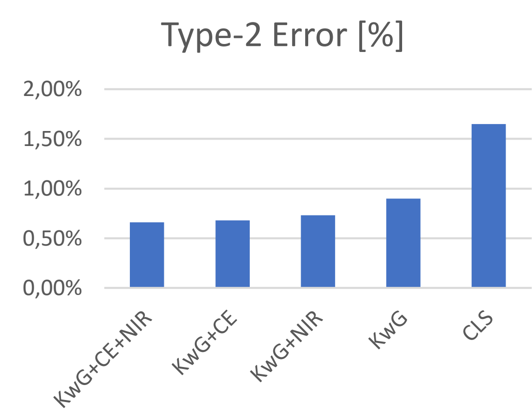
Effectiveness – Validation Results

All GBM models achieve high specificity of 99.10 to 99.34% at good recall of 86.41 to 91.26% **(Goal 3)** and out-perform conventional CLS trained on same data as well as GPT-3.

- GBM(KwG+CE) is overall winner.
- GBM(KwG+NIR) is close behind (and much faster).
- GBM(KwG+CE+NIR) seems to overfit slightly.
- GBM(KwG) still OK, may be used for pre-filtering.

	Recall [%]	Specificity [%]	Type-1 Error [%]	Type-2 Error [%]
GBM Feature Combinations				
KwG+CE+NIR	86.41	**99.34**	14.59	**0.66**
KwG+CE	**91.26**	99.32	**8.74**	0.68
KwG+NIR	89.80	99.27	10.20	0.73
KwG	**91.26**	99.10	**8.74**	0.90
Comparative Examples				
(1) CLS	90.78	98.35	9.22	1.65
(2) GPT-3	***91.98***	80.30	8.02	19.70





Effectiveness – Comparative Examples

CLS:

- Good recall, but type-2 error is more than twice as large as for KwG+CE/NIR.
- Probably, too few training examples for CLS to generalize well and no apparent way to pre-train. → More labeled data required (not sample efficient).

GPT-3:

- High recall, but poor specificity. Main Problems:
  - No good way to estimate confidence. Incorrect answers are returned with high probability.
  - Good at following positive instruction (thus the high recall), but less good at following negative instructions. → We cannot efficiently specify what not to include in a sub-category.
  - Several incorrect answers are completely unfounded making it difficult to adjust the prompt to reduce errors.
- At current state, not useful for deciding whether or not a document belongs to a topic due to poor specificity. This may change with future versions.

Efficiency

How much resources are required for labeling, training, and inference (prediction).

Efficiency – Human Work

Labeling is the most time consuming and tedious part of the work.

- About 50 to 100 records could be labeled per hour.
- Large variance was observed:
  - Some records can be labeled at first glance, other require browsing the whole document.
  - Scientific articles are usually easier to label because the abstract is more comprehensive.

Our approach requires only a low number of labeled examples (high sample efficiency, **Goal 1**), e.g. compared to direct binary classifier (CLS).

Only 1500 labeled examples were used in total for training, eval, and test sets to achieve very high specificity at high recall (KwG+CE: 99.32% at 91.26% recall).

500 training examples is quite a low number for NLP applications.

Efficiency – Compute for Training

Only moderate resources are required for training:

- Less than 20h GPU time for training in total.
- Power consumptions of approximately 7 kWh (3.5 kg CO_2 equivalents).
- Labeling can be performed in parallel to pre-training.

Task	Training Time
CE pre-training	13 h
NIR pre-training	2 h
CE fine-tuning	30 min
NIR fine-tuning	75 min
GBM training and testing	70 min

System: AMD Ryzen 7 3700X 8-Core Processor, 32 GB RAM, NVIDIA RTX 3090

Efficiency – Compute for Inference

Appropriate model for inference (prediction) can be selected based on requirements and available resources (by "mode" argument in prediction script, default is "accurate"):

- <u>Accurate mode:</u> GBM(KwG+CE) requires the most resources but is the most accurate
- <u>Balanced mode:</u> GBM(KwG+NIR) is almost as accurate and requires far less resources.
- <u>Fast mode:</u> GBM(KwG) is still surprisingly accurate and extremely fast and may be used for coarsely estimating the number of positive examples in a dataset.
- <u>Full mode:</u> GBM(KwG+CE+NIR) is inferior to GBM(KwG+CE) and is not recommended.

Features	Mode	Recall [%]	Specificity [%]	Time / 1,000 records [s]
KwG+CE+NIR	full	86.41	99.34	160
KwG+CE	accurate	91.26	99.32	155
KwG+NIR	balanced	89.80	99.27	5.6
KwG	fast	91.26	99.10	1.2

System: AMD Ryzen 7 3700X 8-Core Processor, 32 GB RAM, NVIDIA RTX 3090

Transferability

Which steps are necessary to use TopClassGBM for another topic.

Transferability –
Main Steps for a New Topic

- Define sub-categories of topic (necessary for consistent labeling anyway).
- Define filters (Lucene queries) which separate candidate examples to be labeled from (almost) purely negative examples.
- Run scripts to prepare pre-training data and candidate examples for labeling.
- Run script for pre-training of CE and NIR and (in parallel) label examples:
  - Label enough examples (each set should contain at least 200 positive examples)
  - Extend keyword groups based on observations during labeling.
  - As necessary, refine category definitions and add guidelines to improve consistency and review affected examples.
- Run script for fine-tuning of CE and NIR and training of GBM

Defining sub-categories and filter queries requires domain knowledge.
Labeling is the most time-consuming part but is made much easier by text highlighting based on keyword groups (see Design and Usability).

Design and Usability

Tools provided to the user.

Design and Usability – UI and Scripts

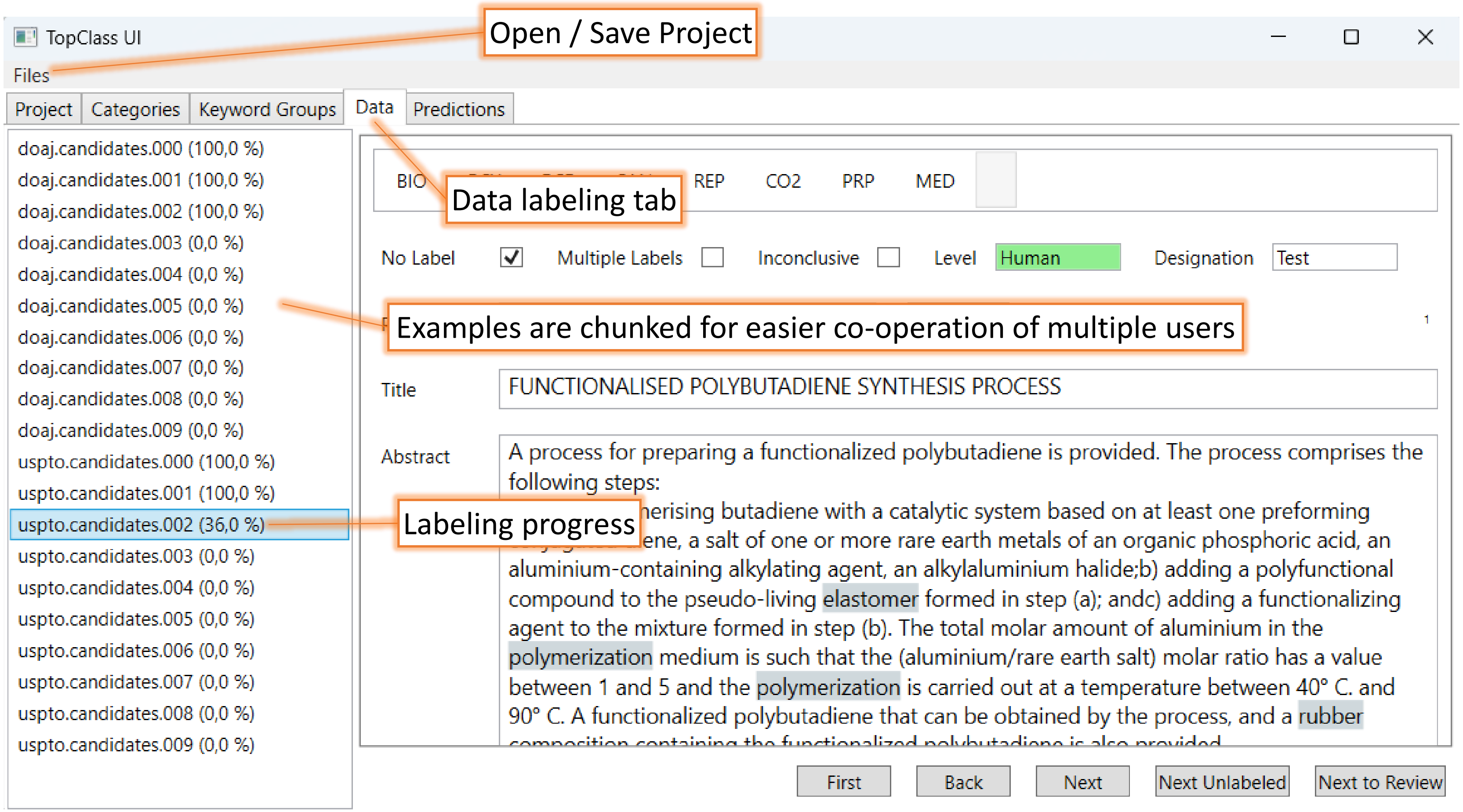
We implemented a comprehensive UI for:

- Labeling examples with text highlighting based on keyword groups
- Editing categories (labels, definitions, labeling guidelines)
- Edit keyword groups (name, guidelines, highlight color, keywords)
- Viewing statistics on labeled examples

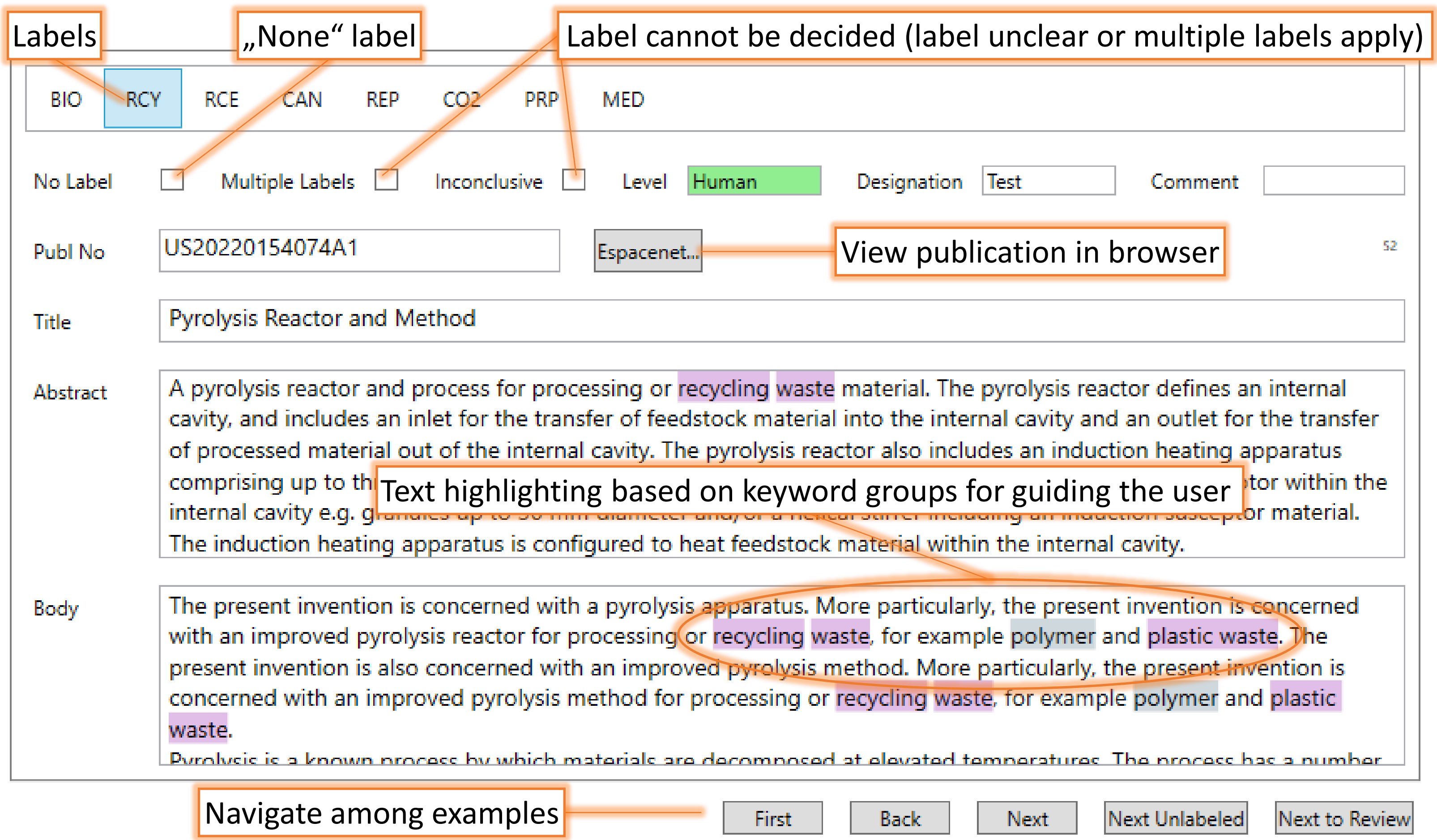
And we provide scripts for all data processing, training, and prediction tasks:

- Many parameters for detailed control
- But straightforward use due to reasonable defaults for almost all parameters

Design and Usability – UI (View Data Sets)



Design and Usability – UI (Label Examples)



Design and Usability – UI (Edit Categories)

Project | Categories | Keyword Groups | Data

Categories tab

Labels and definitions

Positive (1) or negative (-1)

Categories

Label	Definition	Type	Priority
BIO	biodegradable (compostable) plastics, bioplastics (bio-based plastics) and products made thereof, ir	1	10
RCY	recycling or reuse of plastic products, in particular recycling plastic waste into new products, feedstc	1	20
RCE	incineration (combustion) of plastic waste and using the produced energy (heat)	1	30
CAN	vitrimers and plastics from covalent adaptable polymer networks in which covalent bonds reorganiz	1	70
REP	self-repairing and self-healing plastics	1	60
CO2	methods of directly synthesizing plastics from CO2	1	40
PRP	products typically made of plastics where the plastics are (partially) replaced by non-plastic material	1	50
MED	biodegradable or biocompatible polymers used in the human or animal body for repairing tissue, su	-1	100

Press DEL to delete selected

Enter new category

Priority only used for ordering in GPT-3 prompts

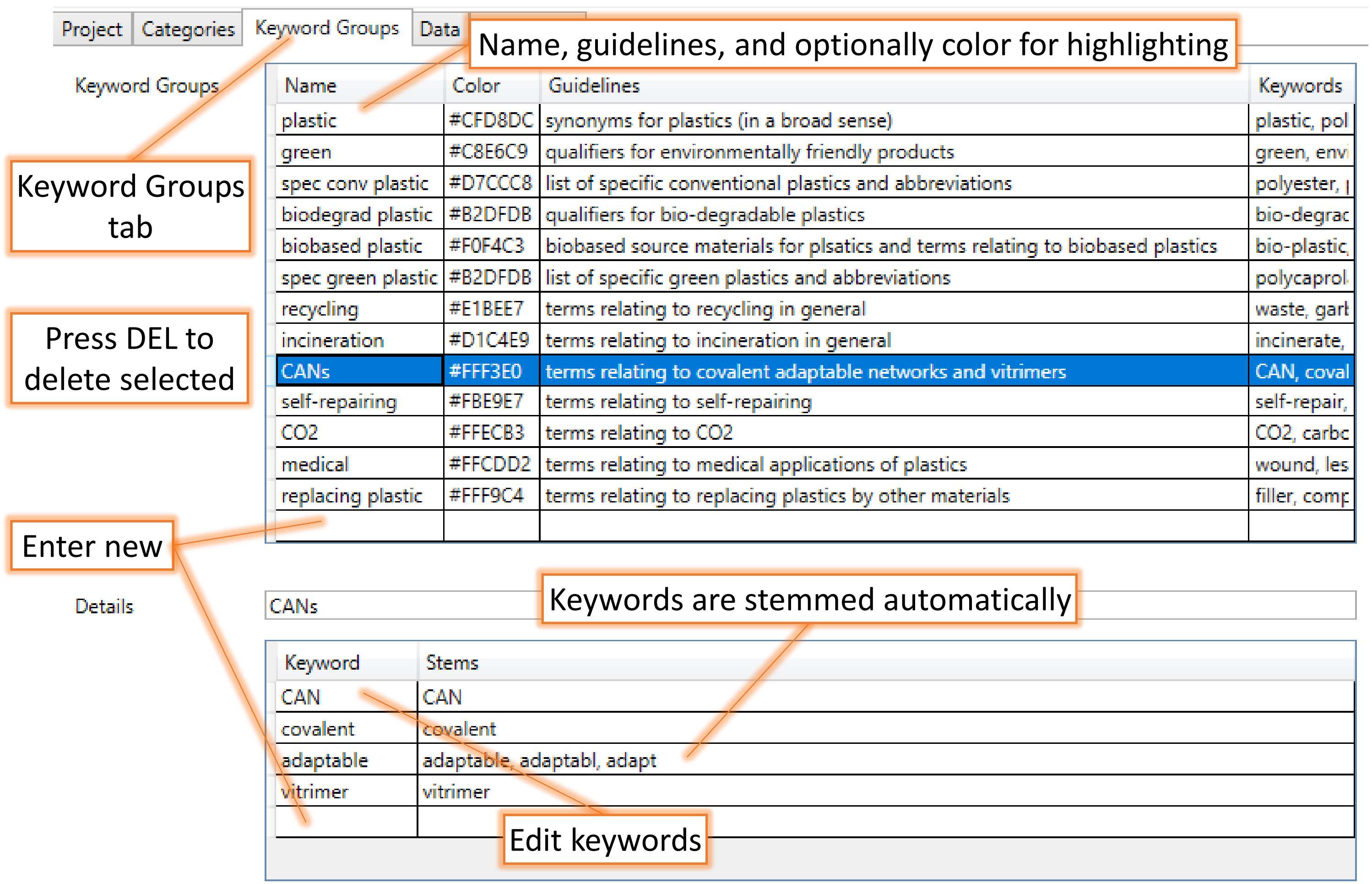
Details - CO2

Definition: methods of directly synthesizing plastics from CO2

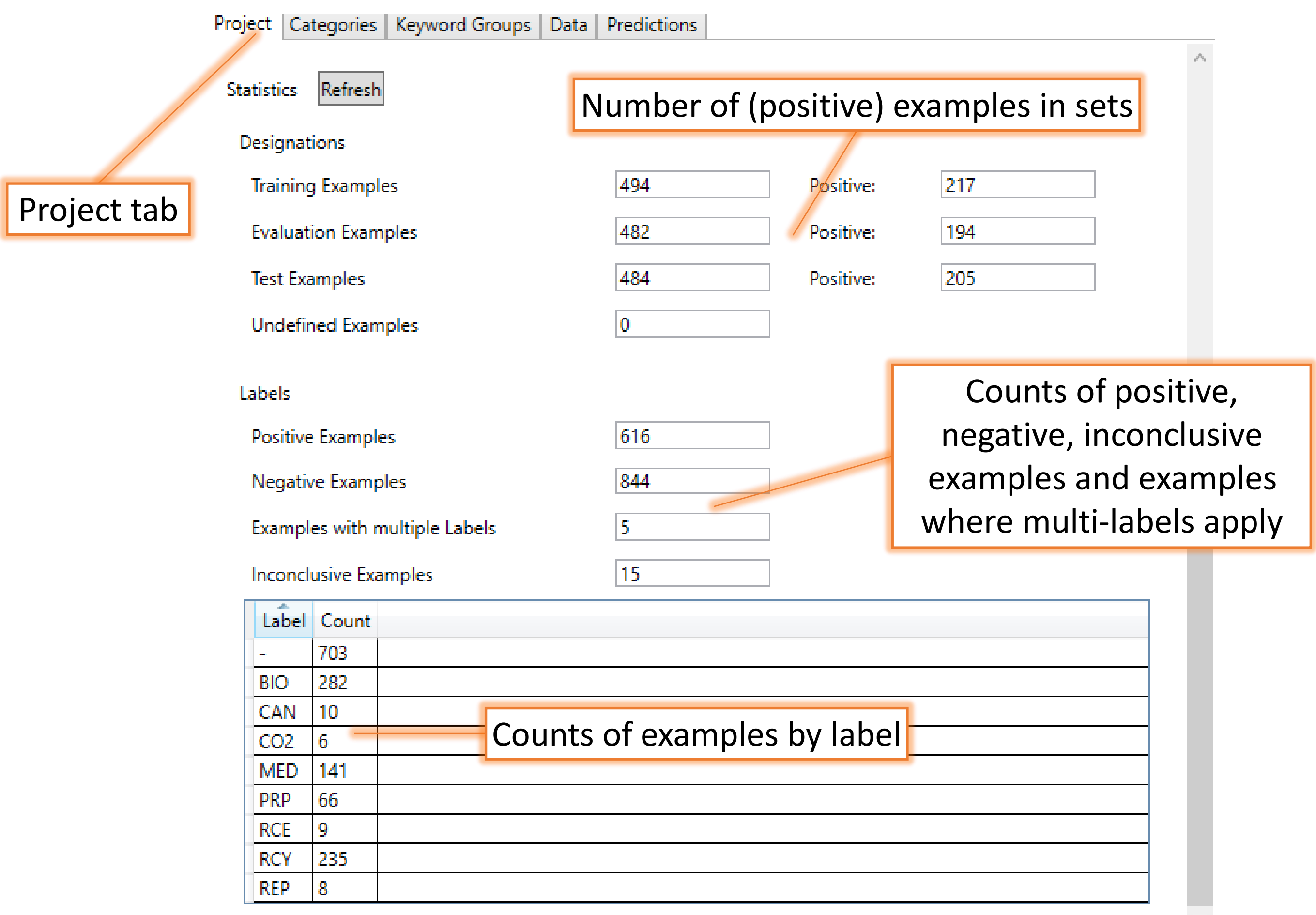
Guidelines:
Polymers used for CO2 capturing should not be included.
Foaming polymers with CO2 should not be included.
The use of polycarbonates per se should not be included.

Detailed labeling guidelines

Design and Usability – UI (Edit Keyword Groups)



Design and Usability – UI (View Statistics)



Design and Usability – Scripts (Basic Usage)

How to perform predictions (infer whether or not documents belong to the topic "green plastics"):

- In command shell run:

```
pwsh predict.ps1 -InputFile <json file> -OutputFile <json or csv file> -Mode [accurate|balanced|fast]
```

- All necessary trained models are downloaded automatically as required.
- For details on the file format, please see README.md in repository root.

How to open UI:

- In explorer double-click on "open-ui.bat".
- Last project (e.g. "green-plastics") is opened automatically

Summary

What we have achieved.

Summary

We provide a classifier for deciding whether a document belongs to a topic with:

1) <u>High sample efficiency</u> through unsupervised pre-training and new NIR fine-tuning objective (doc-vs-doc).
2) <u>Unbiased validation metrics</u> through our data sampling and splitting method.
3) <u>Very high specificity at high recall</u> by combining accurate neural models with robust engineered features using a decision-tree based GBM.

Our solution is believed to be:

- <u>Complete</u> due to data selection (patent + non-patent literature) and sampling
- <u>Transferable</u> since sub-categories and keyword groups can be adapted to any topic and since it is sample efficient such that labeling work is minimized.
- <u>Effective</u> since it achieves sufficiently high recall and specificity to be useful.
- <u>Efficient</u> since only moderate resources are required for training and inference.
- <u>Usable</u> due to UI with a straightforward design and easy-to-use scripts.
- <u>Innovative</u> due to creative use of readily available data for pre-training, our new doc-vs-doc objective for NIR fine-tuning and our GBM-based architecture.