

---

# **ASReview Documentation**

**Methodology and Statistics, Utrecht University**

**Oct 30, 2019**



<b>1</b>	<b>Quick Start</b>	<b>3</b>
<b>2</b>	<b>10 minutes into ASReview</b>	<b>5</b>
2.1	Installing the ASReview software . . . . .	5
2.2	Running Oracle mode with the PTSD dataset . . . . .	6
<b>3</b>	<b>Command Line Interface (CLI)</b>	<b>11</b>
3.1	Oracle mode . . . . .	11
3.2	Simulation mode . . . . .	12
<b>4</b>	<b>Application Programming Interface (API)</b>	<b>15</b>
4.1	Oracle mode . . . . .	15
4.2	Simulation mode . . . . .	16
<b>5</b>	<b>Models</b>	<b>17</b>
5.1	nb . . . . .	17
5.2	svc . . . . .	17
5.3	LSTM-base . . . . .	17
5.4	LSTM-pool . . . . .	17
<b>6</b>	<b>Query Strategies</b>	<b>19</b>
6.1	Random Sampling . . . . .	19
6.2	Uncertainty Sampling . . . . .	19
6.3	Max Sampling . . . . .	19
6.4	Random/Max Sampling . . . . .	19
<b>7</b>	<b>Balance Strategies</b>	<b>21</b>
7.1	Full Sampling . . . . .	21
7.2	Undersampling . . . . .	21
7.3	Triple Balance . . . . .	22
<b>8</b>	<b>Reference</b>	<b>23</b>
8.1	Low level API . . . . .	23
8.2	Models . . . . .	23
8.3	Query strategies . . . . .	23
8.4	Balance Strategies . . . . .	23
8.5	Utils . . . . .	23

<b>9 Indices and tables</b>	<b>25</b>
<b>10 Citation</b>	<b>27</b>

ASReview is software designed to accelerate the process of systematic reviews. It is written in python, and uses deep learning to predict which papers should be most likely included in the review. Our software is designed to accelerate the step of screening abstracts and titles with a minimum of papers to be read by a human with no or very few false negatives.

Automated Systematic Review (ASReview) implements an oracle and a simulation mode.

- **Oracle** The oracle modus is used to perform a systematic review with interaction by the reviewer (the ‘oracle’ in literature on active learning). The software presents papers to the reviewer, whereafter the reviewer classifies them.
- **Simulate** The simulation modus is used to measure the performance of our software on existing systematic reviews. The software shows how many papers you could have potentially skipped during the systematic review.

The source code is freely available at [GitHub](#).



# CHAPTER 1

---

## Quick Start

---

The ASR software requires Python 3.6+. The project is available on Pypi. Install the project with:

```
pip install asreview
```

The quickest way to start using the Automated Systematic Review (ASR) software is the Command Line Interface (CLI). Start an interactive systematic review (Oracle mode) with the following line in CMD or shell:

```
asreview oracle YOUR_DATA.csv --prior_included 29 181 379 2001 3928 3929 4547 \  
--prior_included 31 90 892 3898 3989 4390 --log_file results.log
```

Example output:

To benchmark an already executed review, use the simulation modus (*asreview simulation*). The dataset needs an additional column (“label\_included”) to signify their inclusion in the final review. The command for the simulation modus is similar to the oracle mode.

```
asreview simulate YOUR_DATA.csv --prior_included 29 181 379 2001 3928 3929 4547 \  
--prior_included 31 90 892 3898 3989 4390 --log_file results.log
```





---

## 10 minutes into ASReview

---

This tutorial will guide you in running the ASReview software. It consists of two parts, 0) the installation of the ASReview software and 1) *a demonstration of the Oracle mode of the ASReview software.*

### 2.1 Installing the ASReview software

Before installing the ASReview software, you have to set up the Command Line Interface and Python.

The ASReview can be run through the Command Line Interface (CLI), also known as the bash shell. Command Line Interface allows you to interact with your computer directly. The CLI is operated by text inputs (commands), rather than by a graphical interface with clickable buttons. If this is the first time you're working in the CLI, don't worry, as this tutorial will guide you through all the steps.

First, open the CLI on your computer:

- In **macOS**, you can open the CLI by launching the `Terminal` application. You can find this application in the `Utilities` folder.
- In **Windows**, the CLI is called the `Command Prompt`. However, before you can operate the CLI, you have to install `git bash`. Instructions can be found [here on OpenHash](#).
- In **Ubuntu**, you can open the CLI (called `Terminal shell prompt`) by pressing `CTRL ALT T`.

If you want to learn more about CLI, you can read [this introduction](#).

The ASReview software requires Python 3.6+. To check the Python version on your computer, type the following command in the CLI and press the return button:

```
python --version
```

This call to the CLI returns the version of the Python installation on your computer. For example:

```
Python 2.7.6
```

If your Python version is 3.6.x or higher, you can skip to *section 'Installing the ASReview software'*.

If this is not the case, as in the example above, follow the instructions in *the next section* to install a more recent version of Python.

### 2.1.1 Install Python

To install a recent version of Python, we recommend to install Anaconda. Anaconda offers a user-friendly Python environment. Besides Anaconda, there are other ways to install Python. For example, it is possible to install Python through the CLI. However, this approach requires more effort and is prone to error in novice CLI users, which is why we focus on Anaconda in this tutorial.

The Anaconda distribution is available for Windows, macOS and Linux, and can be downloaded [here](#). Users on a SolisLaptop can download the distribution from the Software Center. Two versions of the distribution are available, 2.x and 3.x. Make sure to download the 3.x distribution, this one is required for the ASReview software.

After downloading the distribution, install Anaconda by following on of the manuals below:

- For Windows users, [click here](#)
- For macOS users, [click here](#)
- For Linux users, [click here](#)

As you did before, you can now call `python --version` in the CLI, to re-check the version of the Python installation on your computer. This should correspond to the 3.x version of Anaconda you installed earlier.

Example: I installed Anaconda 3.7. Calling `python --version` in the CLI returns:

```
Python 3.7.3
```

### 2.1.2 Install ASReview

If your Python version is 3.6+, you can start installing the ASReview software on your computer. To install the ASReview software, follow the steps below. If you've already installed the ASReview software, you can skip to *the next section*.

To install the ASReview software, run the following in the Terminal:

```
pip install asreview
```

If ASReview was installed successfully, you will get output like:

```
Installing collected packages: asreview  
Successfully installed asreview-0.2.1
```

You are now ready to start your first Automated Systematic Review!

## 2.2 Running Oracle mode with the PTSD dataset

As an example, we use the PTSD dataset. This dataset contains titles and abstract of papers on post-traumatic stress disorder. From this dataset, we want to select papers relevant to our systematic review, without reading all the abstracts. To start this process, run:

```
asreview oracle example_ptsd --log_file results.log
```

The following will show on your screen:

```

  /\      /____|_____\      ( )
 /\      | (____| |____) |____      /____      /____      /____
 /\      \____\ |____ //  _\ \ / / | | _\ \ \ / \ /
 /____ \____) | | \ \ _/\ v / | | _/\ v v /
 / / \ \ \____/ | | \ \____| \ / | | \____| \ / \ /

-----
|
| Welcome to the ASReview Automated Systematic Review software.
| In this mode the computer will assist you in creating your systematic review.
| After giving it a few papers that are either included or excluded,
| it will compute a model and show progressively more relevant papers.
| You can stop the review at any time by typing "S" + Enter
| or by pressing Ctrl + C.
| Your progress will be saved.
|
| GitHub page:          https://github.com/msdslab/automated-systematic-review
| Questions/remarks:   asreview@uu.nl
|
|-----
Loading embedding matrix. This can take several minutes.

```

Note that if this is the first time you run the ASReview software, loading the embedding matrix can take a couple of minutes. In subsequent runs, loading the matrix will only take a few seconds.

### 2.2.1 Including and excluding papers beforehand

In case you already know of papers you want to in- or exclude from your systematic review in advance, you can indicate this.

The software will first ask you to indicate papers you definitely want to **include** in your systematic review.

```

Are there papers you definitively want to include?
Give the indices of these papers. Separate them with spaces.
Include:

```

You can now type the indices of papers you want to be included in your systematic review, for example 29 181 379 2001 3928 3929 4547.

In case you don't want to include any papers beforehand, just press the return key.

The software will now ask for papers you want to **exclude** from your systematic review.

```

Are there papers you definitively want to exclude?
Give the indices of these papers. Separate them with spaces.
Exclude:

```

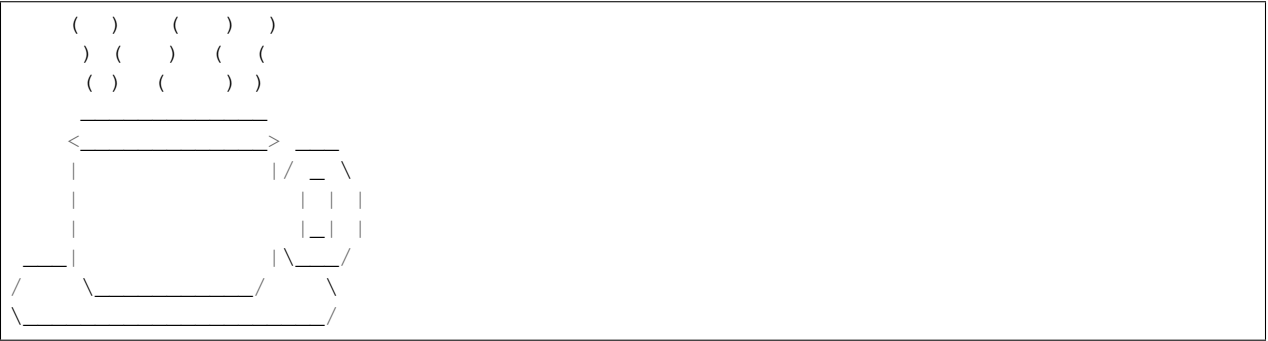
Here, type for example 31 90 892 3898 3989 4390, to indicate the papers that you want to be excluded from your systematic review.

Instead of setting up the ASReview software step by step, it is possible to run the software and indicate papers you want to in- and exclude all in one command:

```
asreview oracle example_ptsd --prior_included 29 181 379 2001 3928 3929 4547 --prior_
↳excluded 31 90 892 3898 3989 4390 --log_file results.log
```

### 2.2.2 Running the model

The software will attempt to classify the papers in the dataset into two categories: papers to be included in, and papers to be excluded from the systematic review. To improve its decisions, it will ask for your input iteratively.



At each iteration, the model will present you with a number of abstracts (20 by default). You have to let the software know whether you want to include or exclude the presented abstract in your systematic review. Indicate your decision by typing 1 (include) or 0 (exclude).

```
-----
Distributional Assumptions of Growth Mixture Models: Implications for Overextraction
↳of Latent Trajectory Classes
Bauer, Daniel J., Curran, Patrick J.

Growth mixture models are often used to determine if subgroups exist within the
↳population that follow qualitatively distinct developmental trajectories. However,
↳statistical theory developed for finite normal mixture models suggests that latent
↳trajectory classes can be estimated even in the absence of population heterogeneity
↳if the distribution of the repeated measures is nonnormal. By drawing on this
↳theory, this article demonstrates that multiple trajectory classes can be estimated
↳and appear optimal for nonnormal data even when only 1 group exists in the
↳population. Further, the within-class parameter estimates obtained from these
↳models are largely uninterpretable. Significant predictive relationships may be
↳obscured or spurious relationships identified. The implications of these results
↳for applied research are highlighted, and future directions for quantitative
↳developments are suggested. (PsycINFO Database Record (c) 2012 APA, all rights
↳reserved) (journal abstract)
-----

| 0.10% read | 0 since last inclusion | 0.00% included | total papers: 5782 |

Include [1] or exclude [0] (stop [S]):
```

When you want to quit reviewing, you can type S. You can always return to your automated systematic review later.

### 2.2.3 Under the hood

At every iteration, ASReview orders each paper by their likeliness of inclusion, with the most likely papers presented first for reading. ASReview bases its decisions on the papers you indicated prior to running the analysis, and the

subsequent decisions you make on the abstracts you get presented with during the analysis. The higher the number of included/excluded papers, the quicker the software recognizes your choices for inclusion.

The underlying technique in the software can be adapted by the user. More specifically, you can choose different [models](#), [query strategies](#), and [rebalancing strategies](#) in the ASReview software.

The options you want the software to use have to be specified in the `asreview oracle` command. If no options are specified, the ASReview software will use its defaults. Run `asreview oracle --help` to view the options and the defaults.

```
usage: asreview oracle [-h] [-m MODEL] [-q QUERY_STRATEGY]
                    [-b BALANCE_STRATEGY] [--n_instances N_INSTANCES]
                    [--n_queries N_QUERIES] [--embedding EMBEDDING_FP]
                    [--config_file CONFIG_FILE] [-s SRC_LOG_FP]
                    [--prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]]
                    [--prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]]
                    [--log_file LOG_FILE] [--save_model SAVE_MODEL_FP]
                    [--verbose VERBOSE]
                    X
```

Automated Systematic Review (ASReview) with interaction with oracle.

The oracle modus is used to perform a systematic review with interaction by the reviewer (the 'oracle' in literature on active learning). The software presents papers to the reviewer, whereafter the reviewer classifies them.

positional arguments:

X File path to the dataset or one of the built-in datasets.

optional arguments:

```
-h, --help            show this help message and exit
-m MODEL, --model MODEL
                    The prediction model for Active Learning. Default 'lstm_pool'.
-q QUERY_STRATEGY, --query_strategy QUERY_STRATEGY
                    The query strategy for Active Learning. Default 'rand_max'.
-b BALANCE_STRATEGY, --balance_strategy BALANCE_STRATEGY
                    Data rebalancing strategy mainly for RNN methods. Helps
↔against imbalanced dataset with few inclusions and many exclusions. Default 'triple_
↔balance'
--n_instances N_INSTANCES
                    Number of papers queried each query. Default 20.
--n_queries N_QUERIES
                    The number of queries. By default, the program stops after all
↔documents are reviewed or is interrupted by the user.
--embedding EMBEDDING_FP
                    File path of embedding matrix. Required for LSTM models.
--config_file CONFIG_FILE
                    Configuration file with model parameters
-s SRC_LOG_FP, --session-from-log SRC_LOG_FP
                    Continue session starting from previous log file.
--prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]
                    A list of included papers.
--prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]
                    A list of excluded papers. Optional.
--log_file LOG_FILE, -l LOG_FILE
                    Location to store the log results.
--save_model SAVE_MODEL_FP
```

(continues on next page)

(continued from previous page)

```
                Location to store the model and weights. Only works for Keras/  
↔RNN models. End file extension with '.json'.  
--verbose VERBOSE, -v VERBOSE  
                Verbosity
```

### 2.2.4 Wrapping up the Automated Systematic Review

The ASReview software will keep presenting abstracts. The longer you continue reading, the better the model will understand your review requirements, and the less likely any important papers are left not reviewed. You can stop reading abstracts at any time, by pressing S.

You can view the results by using the ‘export to csv’ option after stopping your review. Papers are ordered presenting order, those papers who are not presented are ordered by likeliness of inclusion (most likely first).

© 2019, ASReview Team, Gerbrich Ferdinands. This tutorial has been created using asreview v0.2.1 and macOS Catalina 10.15.

---

## Command Line Interface (CLI)

---

There are two modes: oracle (review) and simulation (benchmark) mode.

### 3.1 Oracle mode

Start a review process in the CMD.exe or shell.

```
asreview oracle YOUR_DATA.csv
```

The available parameters are shown with the command `asreview oracle --help`:

```
usage: asreview oracle [-h] [-m MODEL] [-q QUERY_STRATEGY]
                    [-b BALANCE_STRATEGY] [--n_instances N_INSTANCES]
                    [--n_queries N_QUERIES] [-n N_PAPERS]
                    [--embedding EMBEDDING_FP] [--config_file CONFIG_FILE]
                    [-s SRC_LOG_FP]
                    [--prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]]
                    [--prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]]
                    [--log_file LOG_FILE] [--save_model SAVE_MODEL_FP]
                    [--verbose VERBOSE]
                    X
```

Automated Systematic Review (ASReview) with interaction with oracle.

The oracle modus is used to perform a systematic review with interaction by the reviewer (the 'oracle' in literature on active learning). The software presents papers to the reviewer, whereafter the reviewer classifies them.

positional arguments:

X File path to the dataset or one of the built-in datasets.

optional arguments:

(continues on next page)

(continued from previous page)

```

-h, --help                show this help message and exit
-m MODEL, --model MODEL
                        The prediction model for Active Learning. Default 'lstm_pool'.
-q QUERY_STRATEGY, --query_strategy QUERY_STRATEGY
                        The query strategy for Active Learning. Default 'rand_max'.
-b BALANCE_STRATEGY, --balance_strategy BALANCE_STRATEGY
                        Data rebalancing strategy mainly for RNN methods. Helps
↪ against imbalanced dataset with few inclusions and many exclusions. Default 'triple_
↪ balance'
--n_instances N_INSTANCES
                        Number of papers queried each query. Default 20.
--n_queries N_QUERIES
                        The number of queries. By default, the program stops after
↪ all documents are reviewed or is interrupted by the user.
-n N_PAPERS, --n_papers N_PAPERS
                        The number of papers to be reviewed. By default, the program
↪ stops after all documents are reviewed or is interrupted by the user.
--embedding EMBEDDING_FP
                        File path of embedding matrix. Required for LSTM models.
--config_file CONFIG_FILE
                        Configuration file with model parameters
-s SRC_LOG_FP, --session-from-log SRC_LOG_FP
                        Continue session starting from previous log file.
--prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]
                        A list of included papers.
--prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]
                        A list of excluded papers. Optional.
--log_file LOG_FILE, -l LOG_FILE
                        Location to store the log results.
--save_model SAVE_MODEL_FP
                        Location to store the model and weights. Only works for Keras/
↪ RNN models. End file extension with '.json'.
--verbose VERBOSE, -v VERBOSE
                        Verbosity

```

## 3.2 Simulation mode

The CLI for the ASR software in simulation modus is similar to the CLI of the oracle modus. Instead of `asreview oracle`, use `asreview simulate`.

```
asreview simulate YOUR_DATA.csv
```

The available parameters are:

```

usage: asreview simulate [-h] [-m MODEL] [-q QUERY_STRATEGY]
                        [-b BALANCE_STRATEGY] [--n_instances N_INSTANCES]
                        [--n_queries N_QUERIES] [-n N_PAPERS]
                        [--embedding EMBEDDING_FP]
                        [--config_file CONFIG_FILE] [-s SRC_LOG_FP]
                        [--prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]]
                        [--prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]]
                        [--n_prior_included N_PRIOR_INCLUDED]
                        [--n_prior_excluded N_PRIOR_EXCLUDED]
                        [--log_file LOG_FILE] [--save_model SAVE_MODEL_FP]

```

(continues on next page)



(continued from previous page)

```

        [--verbose VERBOSE]
        X
Automated Systematic Review (ASReview) for simulation runs.

The simulation modus is used to measure the performance of our
software on existing systematic reviews. The software shows how many
papers you could have potentially skipped during the systematic
review.

positional arguments:
  X                File path to the dataset or one of the built-in datasets.

optional arguments:
  -h, --help            show this help message and exit
  -m MODEL, --model MODEL
                        The prediction model for Active Learning. Default 'lstm_pool'.
  -q QUERY_STRATEGY, --query_strategy QUERY_STRATEGY
                        The query strategy for Active Learning. Default 'rand_max'.
  -b BALANCE_STRATEGY, --balance_strategy BALANCE_STRATEGY
                        Data rebalancing strategy mainly for RNN methods. Helps
↪against imbalanced dataset with few inclusions and many exclusions. Default 'triple_
↪balance'
  --n_instances N_INSTANCES
                        Number of papers queried each query.Default 20.
  --n_queries N_QUERIES
                        The number of queries. By default, the program stops after
↪all documents are reviewed or is interrupted by the user.
  -n N_PAPERS, --n_papers N_PAPERS
                        The number of papers to be reviewed. By default, the program
↪stops after all documents are reviewed or is interrupted by the user.
  --embedding EMBEDDING_FP
                        File path of embedding matrix. Required for LSTM models.
  --config_file CONFIG_FILE
                        Configuration file with model parameters
  -s SRC_LOG_FP, --session-from-log SRC_LOG_FP
                        Continue session starting from previous log file.
  --prior_included [PRIOR_INCLUDED [PRIOR_INCLUDED ...]]
                        A list of included papers.
  --prior_excluded [PRIOR_EXCLUDED [PRIOR_EXCLUDED ...]]
                        A list of excluded papers. Optional.
  --n_prior_included N_PRIOR_INCLUDED
                        Sample n prior included papers. Only used when --prior_
↪included is not given. Default 10
  --n_prior_excluded N_PRIOR_EXCLUDED
                        Sample n prior excluded papers. Only used when --prior_
↪excluded is not given. Default 10
  --log_file LOG_FILE, -l LOG_FILE
                        Location to store the log results.
  --save_model SAVE_MODEL_FP
                        Location to store the model and weights. Only works for Keras/
↪RNN models. End file extension with '.json'.
  --verbose VERBOSE, -v VERBOSE
                        Verbosity

```



---

## Application Programming Interface (API)

---

For more control over the workings of the ASReview software, a API is provided. For example, it is possible to define a new model or sampling strategy and use it with ASReview.

There are two modes: oracle (review) and simulation (benchmark).

### 4.1 Oracle mode

An example that uses the ASReview API (fill in DATA\_FILE and EMBEDDING\_FILE with valid filenames):

```
import asreview
from asreview.models import create_lstm_pool_model

from tensorflow.python.keras.wrappers.scikit_learn import KerasClassifier

# load data
data, texts, _ = asreview.read_data(DATA_FILE)

# create features and labels
X, word_index = asreview.text_to_features(texts)

# Load embedding layer.
embedding = asreview.load_embedding(EMBEDDING_FILE, word_index=word_index)
embedding_matrix = asreview.sample_embedding(embedding, word_index)

# create the model
model = KerasClassifier(
    create_lstm_pool_model(embedding_matrix=embedding_matrix),
    verbose=1,
)

# start the review process.
reviewer = asreview.ReviewOracle(
```

(continues on next page)

```
X,  
data=data,  
model=model,  
n_instances=10,  
prior_included=PRIOR_INC_LIST, # List of some included papers  
prior_excluded=PRIOR_EXC_LIST, # List of some excluded papers  
)  
reviewer.review()
```

## 4.2 Simulation mode

An example of use the API for the simulation mode:

```
import asreview  
from asreview.models import create_lstm_pool_model  
  
from tensorflow.python.keras.wrappers.scikit_learn import KerasClassifier  
  
# load data  
_, texts, y = asreview.read_data(DATA_FILE)  
  
# create features and labels  
X, word_index = asreview.text_to_features(texts)  
  
# Load embedding layer.  
embedding = asreview.load_embedding(EMBEDDING_FILE, word_index=word_index)  
embedding_matrix = asreview.sample_embedding(embedding, word_index)  
  
# create the model  
model = KerasClassifier(  
    create_lstm_pool_model(embedding_matrix=embedding_matrix),  
    verbose=1,  
)  
  
# start the review process.  
reviewer = asreview.ReviewSimulate(  
    X,  
    y=y,  
    model=model,  
    n_instances=10,  
    prior_included=PRIOR_INC_LIST, # List of some included papers  
    prior_excluded=PRIOR_EXC_LIST, # List of some excluded papers  
)  
reviewer.review()
```

There are several models implemented currently. The best performing is at the moment the `lstm_pool` algorithm.

### 5.1 nb

See `asreview.models.create_nb_model()`

Naive Bayes model, default parameters.

### 5.2 svc

See `asreview.models.create_svc_model()`

Support Vector Machine algorithm. Unoptimized.

### 5.3 LSTM-base

See `asreview.models.create_lstm_base_model()`

LSTM model that consists of an embedding layer, LSTM layer with one output, dense layer, single sigmoid output node.

### 5.4 LSTM-pool

See `asreview.models.create_lstm_pool_model()`

LSTM model that consists of an embedding layer, LSTM layer with many outputs, max pooling layer, single sigmoid output node.



There are a few query strategies available, and depending on the needs of the simulation/review some will work better than others.

Parameters should be under the section [query\_param].

### 6.1 Random Sampling

See `asreview.query_strategies.random_sampling()`

As it says: randomly select samples with no regard to model assigned probabilities.

### 6.2 Uncertainty Sampling

See `asreview.query_strategies.uncertainty_sampling()`

Choose the most uncertain samples according to the model (i.e. closest to 0.5 probability). Probably doesn't work very well in the case of LSTM's, since the probabilities are rather arbitrary.

### 6.3 Max Sampling

See `asreview.query_strategies.max_sampling()`

Choose the most likely samples to be included according to the model.

### 6.4 Random/Max Sampling

See `asreview.query_strategies.rand_max_sampling()`

Use a combination of random and max sampling. By default it does 5% random sampling and 95% max sampling. Works well in combination with the triple balance strategy. This parameter can be set in the configuration file:

```
# Set to 5% random, 95% max sampling.  
rand_max_frac=0.05
```



---

## Balance Strategies

---

There are several balance strategies that rebalance and reorder the training data. This is sometimes necessary, because the data is often very imbalanced: there are many more papers that should be excluded than included (otherwise, automation cannot help much anyway).

Parameters in the config file should be under the section `[balance_param]`.

We have currently implemented the following balance strategies:

### 7.1 Full Sampling

See `asreview.balance_strategies.full_sample()`

This just uses all the data.

### 7.2 Undersampling

See `asreview.balance_strategies.undersample()`

This undersamples the data, leaving out excluded papers so that the included and excluded papers are in some particular ratio (closer to one). Configuration options are as follows:

```
# Shuffle the samples
shuffle=True

# Set the ratio of included/excluded to 1
ratio=1.0
```

## 7.3 Triple Balance

See `asreview.balance_strategies.triple_balance()`

This divides the training data into three sets: included papers, excluded papers found with random sampling and papers found with max sampling. They are balanced according to formula's depending on the percentage of papers read in the dataset, the number of papers with random/max sampling etc. Works best for stochastic training algorithms.

```
# Shuffle the samples
shuffle=True

# Start with a random/max weight ratio of 10 at 0% read papers.
rand_max_b=10

# Decay to 1 with the following decay exponent:
rand_max_alpha=1.0

# Decrease the included/excluded ratio with the power:
one_zero_beta=0.6

# Cap the included/excluded ratio at:
one_zero_delta=0.16
```

**8.1 Low level API**

**8.2 Models**

**8.3 Query strategies**

**8.4 Balance Strategies**

**8.5 Utils**



## CHAPTER 9

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



## CHAPTER 10

---

### Citation

---

A research paper is upcoming for this project. In the mean time, it can be cited with (fill in x and y for the version number):

ASReview Core Development Team (2019). ASReview: Software for automated systematic reviews [version 0.x.y]. Utrecht University, Utrecht, The Netherlands. Available at <https://github.com/msdslab/automated-systematic-review>.

```
@Manual{
  title = {ASReview: Software for automated systematic reviews},
  author = {{ASReview Core Development Team}},
  organization = {Utrecht University},
  address = {Utrecht, The Netherlands},
  year = 2019,
  url = {https://pypi.org/project/asreview/}
}
```