Данные для обучения и тестирования алгоритмов доступны по адресу:
https://cloud.mail.ru/public/9XXY/WAfXAWLnW

Описание данных по папкам:
1. source_retrieval – данные для дорожки поиска источников заимствований (дорожка 1)
    src – источники возможных заимствований
    susp – тексты с заимствованиями из источников
    tasks – файлы с информацией, сопоставляющей тексты с заимствованиями и источники заимствований

2. text_alignment – данные для дорожек поиска заимствований в тексте (дорожки 2 и 3)
  src – источники заимствований
  susp – тексты с заимствованиями из источников
  tasks – файлы с информацией, сопоставляющей заимствованные фрагменты текстов из susp с фрагментами из источников sources

Задания сгруппированы в архивы по типу заимствований, например, generated_paraphrased означает данные для автоматически сгенерированных парафразированных заимствований. Там же доступна методика, по которой создавались парафразированные тексты, включенные в коллекцию в качестве заданий.

Инструкция по использованию данных для дорожки 1
Папка src содержит множество источников, из которых мог быть позаимствован текст.
Папка susp содержит документы с заимствованиями.
Документы разделены по типу заимствований. Так, в архиве academic_plagiarism находятся реальные заимствования, выявленные в российском научном сообществе. В архиве manually_paraphrased находятся эссе, написанные студентами на заданную тему, с большим количеством заимствований из разных источников. В архивах generated_copypast и generated_paraphrased находятся тексты с автоматически сгенерированными дословными и перефразированными заимствованиями.
При написании эссе студенты использовали техники сокрытия (парафраз). Подробнее методика составления эссе описана в документе "Создание текстов с заимствованиями", который распространяется вместе с обучающей выборкой.
В папке tasks в архивах находятся файлы, описывающие источники заимствований для каждого текста из папки susp.

Программа обнаружения источников заимствований должна выдавать json-файл с именем XYZ.json, который содержит мета-информацию об обнаруженных источниках. Пример:
```
{
    "suspicious-document": "XYZ.txt"
    "detected-plagiarism": [
        {
            "id": "5216589"
        },
        {
            "id": "323433"
        },
        {
            "id": "3838483"
        }
    ]
}
```
В примере выше для подозрительного документа XYZ.txt было найдено три источника заимствований, идентификаторы источников указываются в поле id.

Документы должны быть отсортированы по объему заимствований в символах, словах или фрагментах, т.е. документ, из которого было больше всего заимствовано текста должен находится на первом месте.

Для оценки качества обнаружения источников будут использоваться макро-усредненные точность, полнота, F-мера, средняя точность (average precision).
Для оценки качества во время обучения можно использовать скрипт:
https://cloud.mail.ru/public/9XXY/WAfXAWLnW/source_retrieval/source_retrieval_measures.py
Пример запуска:
$ python source_retrieval_measures.py -p tasks/manually-paraphrased/ -d manually-paraphrased-result/

После этапа обучения участникам будет выдан тестовый набор подозрительных документов. Для каждого документа из этого набора нужно будет выполнить поиск источников и сгенерировать мета-информацию в указанном выше формате с ранжированием источников.

Инструкция для использования данных для дорожек 2 и 3
В папке text_alignment/tasks находятся файлы с заданиями, описывающими заимствованные фрагменты. Задания сгруппированы в архивах и разделяются по типу заимствований.

Для каждого типа заданий есть файл pairs. Этот файл перечисляет все пары подозрительных документов и источников, которые нужно сравнить друг с другом. Первая колонка в файле указывает на подозрительный документ (сам файл находится в директории susp), вторая колонка указывает на источник (файл находится в директории src).

Программа обнаружения заимствований должна сгенерировать XML-файл suspicious-documentXYZ-source-documentABC.xml, который содержит метаинформацию об обнаруженных заимствованиях. Пример:

```
<document reference="XYZ.txt">
<feature
  name="detected-plagiarism"
  this_offset="5"
  this_length="200"
  source_reference="ABC.txt"
  source_offset="100"
  source_length="150"
/>
<feature ... />
...
</document>
```

В примере выше заимствованный текст в документе XYZ.txt начинается с 5-ого символа и имеет длину 200 символов. В источнике ABC.txt текст, который был заимствован, начинается с 100-ого символа и имеет длину 150 символов.
В качестве базового метода (baseline) можно использовать программу http://www.uni-weimar.de/medien/webis/events/pan-12/pan12-code/pan12-text-alignment-baseline.py

Пример запуска:
$ python pan12-text-alignment-baseline.py tasks/manually-paraphrased/pairs src susp manually-paraphrased-result
С результатами базового метода можно сравнивать результаты своих методов.

Для оценки качества обнаружения заимствований будут использоваться макро-усредненные точность, полнота и др. Подробнее прочитать про использованные метрики можно по ссылке: http://www.uni-weimar.de/medien/webis/publications/papers/stein_2010p.pdf#page=2

Для оценки качества во время обучения можно использовать скрипт http://www.uni-weimar.de/medien/webis/events/pan-09/pan09-code/pan09-plagiarism-detection-performance-measures.py

Пример запуска:
$ python pan09-plagiarism-detection-performance-measures.py -p tasks/manually-paraphrased/ -d manually-paraphrased-result/

На этапе оценки прогонов участники должны будут сдать свои программы (скрипты), которые будут автоматически запускаться на платформе TIRA на закрытом множестве контрольных заданий. Программы будут запускаться следующим образом:
mySoftware -i path/to/corpus -o path/to/output/directory
На платформе TIRA участникам будет выделена персональная виртуальная машина, с одной из следующих ОС: Windows 7 или Ubuntu 12.04. Можно использовать любой язык программирования. Доступ к виртуальной машине будет организован через ssh или rdp. Детальная информация о работе с ВМ http://pan.webis.de/clef14/pan14-web/pan14-virtual-machine-user-guide.pdf