Лабораторная 3.

Так интересно ещё не было никогда.

В лабораторной работе мы напишем свою реализацию решения задачи тематического моделирования на основе модели PLSA (probabilistic latent semantic analysis) , a.k.a. PLSI (Probabilistic latent semantic indexing). Почему с помощью неё, а не LDA? Причин для этого несколько:

1. Он проще в реализации;
2. Он работает быстрее.

Да, у PLSA есть проблемы с точностью, но не переживайте, до LDA мы обязательно дойдём.

Что делать:

1. Вспомнить:
   - Лекции по тематическому моделированию.

2. Если требуется, то почитать:
   - Краткая обзорная статья
   - Пособие по ТМ
   - Здесь дополнительно можно почитать про EM-алгоритмы
   - Здесь всё понятно и наглядно

3. Определиться с архитектурой, приступить к написанию компонентов. Например, можно выделить следующие части:
   - Класс документа
   - Класс корпуса документов
   - Препроцессинг данных (удаление стоп-слов и знаков препинания, лемматизация или стемминг и т.д; хорошо подумайте где лучше его разместить)
   - Класс самой модели (хранит в себе гиперпараметры, напр.: количество тем, максимальное количество итераций и т.д.)
   - Класс EM-алгоритма (лучше вынести отдельно)

4. Какие библиотеки можно использовать:
   - Стандартную библиотеку (ввод-вывод, PRNG, math, …)
   - pymystem3 или nltk.stem.

5. Провести тестирование на данных из data/lenta_ru.csv.
   - Достаточно запустить на 10-15 первых статьях из 100. Дальше уже ориентируйтесь относительно доступных мощностей;
   - Поэкспериментируйте с количеством тем (ориентируйтесь на те, что указаны в датасете);
   - Вывести топ слов для каждой темы;
   - Вывести распределение документов по темам.

6. Требования и регламент
   - При написании руководствоваться принципами ООП(!!!)
   - Оформлять код в соответствии с PEP8.

- **Код оформляется в виде модуля в пакете из предыдущей лабораторной работы**, а также клиентской части использующей этот код. Небольшой совет: клиентский код для прошлой и этой лабораторной работы можно поместить в папку examples.
- Презентовать: код, результат работы.
- Сдает один человек.
- Сроки указаны в https://tinyurl.com/DA2018-MO151

7. Дополнительно (**+1 балл**):
    - Использовать систему контроля версий git и модель ветвления gitflow (или её производные: gitflow light, gitlab flow и т.д.).
    - Разместить на github.com или gitlab.com.

8. Дополнительное “дополнительно” (**+1 балл** при сдаче своевременно):
    - Кроме EM-алгоритма реализован стохастический EM-алгоритм

9. Контакты:
    - mail: ivan@sha.run
    - tg: @zaryanezrya