Рогович Татьяна

Анализ данных в Python (политология)

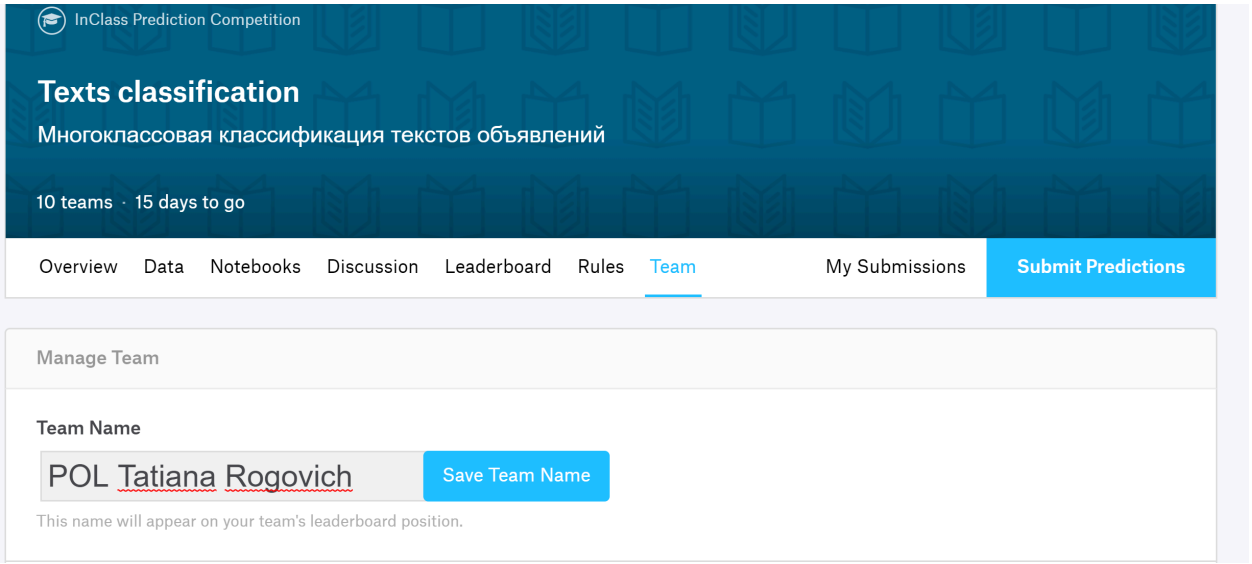
Дедлайн: 9 декабря 23.59
Поздний дедлайн: 12 декабря 23.59 (12 декабря закрывается соревнование, вы не сможете больше отправлять решения, до 10 декабря 23.59 штраф один балл, до 12 декабря 23.59 штраф 2 балла)

Домашнее задание 2

Участвуем в соревновании на kaggle.
https://www.kaggle.com/t/78b4150e88c84935b37e550fbb636a4f
По ссылке можно скачать данные и baseline решение (базовое решение, которое будем стараться улучшать).

Чтобы выполнить это ДЗ физически нужно время, потому что данные немаленькие и считаться это будет не быстро. Поэтому каждый эксперимент это 1-3 часа (вы естественно можете заниматься своими делами, но лучше следить, что там происходит).

Что понадобится: аккаунт kaggle, акканут Google и 3,5 гб свободного места на гугл диске.
Аккаунт kaggle может быть создан с любой почты, логин любой. Но когда вы будете участвовать в соревновании, вам нужно будет во вкладке Team указать префикс POL и полностью имя и фамилию.



Оценивание

Макс. 11 баллов

Если вы делаете любой пункт, кроме 1, вы кратко комментируете свои действия в блокноте и сдаете в лмс на проверку блокнот (все шаги должны быть в нем зафиксированы).

Перед началом работы, из вкладки data на kaggle вам нужно скачать файлы train и test и положить их в папку на своем гугл-диске. Дальше загрузить их по инструкции в видео: https://www.youtube.com/watch?v=DKGmFZgR8K8&feature=youtu.be

1. **4 балла**

Вы запускаете блокнот на базовое предсказание, генерируете файл submission и отправляете его на kaggle с названием baseline submission.
Если вы делаете только эту часть, вы не отправляете мне блокнот (так как ничего не правите), но на экзамене я спрошу вас по одному из шагов в этом блокноте и попрошу объяснить, что именно мы делаем (мы все это будем разбирать на занятиях).

2. **2 балла**

- Вы очищаете текст от стоп-слов, чисел и пунктуации, применяете стемминг (это нужно сделать внутри собственной функции).
- Дальше делаете все то же самое, что в блокноте. Не забудьте применить функцию по очистке к Train и Test.
  Рекомендую сохранить файлы train и test с очищенным текстом в csv у себя на диске. Этот шаг занимает много времени и потом вам будет проще запускать модель без него (если что-то сорвалось или вы делаете задание в несколько подходов).
- Напишите, улучшилось ли качество предсказания на тестовой выборке внутри блокнота по сравнению с базовым блокнотом.
- Отправляете на kaggle submission с название baseline clean text submission.
- После сабмита напишите в блокноте, улучшилось ли качество предсказания по сравнению с базовым блокноте на тестовой выборке kaggle.

3. **1 балл**

- Если вы делали пункт 2, то к очищенному тексту применяете tf idf vectorizer, но с такими настройками, чтобы он не учитывал слова, которые встречаются очень часто во всех документах (стоп-слова для корпуса). Оставьте только 100 000 признаков. Посмотрите в документации на параметры max_df и max_features.
  https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
- Дальше делаете то, что в блокноте. Напишите, улучшилось ли качество предсказания на тестовой выборке внутри блокнота по сравнению с базовым блокнотом
- Отправляете на kaggle submission с название tf idf max submission.
- После сабмита напишите в блокноте, улучшилось ли качество предсказания по сравнению с базовым блокноте на тестовой выборке kaggle.

4. **1 балл**

- Если делали пункт 2 и 3, то к tf idf матрице, которая у вас уже есть (после пункта 3), применяете классификатор MultinomialNB из коробки и считаете метрику ошибки для тестовой выборки в блокноте.

- Реализуете подбор параметра сглаживания alpha по сетке для значений 0.01, 0.05, 0.1, 0.5, 1. Для каждой модели считаете метрику ошибки на отложенной тестовой выборке внутри блокнота.
- Выбираете параметр alpha, который дает лучший результат. Обучаете MultinomialNB с этим параметром на всей выборке.
- Отправляете на kaggle submission с названием naive bias submission.
- После сабмита напишите в блокноте, улучшилось ли качество предсказания по сравнению с базовым блокноте на тестовой выборке kaggle.

5. **2 балла**

- К очищенному тексту применяете CountVectorizer с ограничением 10000 признаков и n-граммами от 1 до 2 (включаем только отдельно стоящие слова и пары слов).
- Как модель для предсказания используете MultinomialNB и опять реализуете подбор параметра сглаживания alpha по сетке для значений 0.01, 0.05, 0.1, 0.5, 1.
- Напишите, улучшилось ли качество предсказания на тестовой выборке внутри блокнота по сравнению с базовым блокнотом и вашими предыдущими экспериментами.
- Отправляете на kaggle submission с названием naive bias submission.
- После сабмита напишите в блокноте, улучшилось ли качество предсказания по сравнению с базовым блокноте на тестовой выборке kaggle.

Дополнительные баллы

Если после 12 декабря окажется, что ваша модель пробила второй baseline, то прибавляем 1 балл к итоговой оценке. Не забудьте на странице My Submissions поставить галочку Use for Final Score модель с самой высокой оценкой на части тестовых данных.

Вы можете делать дополнительные эксперименты, не описанные в этом блокноте.

baseline2 - 10 баллов

Что присылаем:

1. В лмс загружаете скачанный блокнот, в котором зафиксированы все ваши шаги с короткими комментариями (перед каждым шагом пишете, что вы делаете).
2. Также в лмс загружаете скриншот вашей страницы My Submissions, на котором видны все ваши submissions и оценка за них.
3. Пункт засчитывается выполненным только при наличии блокнота с этими шагами + скриншота с нужным сабмитом. Если делаете только первый пункт на 4 балла - присылаете только скриншот.