Часть 1.

1.1. Были выделны 1000 самых встречаемых слов (без учета стоп-слов). - зачастую ими оказывались артикли и существительные: the, to, and и прочие.

1.2. Регулярными выражениями были токенизированы слова и переведены к нижнему регистру и удалены стоп-слова:

1. выделены топ-10 по именам
2. топ - 10 по упоминаниям персонажей (first name + second name)
3. топ - 10 по упоминаниям лиц профессорского состава.

Часть 2.

2.1. Мы продемонстрировали как работает поиск синонимов, ассоциация, и лишних слов в обучении модели, использовав модель FastText с подобранными гиперапараметрами и находит наилучшее сочетание пар.

2.2. Построен FastText с подбором гиперпараметров и с помощью модели FastText T-SNE визуализировано топ - 1000 слов.

2.2. Построен FastText с подбором гиперпараметров и с помощью модели FastText T-SNE визуализировано топ 1000 слов.

Часть 3.

3.1. Построен baseline-классификатор на FastText с учетом подбора параметров.

Прогноз получился нормальным, если не учитывать что при усреднении ф-меры по лэйблам без взвешивания по количеству объектов точность кажется ниже, чем у случайного угадывания.

Выбрали рандомно часть данных, и случайно применили к каждому отобранных текстов одну из 3 функций: удаление слов с вероятнотью p, замена n слов на синонимы, добавление m синонимов для n слов.

Далее размножили те заклинания, которые меньше 1000 в

4 раза, оставив самые популярные неименными, По итогу макро ф1 выросло на 38%, но микро сократилось.

3.2.

Построена модель нейронной сети CNN (CustomCNN), оптимизатором выступил Adam, а критерием - критерий CrossEntropyLoss. Модель обучилась на 15 эпохах. Значение точности f1 меры составило 0.120 на тестовой выборке.