Очистка данных и обработка текста новостей

На данном этапе нам необходимо было подготовить данные для последующей кодировки (т.к. работаем с текстовыми данными).

Этап состоял их трех смысловых частей:

**1. Парсинг новых данных с сайта Lenta.ru**

В изначальном датасете имелись 2 проблемы, которые представляли помеху для дальнейших моделей:

* отсутствия данных за декабрь 2018г
* полное отсутствие данных за 2019 -2022года.

Ноутбук с кодом парсинга модно найти на гитхабе.

**2. Очистка датасета и обработка текста новостей**

Поскольку наши данные представлены в виде связного текста, нам необходимо преобразовать их в понятный для машины язык - набор векторов. Основная идея обработки текстов сводится преобразованию их в массивы слов, стоящих в начальной форме.

Производилась следующая очистка данных: удаление дубликатов строк в датасете и пропущенных строк в колонке “text”

Подготовка данных для последующей кодировки была произведена двумя путями (оба ноутбука приложены). Основное отличие подходов заключалось в разных принципах работы лемматизаторов (библиотеки pymystem3 и pymorphy2).

Выбранные библиотеки были разработаны для обработки текстов русского языка. На данном этапе отпала всемиизвестная библиотека nltk. По визуальному анализу токенизированных текстов было видно, что она работает на порядок хуже.

Отличие между этими библиотеками состоит в том, что pymystem3 токенизирует и лемматизирует слова с учетом контекста слова, что является большим преимуществом для русского языка. А библитека pymorphy2 лемматизиует слова уже после токенизации. Ее приятная черта в том, что можно настроить лемматизацию вручную.

Логика обработки библиотекой pymorphy2

Как уже было сказано ранее, библиотека pymorphy2 обрабатывает уже токенизированные слова. В следствии этого встает необходимость использовать токенизатор. Здесь был выбор из двух библиотек: классическая nltk и русская natasha. Проводилось тестирование работы обоих библиотек на небольшой части датасета и применением после визуального анализа результатов. Практически сразу было видно явное преимущество по качеству библиотеки natasha.

Примером может послужить следующая фраза: «битвы…Величие». Фраза взята из одной новости. К сожалению, подобные опечатки довольно часто встречались в датасете. Библиотека nltk не распознавала из и после токенизации они так и оставались «слипшимися». Однако Natasha спокойно с этим справлялась и выдавала «битвы» и «Величие».

После применения токенизатора от natasha схема обработки текста была достаточно стандартной:

* удаление лишних символов (по типу !"#$%&\'()\*+,-./:;<=>?@[\\]^\_`{|}~\n«»\t—…–№)
* удаление стоп-слов (словарь стоп-слов был взят из библиотеки nltk)
* лемматизация с помощью библиотеки pymorphy2

Отдельное приведение в нижний регистр слов отсутствовало, поскольку Natasha при токенизации автоматически переводит слова в нижний регистр

Еще одно из проблем выступила проблема «слипшихся слов». Например, многие новости имели такого вида слова: «пустыниСражались». То есть конец и начало следующего предложения оказывались слипшимися.

Для решения данной проблемы было решено использовать YandexSpeller. Однако потом пришлось отказаться от его применения по нескольким причинам. Во-первых, это было очень это было очень времязатратно, потому что спеллер мог исправлять ошибки только на небольшом фрагменте текста и приходилось ему давать по одному слову из списка из токенизированных слов. Во-вторых, не ней стоит ограничение по кол-посылок в день (оно довольно небольшое).

**3. Мешок слов**

Для анализа наиболее популярных слов в разных разрезах необходимо было сделать так называемый «мешок слов» - количество употребления в тексте каждого слова. Мы сделали мешки слов по всему датафрейму, по топикам, по годам и месяцам.

Результаты получились следующие:

* На всем датафрейме наиболее популярные слова, на наш взгляд, не несут довольно большого смысла (слова «год», «это», «который»)
* Гораздо интереснее посмотреть на популярные слова в каждом топике. Так, например, в топике «Спорт» популярны слова «матч», «команда», «сборная». На наш взгляд у таких топиков будут хорошие метрики при применении моделей классификации. В то же время у некоторых популярные слова довольно похожи, например у топиков «Экономика» и «Бизнес» популярны слова «процент», «компания», «миллиард», «рубль». Здесь мы ожидаем получить более худшее качество по метрикам, хотя возможно при сочетании с другими словами, оно будет лучше.
* Мешок слов по годам показался также довольно информативным. Так, например, одно из популярных слов в 2020 году было слово «коронавирус», в 2014 и 2022 - «Украина».
* По месяцам оказался наименее информативный мешок слов, так как топ-10 слов практически идентичен у каждого месяца.