Слайды 3-14

Хотим посмотреть, как реализуются цифровые издания разных авторов, чтобы понять, что можем сделать мы. Ориентируемся на 3 проекта, все – семантические издания (и включают в себя семантический поиск).

Пушкин Диджитал → неполное (по крайней мере пока) академическое собрание сочинений Пушкина. Здесь собраны сами тексты, рукописи, прижизненный издания, постановки, фильмы, основанные на этих текстах, варианты, комментарии. Есть система семантического поиска, но поиск только по фразам, читательские сценарии не реализуются. Нас может заинтересовать оформление текста и его вариантов (можно будет ориентироваться на это, если встретятся (точно встретятся) тексты Вознесенского, где разночтения не будут сильно влиять на характеристики → нельзя будет разнести их по разным категориям).

Чехов Диджитал → тоже семантическое издание, поиск устроен интереснее: можно говорить о реализации разных разносторонней навигации по текстам(?), но ограничено: читатель должен очень точно знать, чего хочет (нет рекомендательной системы, а мы хотим); не для широкого круга читателей, которые хотят познакомиться с автором/просто почитать что-нибудь и т.д.

Слово Толстого → наиболее близкий нам проект, есть реализация читательских сценариев, один из которых предлагается на главной странице; есть поиск по фразам, удобные фильтры; тексты делятся на разные категории, по которым читатель может гулять (при этом внутри каждой категории есть подкатегории и поиск); сами категории довольно необычные и интересные (редкие слова). Опять же можно посмотреть, как располагается текст со всеми его вариантами, с комментариями (у читателя есть выбор). Кроме того, можно искать по дате/датам/временному промежутку. Представлена биография Толстого (с иллюстрациями, цитатами из дневника) → у читателя есть возможность ближе познакомиться не только с творчеством, но и с самим писателем.

Всё очень вдохновляет, особенно слово Толстого, но мы хотим дополнить это рекомендательной системой, чтобы неподготовленный читатель смог комфортно пользоваться и знакомиться с Вознесенским (и будет не только семантическая разметка). Рекомендательных систем поэтических текстов много, поэтому ещё будем реализовывать читательские сценарии, как уже говорили. О них расскажет моя коллега.

***Максим***

**Тематическое моделирование**

**Общее**

Суть технологии: ***«в большой коллекции находятся совместно встречающиеся слова, обычно семантически близкие в рамках некоторого унифицированного контекста. Ряды таких слов служат репрезентантами «тем», т. е. того, что характеризует текст с содержательной точки зрения»*** [Орехов 2022]

При этом количество тем – это параметр задающийся вручную (правильно выбрать количество тем – самый проблемный момент в использовании инструмента и основной предмет его калибровки)

Важно отметить разницу ***«обычных» метрик***, нацеленных на фиксировании коэффициентов того или иного параметра на используемом материале, и ***тематического моделирования***, представляющего собой, по сути, способ кластеризации данных.

Кластеризация – на двух уровнях:

1. распределение слов по темам;

2. распределение документов (в нашем случае стихов) по темам

При этом одно и то же слово может относиться к нескольким темам сразу, а документ – являться смесью нескольких тем (поэтому тематическое моделирование описывают как алгоритмы ***«мягкой кластеризации»:*** такие алгоритмы разрешают одной и той же сущности принадлежать сразу к нескольким классам.)

Существуют разные алгоритмы тематического моделирования корпуса. У каждого есть свои недостатки, но более совершенным считается ***LDA (Латентное размещение Дирихле)****.* Однако мы могли бы опробовать и несколько иной алгоритм *–* ***LSA (Латентный семантический анализ)****.* В целом, их можно разделить по принципу применения / неприменения вероятностных расчетов, основа у них одна.

**LSA**

Его суть состоит в том, чтобы:

1. Показать в таблице, как часто все слова встречаются в документах

2. Переупорядочить таблицу так, чтобы слова одной темы и документы похожих тем стояли рядом

Далее, применяя метод разложения матрицы по сингулярным числам из линейной алгебры, используя матрицы-множители, где документы и слова выражены через темы, алгоритм группирует документы, где встречаются одни и те же слова, а затем — слова, найденные в одних и тех же документах.

**LDA**

В отличие от LSA этот алгоритм оперирует категориями распределения вероятностей. Распределение Дирихле предсказывает, какими именно могут быть распределения вероятностей встретить тему в документе или слово в теме.

При этом данный алгоритм идет на ступень выше — показывает, насколько вероятно выбрать **одно из разрешенных распределений значений**.

Наверное, нет смысла более подробно излагать структуру работы этого алгоритма, тем более что ему посвящено наше следующее занятие по R

Пара слов о том, как это реализуется на практике.

Поступающий на вход текст нашего корпуса сперва должен быть очищен от неинформативных стоп-слов и лемматизирован (например, с помощью Mystem, который также может снять омонимию).

Далее, алгоритм работает со специальным типом данных - ***терм-документной матрицы (dtm),*** для преобразования в который можно использовать специальную функцию из R.

Наконец, сам алгоритм также реализуется с помощью готовой функции в R, называющейся LDA и принимающей в качестве параметра, кроме самих данных, количество тем для разбиения.

В дальнейшем, из получившейся модели можно извлекать вероятности попадания в определенную тему для конкретного слова, или же (что для нас более интересно) измерить распределение тем по документам (стихам). Это позволит рекомендательной системе нашего ресурса советовать пользователю стихи А.А. Вознесенского, тематически более или менее далекие от того, которое он читает в данный момент.

Об иных способах кластеризации текстов, применимых к другим способам квантификации данных расскажет моя коллега.