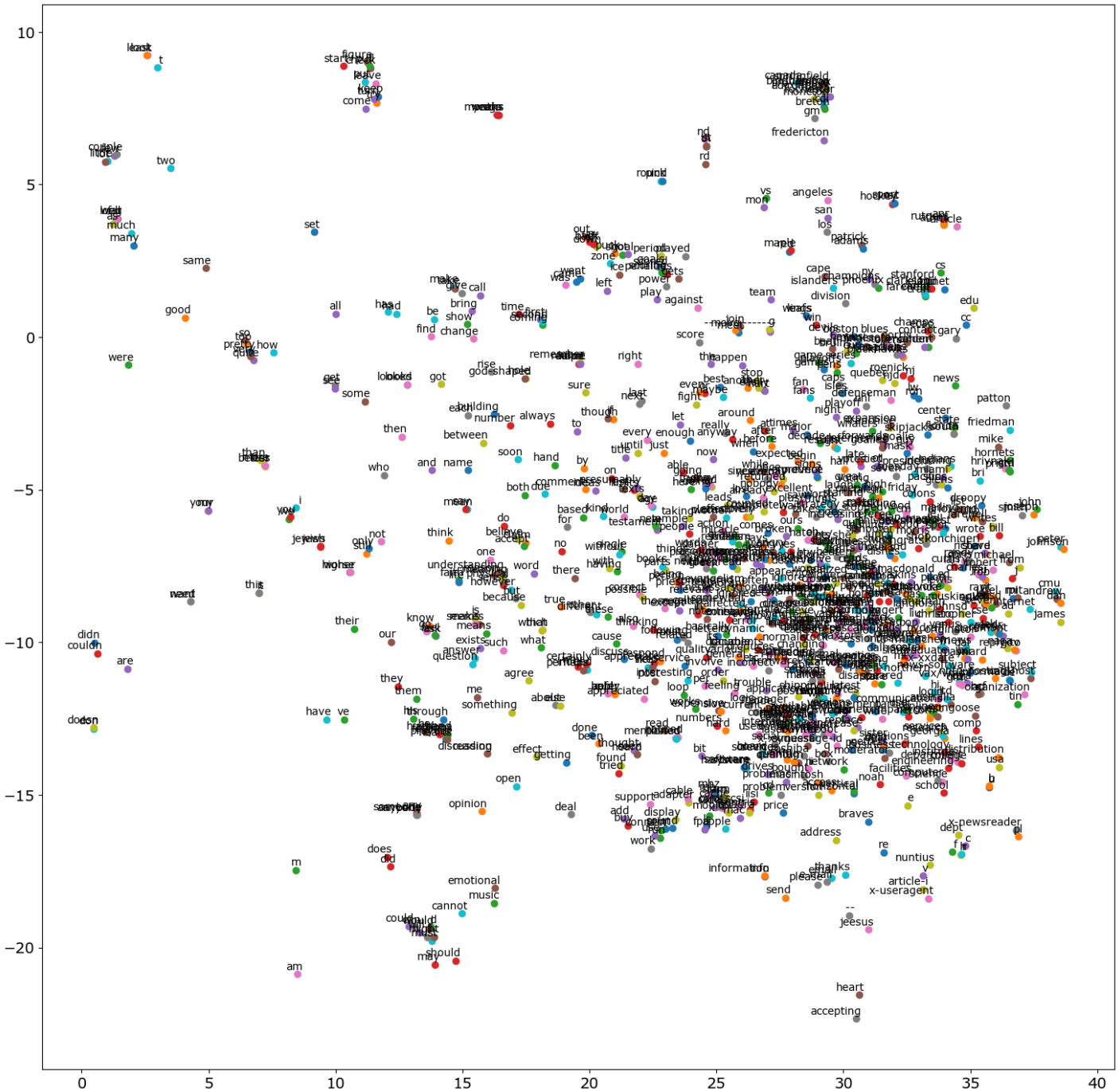
# Отчет Жеглов Дмитрий

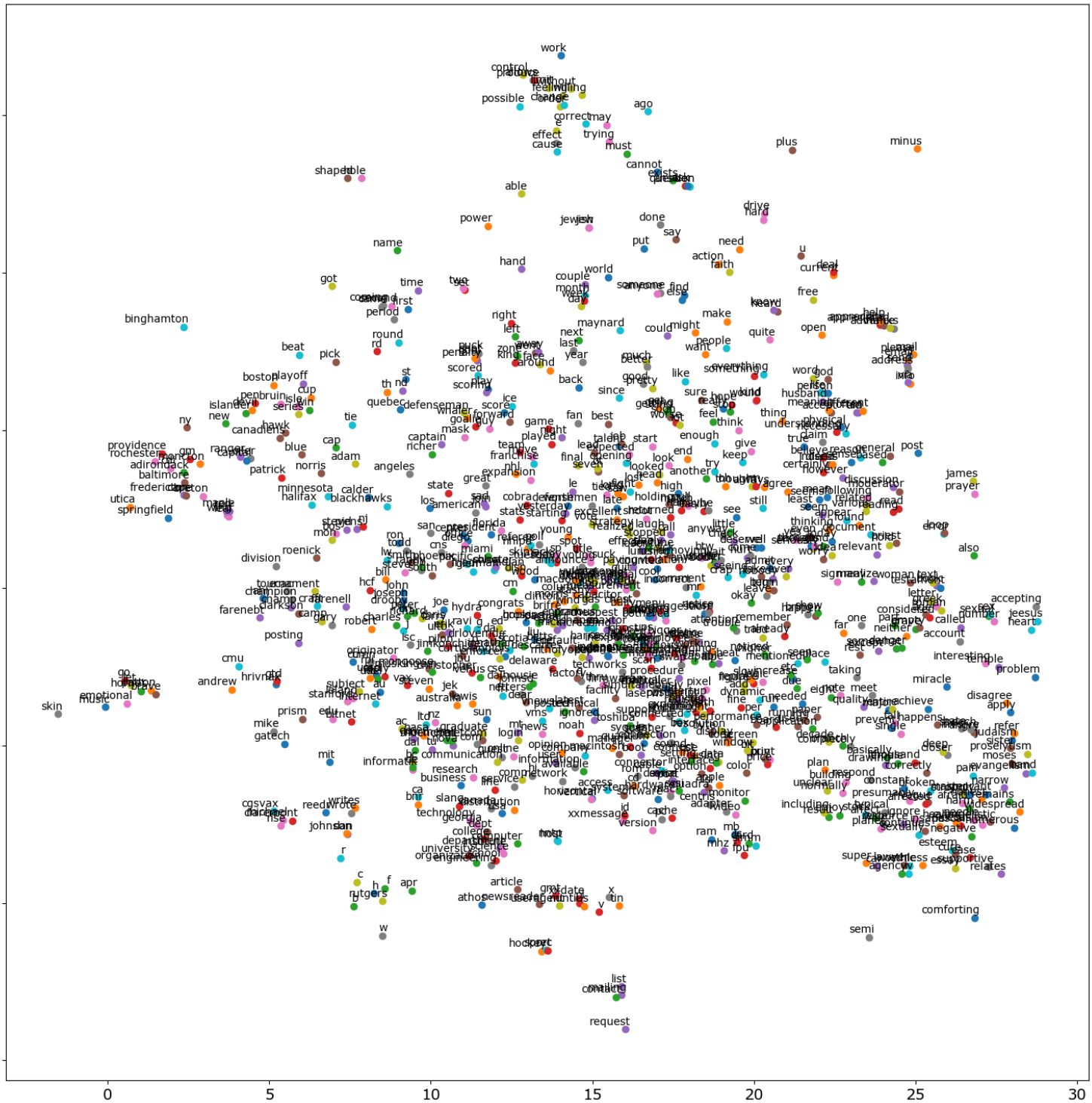
## Пункт 0:

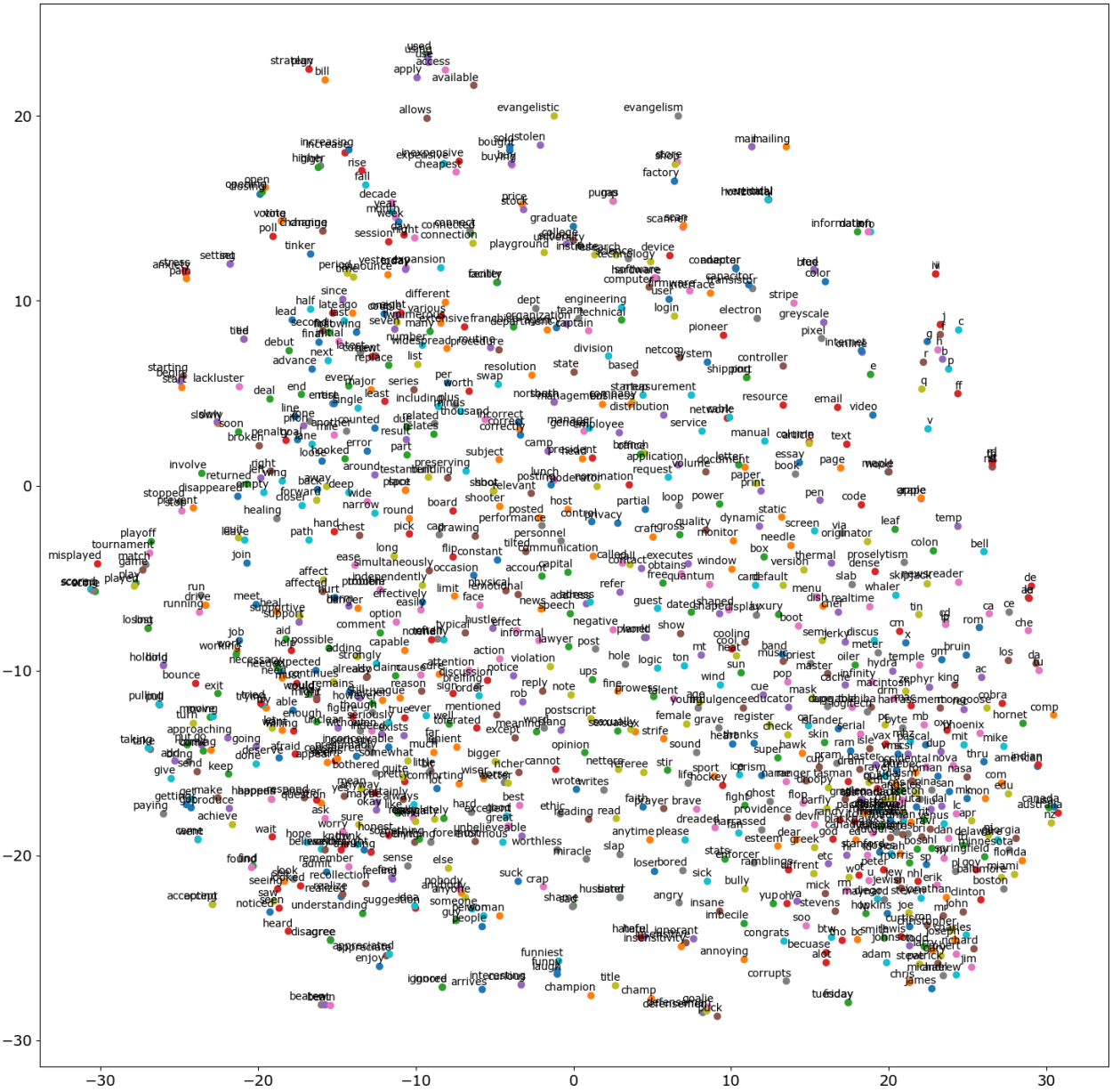
1)добавил в функцию фильтр стоп слов русских и английских
2)удаляю все кроме букв русского и английского алфавита
3)добавил 2 вида стеммина, лематизацию и pymorphy для русских слов

Our model со старой нормализацией



построим tsne на нашем обученным word2vec с размерностью 300 и с новой лемматизацией





Google modelВидим что модели имеют что-то общее, например обе выделили класс образования, локации
Но в тоже время модель от google более разряженно расставляет слова, забавно что god и devil Достаточно близки.

# Пункт 1:

будем преобразовывать наши тексты с помощью новой функции нормализации, использую лемматизацию

воспользуемся count_vectorizer и построим алгоритм k-means++
Получили качество 0.3376477208778841

воспользуемся обученным нами word2vec и построим алгоритм k-means++
Получили качество 0.9690489589195272
Сделаем выбор главных компонент с числом равным 2
Получили качество 0.9656724817107485

воспользуемся GOOGLE word2vec и построим алгоритм k-means++
Получили качество 0.9420371412492966
Сделаем выбор главных компонент с числом равным 2
Получили качество 0.967923466516601
Сделаем выбор главных компонент с числом равным 5
Получили качество 0.6190208216094542

обучим fasttext на нормализованных текстах и построим алгоритм k-means++
Получили качество 0.9780528981429375
Сделаем выбор главных компонент с числом равным 2
Получили качество 0.967923466516601

Видно, что мешок слов проигрывает по качеству
google word2vec лучше при сжатии компонент, чем обученный нами, возможно разница в том, что у нас 100 мерные вектора, а у google 300 мерные и при сжатии выделяются лучше компоненты
Но лучше всех показал себя обученный fasttext, наеврно если скачать 10ГБ предобученный, то получим еще большее качество.