

# Статьи, полезная литература

Надя, Тома



# How to Use t-SNE Effectively

Wattenberg, Martin and Viégas, Fernanda and Johnson, Ian



# How to Use t-SNE Effectively

## About t-SNE

T-distributed Stochastic Neighbor Embedding (t-SNE) — алгоритм снижения размерности, представленный Laurens van der Maaten и Geoffrey Hinton в 2008 году и повсеместно используемый для визуализации векторных пространств.

Основан на машинном обучении.

Схожие многомерные объекты моделируются близко расположенными точками в 2D или 3D-пространстве, отличные — далеко.

## t-SNE

$$p_{j|i} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\mathbf{x}_i - \mathbf{x}_k\|^2 / 2\sigma_i^2)},$$

Если дан набор из  $N$  объектов высокой размерности  $\mathbf{x}_1 \dots \mathbf{x}_N$ , t-SNE сначала вычисляет вероятности  $P(ij)$ , которые пропорциональны похожести объектов  $\mathbf{x}_i$  и  $\mathbf{x}_j$ .

“Похожесть точки данных  $\mathbf{x}_j$  на точку  $\mathbf{x}_i$  является условной вероятностью  $P(j|i)$ , что для  $\mathbf{x}_i$  будет выбрана точка  $\mathbf{x}_j$  в качестве соседней точки, если соседи выбираются пропорционально их гауссовой плотности вероятности с центром в  $\mathbf{x}_i$ ”. ( $P(i|i) = 0$ )

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

## t-SNE (continued)

- Евклидово расстояние подвержено проклятию размерности, и в данных высокой размерности все  $P(ij)$  становятся слишком похожи (сходятся к константе).
- Расстояние корректируется с помощью экспоненциального преобразования.
- Определяется похожее распределение вероятностей для точек в пространстве малой размерности.
- Минимизируется дивергенция Кульбака — Лейблера между двумя распределениями с учётом положения точек.

# t-SNE is useful — but

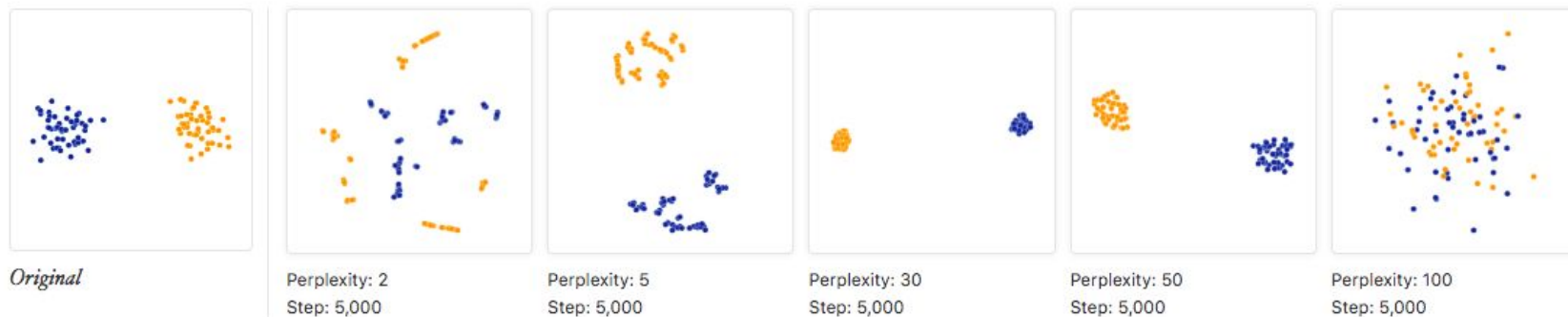
“only if you know, how to interpret it.”

- Нелинейный алгоритм, следовательно разные трансформации на разных участках.
- Изменяемый параметр, перплексия, может сильно исказить график.  
(Перплексия определяет фокус, будет ли он на более глобальных или локальных характеристиках. ~количество близких соседей у точки.)
- Иногда повторные запуски алгоритма могут давать разные

# Важность гиперпараметров (1)

- perplexity: 5-50

unexpected behavior

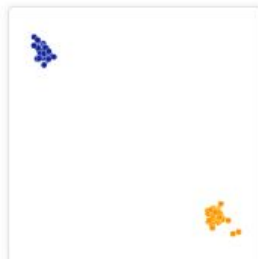


## Важность гиперпараметров (2)

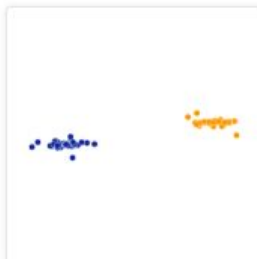
- 1,000 iterations in this case (подбирается индивидуально)



*Original*



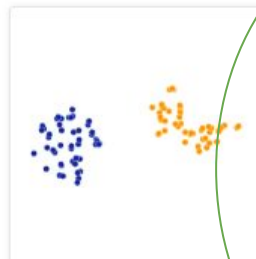
Perplexity: 30  
Step: 10



Perplexity: 30  
Step: 20



Perplexity: 30  
Step: 60



Perplexity: 30  
Step: 120

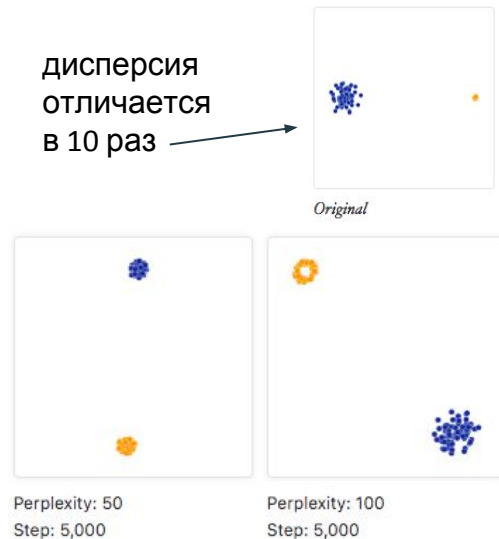


Perplexity: 30  
Step: 1,000



# Размер кластера не имеет значения

- Разница в среднеквадратическом отклонении не отразится на размере кластеров в выводе, потому что t-SNE нормирует расстояния между точками в кластере к плотности кластера (т.е. высокая плотность станет меньше, и наоборот.)



# Расстояние между кластерами может что-то значить

- 3 распределения Гаусса по 50 точек, расстояние варьируется в 5 раз



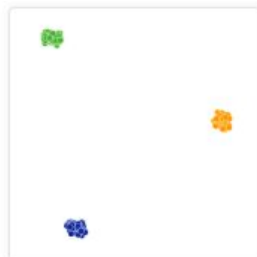
*Original*



Perplexity: 2  
Step: 5,000



Perplexity: 5  
Step: 5,000



Perplexity: 30  
Step: 5,000



Perplexity: 50  
Step: 5,000



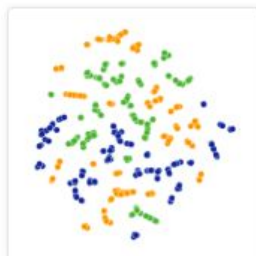
Perplexity: 100  
Step: 5,000

## ...а может и не значить

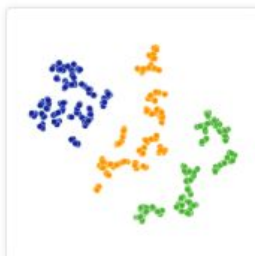
- 3 распределения Гаусса по 200 точек, расстояние варьируется в 5 раз



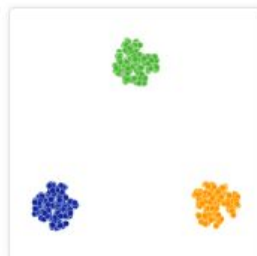
*Original*



Perplexity: 2  
Step: 5,000



Perplexity: 5  
Step: 5,000



Perplexity: 30  
Step: 5,000



Perplexity: 50  
Step: 5,000



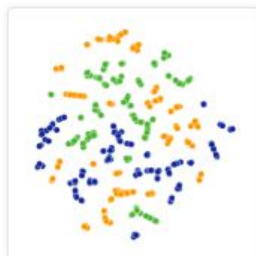
Perplexity: 100  
Step: 5,000

## ...а может и не значить

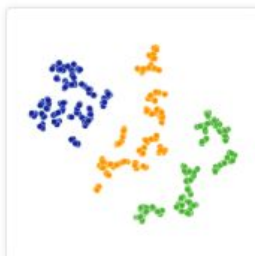
- 3 распределения Гаусса по 200 точек, расстояние варьируется в 5 раз



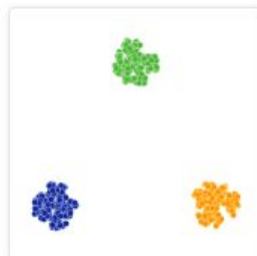
*Original*



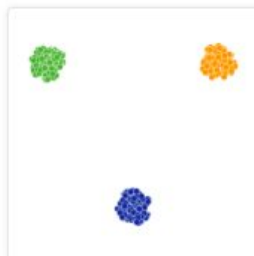
Perplexity: 2  
Step: 5,000



Perplexity: 5  
Step: 5,000



Perplexity: 30  
Step: 5,000



Perplexity: 50  
Step: 5,000



Perplexity: 100  
Step: 5,000

# Структура на выходе бывает обманчива

равномерные распределения в многомерных пространствах ведут себя, как на сфере

- 500 точек распределения Гаусса в 100 измерениях



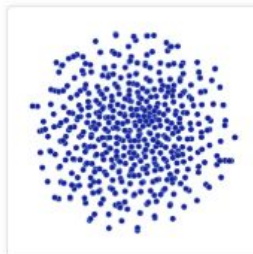
*Original*



Perplexity: 2  
Step: 5,000



Perplexity: 5  
Step: 5,000



Perplexity: 30  
Step: 5,000



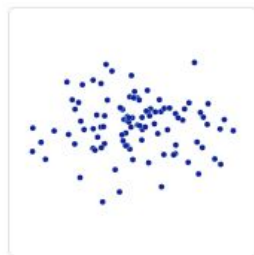
Perplexity: 50  
Step: 5,000



Perplexity: 100  
Step: 5,000

controversial

# Форму можно обнаружить



*Original*



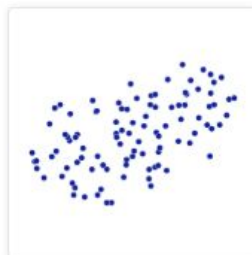
Perplexity: 2  
Step: 5,000



Perplexity: 5  
Step: 5,000



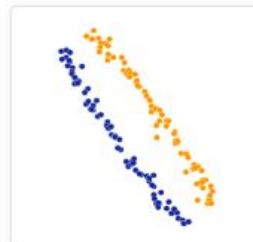
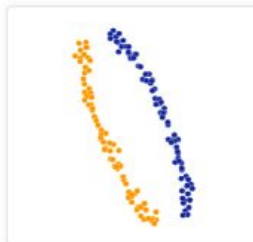
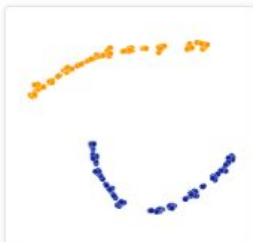
Perplexity: 30  
Step: 5,000



Perplexity: 50  
Step: 5,000



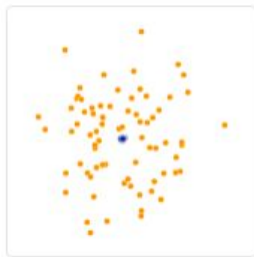
Perplexity: 100  
Step: 5,000



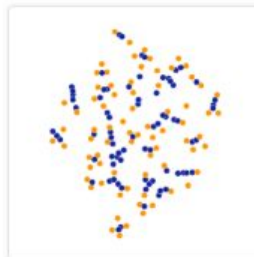
# Топологию определить сложно

только это уже изыски

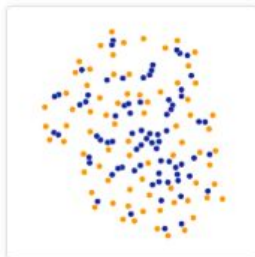
- 75 точек, симметричные распределения, размерность 50, плотность отличается в 50 раз, одно распределение внутри



*Original*



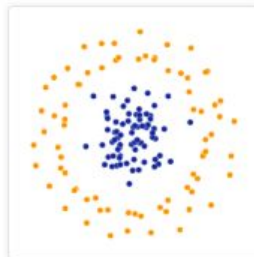
Perplexity: 2  
Step: 5,000



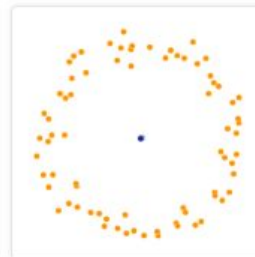
Perplexity: 5  
Step: 5,000




Perplexity: 30  
Step: 5,000



Perplexity: 50  
Step: 5,000



Perplexity: 100  
Step: 5,000



# Visualization of Dynamics Reference Graphs

Ivan Rodin, Ekaterina Chernyak, Mikhail Dubov, Boris Mirkin





# Visualization of Dynamics Reference Graphs

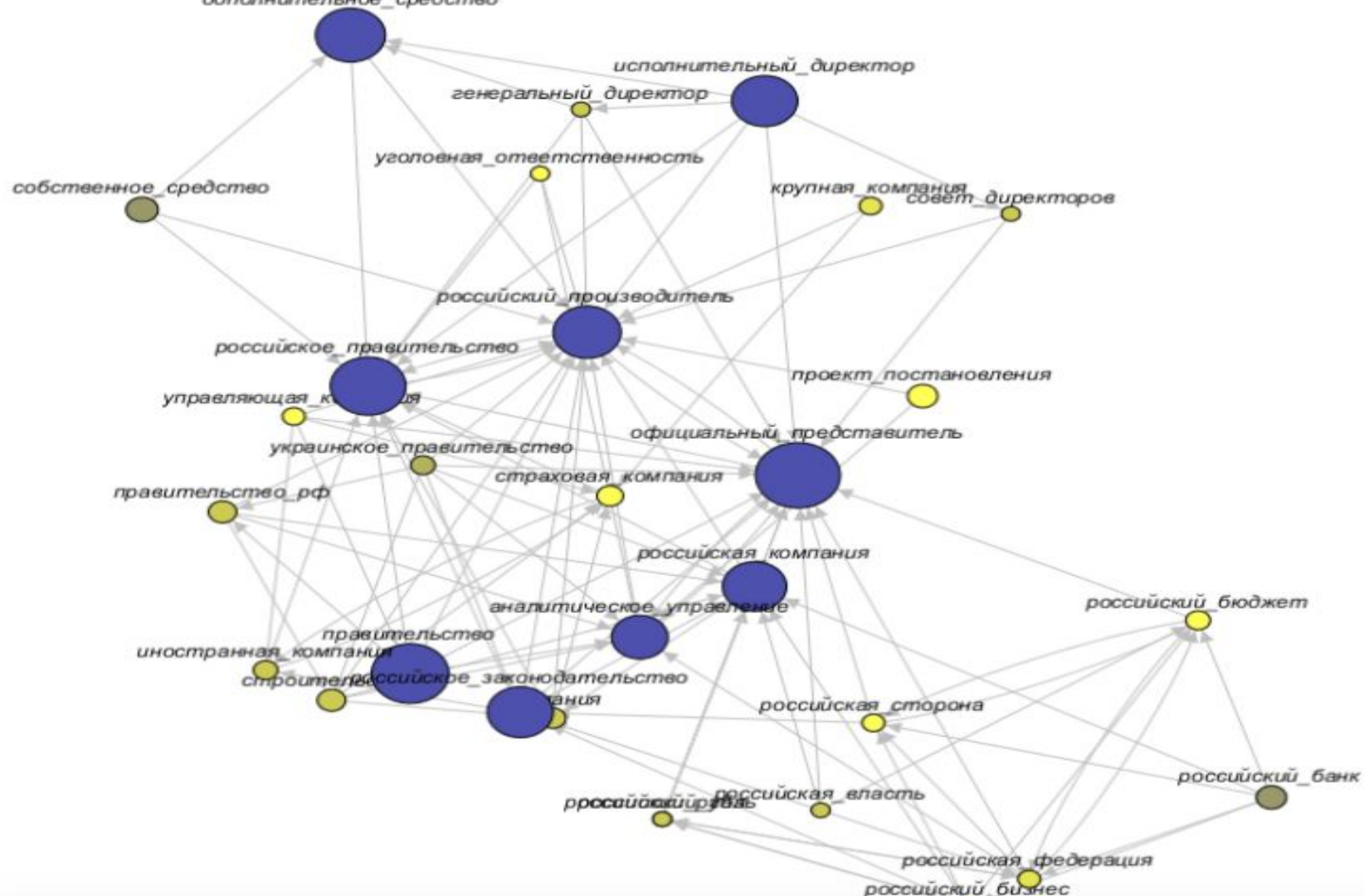
## About

- референс граф основан на ключевых словах из корпуса текстов с временной разметкой
- отличие от графа со-вхождений: референс граф - ориентированный, что позволяет подробнее понимать отношения между ключевыми словами
- полезен для отслеживания изменения значимости концептов

1. Необходимо разметить корпус по временным периодам
2. Выделить список концептов (= ключевых слов (пишут про топ самых частотных слов (?)) (мб лучше tf-idf?)).
3. Для каждого временного отрезка система считает а) значимость концептов; б) значимость связей концептов попарно
4. Определение порога, после которого связи будут показаны в графе

## Датасет 1: RuNeWC

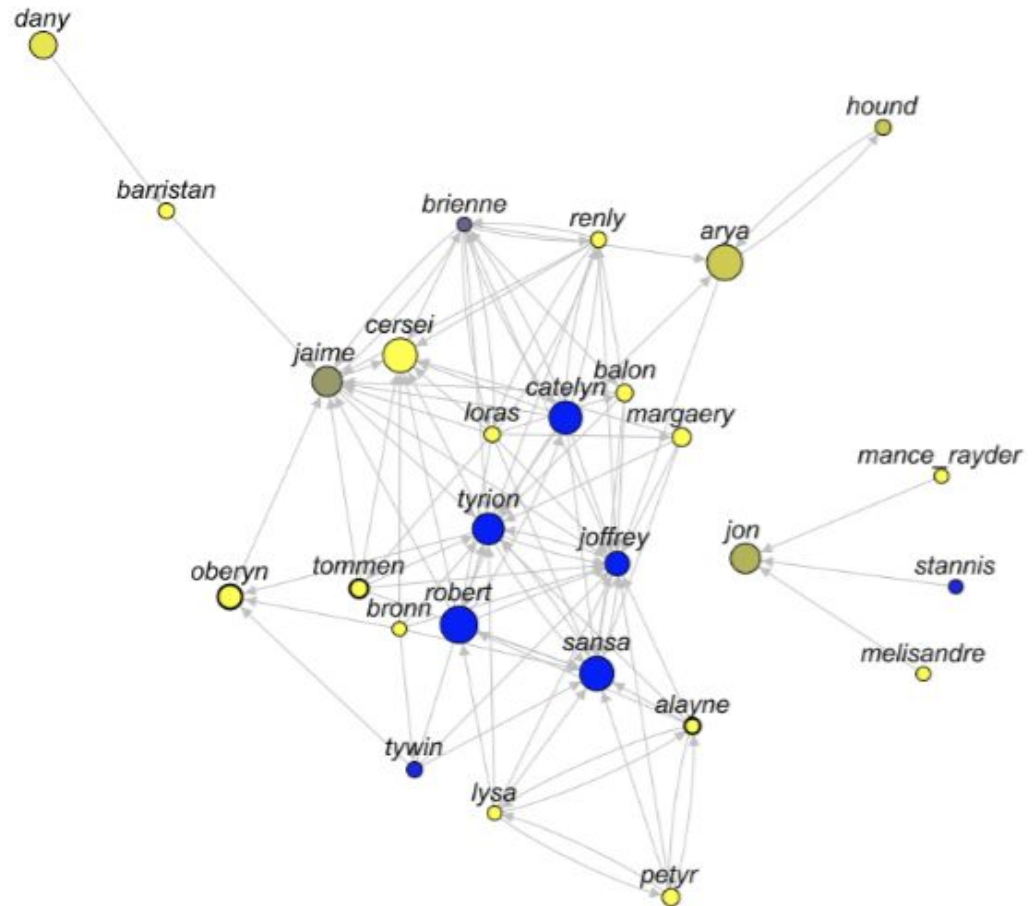
- 4061 газета, 26 периодов (каждый - 2 недели)
- Извлекли автоматические концепты (Adjective + Noun, Noun + Noun)
- Взяли 250 самых популярных фраз и 100 популярных слов.
- Отсеяли стоп-слова
- Порог - 29+ упоминаний



## Датасет 2: персонажи из “Песни льда и пламени”

- 4 книги, каждая глава дробится на 2-7 частей
- получилось 50 документов - 14 временных промежутков.
- Порог - 1. Персонаж есть в графе, если появлялся 2+ раза

# Временной промежуток 11 - конец третьей книги



## Дальнейшие планы авторов:

- позволить юзерам создавать свой корпус и визуализации на базе их софта
- поддержка извлечения контекстуальных синонимов
- решение проблем с повторяющимися нодами

## Тулзы для построение динамического графа:

- **GraphStream** (использовался авторами статьи)
- **KeyLines**
- **Gephi**



## Ссылки на статьи

How to Use t-SNE Effectively

- <https://distill.pub/2016/misread-tsne/>

Visualization of Dynamics Reference Graphs

- <https://aclweb.org/anthology/W16-1406>

Литература, на которую мы пока  
поглядываем:

‘Data visualization with python and javascript’ by  
Kyran Dale