

# Методы и алгоритмы решения задач классификации и рекомендации текстов



Виталий Зайчук / А-нью Технолоджи



# Виталий Зайчук

team-lead фронтенд разработки в А-нью Технолоджи

 anews.com

 smayluk@gmail.com

 @smayluk

 github.com/smayluk

# О чем пойдет речь

- Как из блога создать проект кулинарного сайта с внедрением машинного обучения

# Почему вам это может быть интересно?

- Можете ли вы использовать текстовые данные, чтобы усовершенствовать продукт над которым работаете?
- А как расширить его функциональность?

Сегодня, я вам расскажу именно об этом. И не важно, кто вы — опытный Data Scientist, или только начинающий Python разработчик

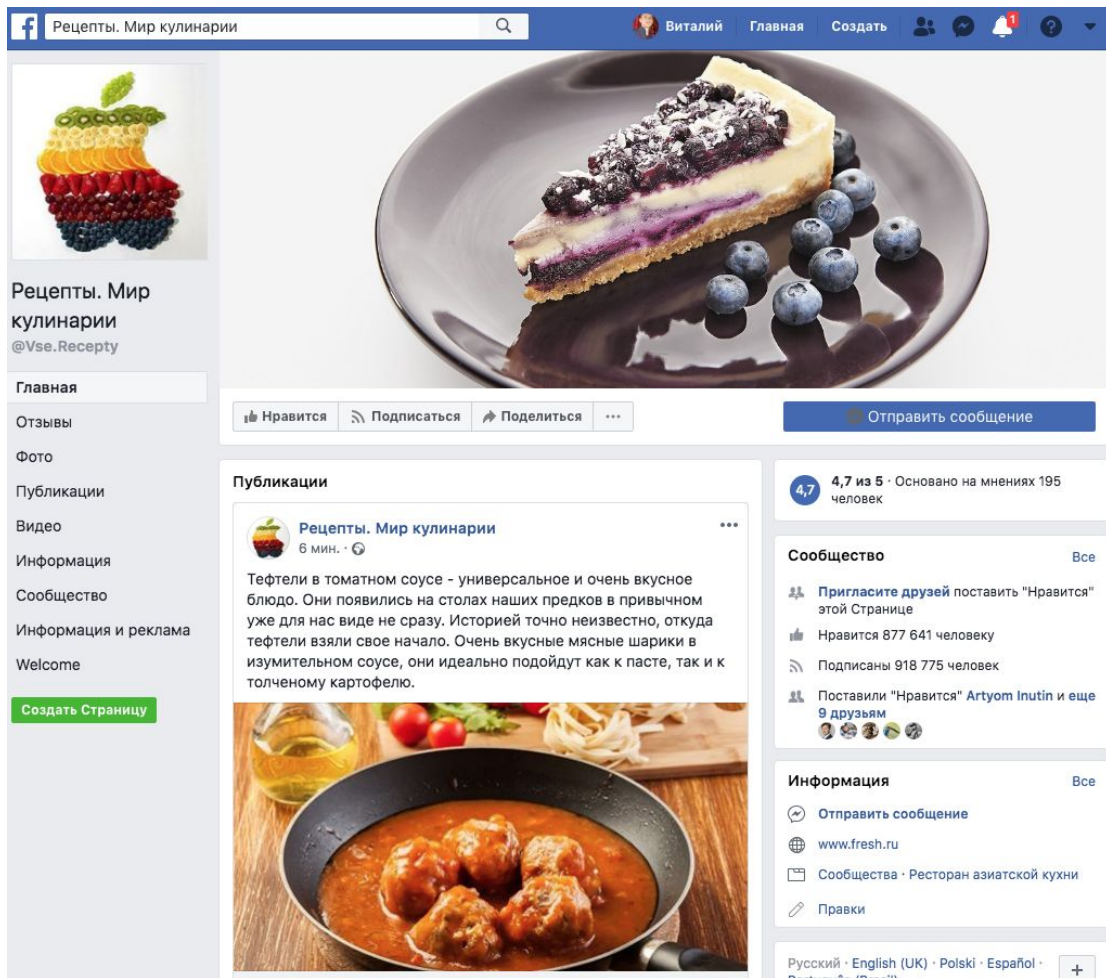
# План

- постановка задачи
- подготовка данных для обучения
- сравнительный анализ методов классификации
- поиск близких по смыслу текстов
- выводы

# Предыстория...

## Что приготовить?

Над этим вопросом ежедневно задумываются миллионы людей по всему миру.



Рецепты. Мир кулинарии  
@Vse.Recepty

Главная  
Отзывы  
Фото  
Публикации  
Видео  
Информация  
Сообщество  
Информация и реклама  
Welcome

Создать Страницу

Нравится Подписаться Поделиться ...

Отправить сообщение

4,7 из 5 · Основано на мнениях 195 человек

Сообщество Все

Пригласите друзей поставить "Нравится" этой Странице

Нравится 877 641 человеку

Подписаны 918 775 человек

Поставили "Нравится" Artyom Inutin и еще 9 друзьям

Информация Все

Отправить сообщение

www.fresh.ru

Сообщества · Ресторан азиатской кухни


Правки

Русский · English (UK) · Polski · Español · Português (Brasil)

Публикации

Рецепты. Мир кулинарии 6 мин. · 🌐

Тефтели в томатном соусе - универсальное и очень вкусное блюдо. Они появились на столах наших предков в привычном уже для нас виде не сразу. Историей точно неизвестно, откуда тефтели взяли свое начало. Очень вкусные мясные шарики в изумительном соусе, они идеально подойдут как к пасте, так и к толченому картофелю.



# Что нужно было сделать?

При разработке  
кулинарного сайта решить  
следующие задачи  
Machine Learning:

1. Разделить рецепты на категории.
  2. Сделать подборку похожих рецептов.
-

# Классификация текстов

---



# Задача: Разделение рецептов на категории

**fresh**

Войти

Искать рецепт по названию, кухне и т.д.

☐ Поиск ингредиентов



Закуски

Салаты

Первые блюда

Вторые блюда

Гарниры

Соусы

Десерты

Завтраки

Перекусы

Пикник

Заготовки на зиму



Рецепты дня

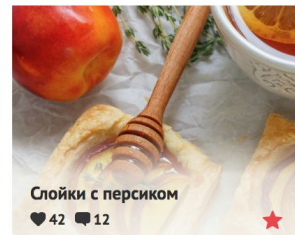
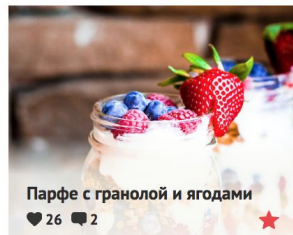
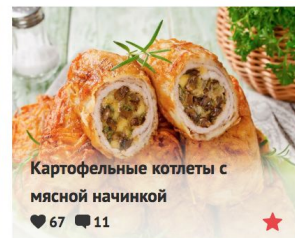
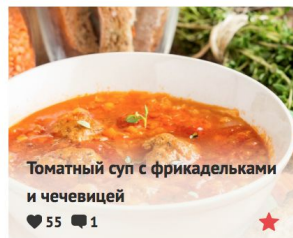
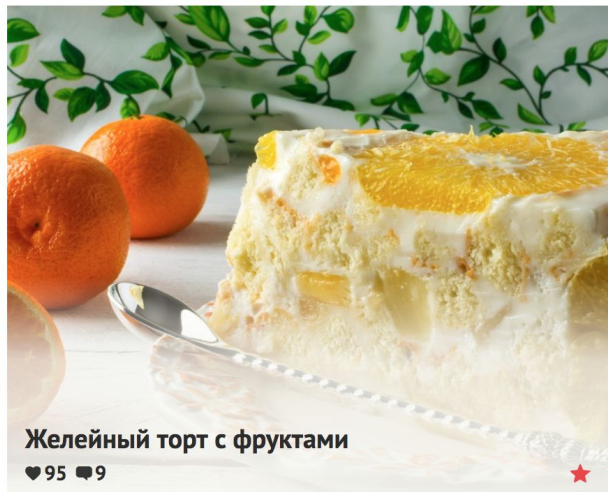
Популярные рецепты

Новинки

Сезонные рецепты

Видео рецепты

Рекомендации рецептов



# Добавление рецепта

Автозаполнение системой полей:

- Категория
- Кухня

## Добавление рецепта

Название рецепта

Креветки в чесночно-винном соусе

## Заглавное фото



## Описание

Креветки в чесночно-винном соусе можно подать как отдельное блюдо с гарниром, например, рисом; как соус для пасты либо как отличную горячую закуску к пиву или другим алкогольным напиткам. Блюдо готовится за кратчайшее время из минимума продуктов. Креветки получаются просто изумительными, а чесночно-винный соус настолько вкусным, что его можно просто вымакивать хлебом!

## Категория

Закуски

## Кухня

Азиатская

# Задачи классификации:

- при миграции базы с одного проекта на другой разделить рецепты на категории;
- при добавлении пользователем нового рецепта, “подсказать” ему тип кухни и категорию.

# Формат исходных данных рецепта в блоге

- Название рецепта
- Картинка
- Текст рецепта



## Торт "Кокосовое наслаждение"

### Приготовление:

1. Сгущенное молоко комнатной температуры взбить с яйцами комнатной температуры.
2. Добавить просеянную муку и разрыхлитель и ещё раз взбить. Должно получиться густое, как на оладьи, тесто.
3. Форму для выпечки накрыть пергаментом и слегка смазать сливочным маслом.
4. Вылить тесто в форму, разровнять лопаткой и отправить в разогретую до 170 градусов духовку на 25-30 минут.  
*(По желанию, разделить тесто на 2 части, чтобы торт состоял из 4-х*

# Требования и ресурсы

- по возможности найти и использовать **готовые методы и алгоритмы;**
  - внедрить в проект **быстро и дешево.**
-

# Что нужно для классификации?

Размеченные данные для обучения модели

# Способы получения размеченных данных


- проставить метки классов вручную
- найти готовый data set
- спарсить из web

# Сбор данных из web

Собираем данные с метками  
(категориями), на которых модель  
будет обучаться

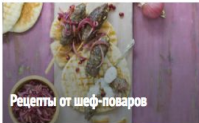
[Закуски](#) [Салаты](#) [Супы](#) [Вторые блюда](#) [Выпечка](#) [Каши](#) [Запеканки](#) [Пироги](#) [Торты](#) [Десерты](#) [Напитки](#) [Найти рецепт](#)

### РЕЦЕПТЫ ПРИГОТОВЛЕНИЯ БЛЮД ОТ ПРОФИ

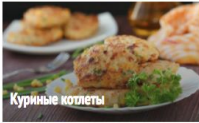


Невозможно не испечь: 3 супер-рецепта домашнего хлеба


### ПОПУЛЯРНОЕ





Рецепты от шеф-поваров



Куриные котлеты




Пироги с капустой



Поваренок.ру


## Каталог рецептов

#### Первое блюдо



борщ ботвинья бульон  
гаспачо капуста кулеш  
лагман мисо окрошка  
рассольник свекольник  
сладкие супы солянка суп уха  
харчо хаш шурпа  
щи из капусты


#### Основное блюдо



азу бефстроганов бешбармак  
бигус биточки бифштекс  
бризоль бурито в горшочке  
в кляре вареники галушки  
гарнир голубцы гратен  
грибные блюда гуляш деруны  
долма draniki

**Развернуть все рубрики**

#### Закуски



бастурма буженина бургер  
бутерброды гренки жульен  
жюльен заливное  
икра овощная канапе кимчи  
лечо мидии морковь по-  
корейски пастрома паштет  
печеночный торт роллы  
салаты селедка

**Развернуть все рубрики**



# Очистка и подготовка текстовых данных

Прежде чем мы сможем передать категориальные данные, такие как текст или слова, на вход алгоритма машинного обучения, нам нужно их преобразовать в числовую форму, предварительно обработав текст.

# Обработка текста



# Методы классификации

Все методы основаны на  
векторном представлении  
текста

При решении задачи были  
использованы следующие  
методы:

- mean word2vec
  - Tf-idf weighted word2vec
  - Doc2vec
  - fastText
-

# Метрики качества классификации

**Accuracy** (доля правильных ответов) - считаем на сколько объектах даем правильный ответ и делим на размер выборки

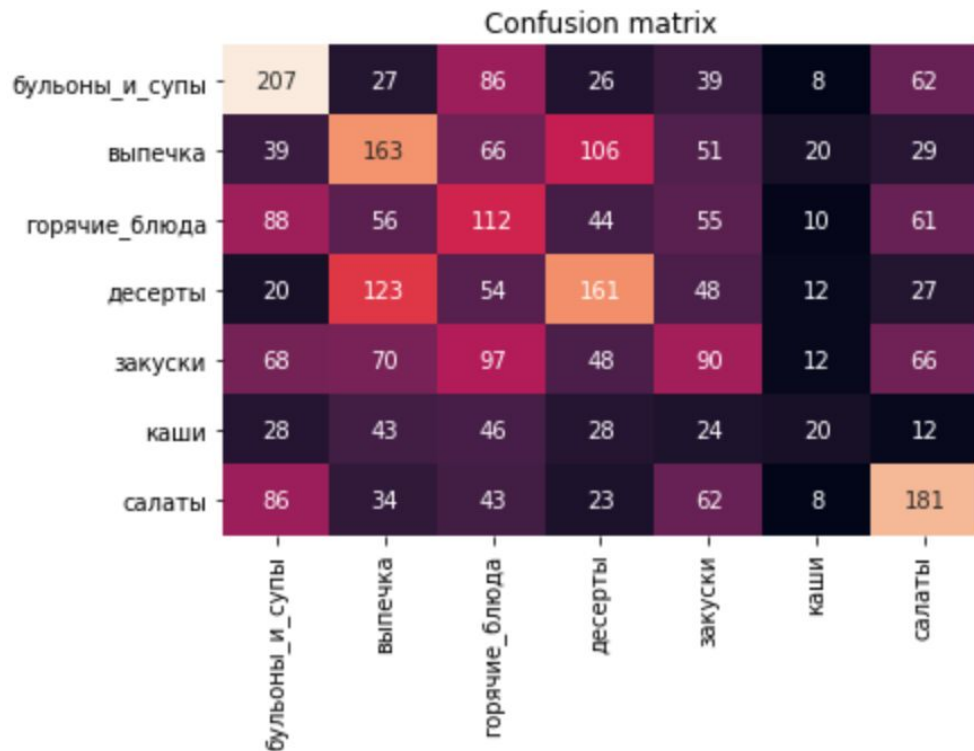
# Классификация с помощью word2vec

- технология разработанная Google
- технология основана на векторном представлении слова
- метод доступен в пакете Gensim:

```
from gensim.models import Word2Vec
```

# Результаты усредненного word2vec

**Accuracy: 0.32**



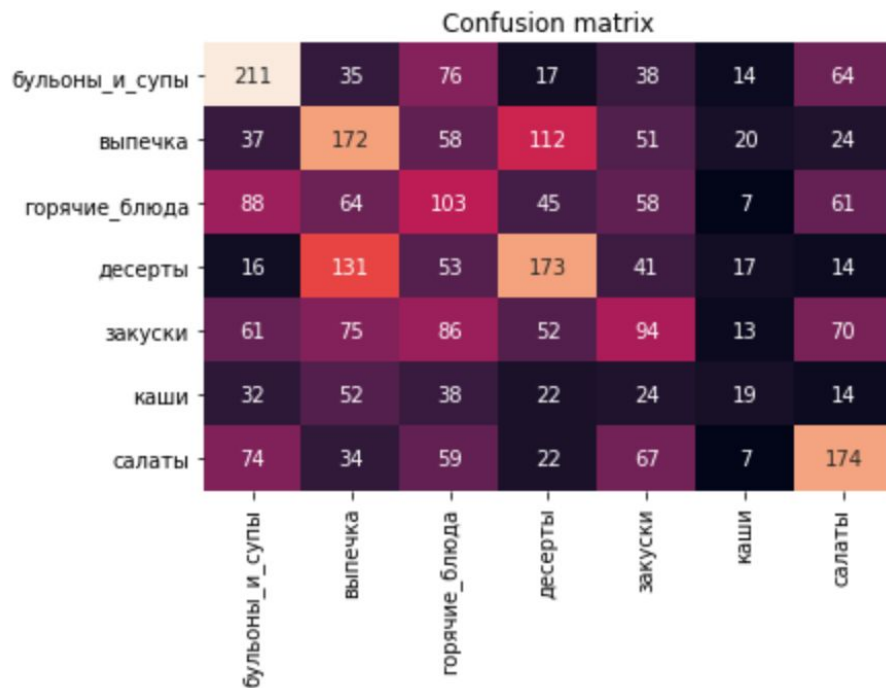
# Классификация с помощью Tf-idf weighted word2vec

**Tf-idf** - это мера, используемая для оценки важности слова.

Например, слово **ГОТОВИТЬ**, встречается во всех текстах и не несет полезной информации.

# Результаты усредненного word2vec с Tf-idf весами

**Accuracy: 0.33**





# Классификация методом doc2vec

- алгоритм дает возможность получить вектор документа
- метод доступен в пакете Gensim:

```
from gensim.models.doc2vec import Doc2Vec
```

# Результаты Doc2vec

**Accuracy: 0.55**

Confusion matrix

бульоны_и_супы	350	10	41	7	22	5	20
выпечка	12	306	48	79	23	1	5
горячие_блюда	76	57	185	10	64	6	28
десерты	8	127	23	267	13	2	5
закуски	37	63	116	37	131	3	64
каши	42	21	25	31	17	54	11
салаты	27	9	29	7	52	3	310
	бульоны_и_супы	выпечка	горячие_блюда	десерты	закуски	каши	салаты

# Классификация библиотекой fasttext

- open-source библиотека от Facebook
- основана на представлении трёхсимвольных n-грамм:

where  
↓ $n=3$   
[<wh, whe, her, ere, re>]

- fasttext быстрее чем word2vec, так как трёхсимвольных n-грамм меньше, чем слов

# Результаты fastText

**Accuracy: 0.74**

Confusion matrix

бульоны_и_супы	387	1	28	1	2	2	11
выпечка	0	379	30	48	7	0	0
горячие_блюда	29	22	353	3	54	8	4
десерты	0	74	0	324	2	7	0
закуски	13	45	161	7	168	2	51
каши	34	0	20	30	0	131	2
салаты	5	0	15	0	20	0	409
	бульоны_и_супы	выпечка	горячие_блюда	десерты	закуски	каши	салаты

# Пример работы классификатора fasttext

Картофельные лепешки с грибами и сыром

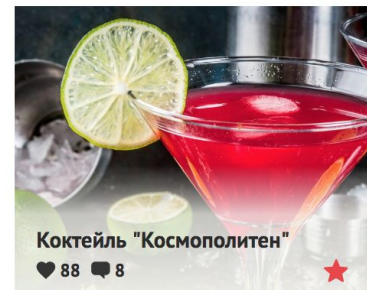
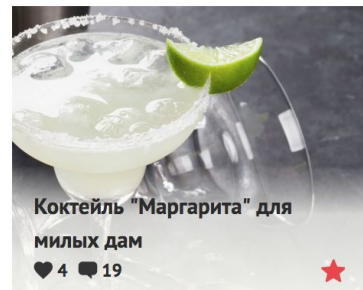
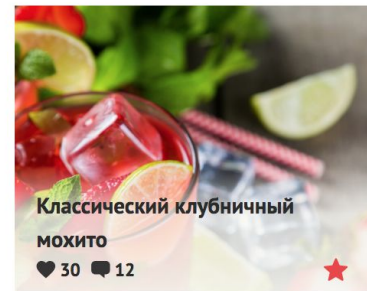
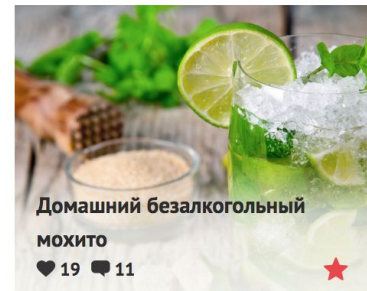
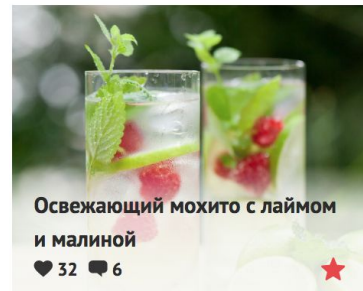
`[['выпечка', 0.359375), ('горячие_блюда', 0.304688), ('закуски', 0.242188)]]`



# Что удалось выяснить?

1. **word2vec** дал низкую точность классификации;
2. на малом объеме данных **Doc2Vec** дает лучшие показатели классификации;
3. **fastText** самый быстрый и точный в работе.

## Напитки:



# Использование классификатора

---

# Загрузка обученной модели классификатора

```
start = time.time()
classifier = fasttext.supervised('classifier_model.txt', 'model')#10,4 МБ
print('Время загрузки модели:', (time.time() - start)*1000, 'ms')
```

Время загрузки модели: 427.95681953430176 ms



# Подготовка данных для модели классификатора

```
article[:100]
```

```
'<p><u><span>Ингредиенты на 4 порции:</span></u></p><ul><li><span>капуста брокколи — 250 г </span>'
```

```
start = time.time()
#cleanhtml
#tokenize
#remove_stopwords
#lemmatize
content = text_processing(article)
print('Время обработки текста:', (time.time() - start)*1000, 'ms')
```

Время обработки текста: 12.279987335205078 ms

```
content[:100]
```

```
'ингредиент порция капуста брокколи г пшеничный мука г яйцо куриный шт соль вкус тертый сыр пармезан '
```

# Классификация

```
start = time.time()
pred = classifier.predict_proba([content])
print('Время классификации:', (time.time() - start)*1000, 'ms')
pred
```

Время классификации: 0.14829635620117188 ms

[['горячие\_блюда', 0.408203]]

# Рекомендация текстов

---

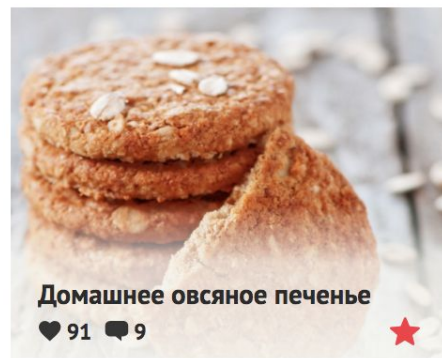
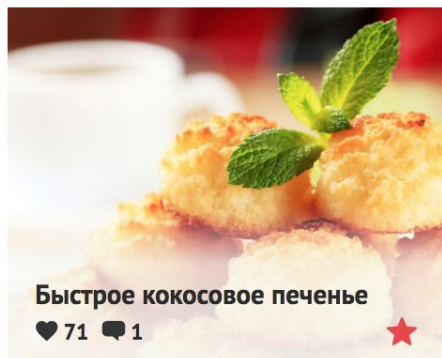
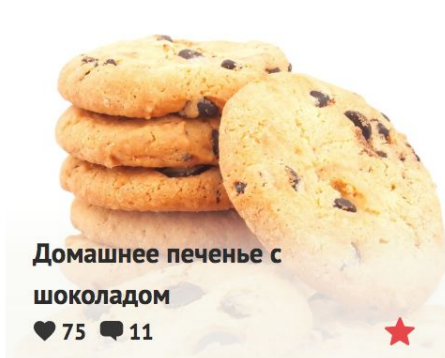
# Задача рекомендательной системы:

- найти и предложить пользователю похожие рецепты

# Поиск похожих рецептов



Похожие рецепты:



# Методы рекомендательной системы

Задача решалась  
определением сходства между  
двумя предложениями  
методом транспортной задачи  
и методом векторного  
представления документа

При решении задачи были  
использованы следующие  
методы:

- Word Mover's Distance
  - Doc2Vec
-

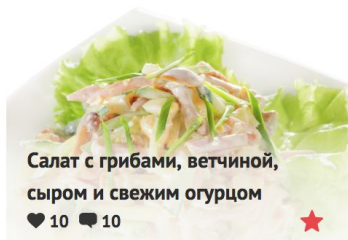
# Word Mover's Distance

- Помогает найти близкие по смыслу тексты.
- Определяет расстояние между двумя документами, как оптимальную стоимость перемещения слов из одного документа в другой, с помощью векторного представления слов.
- Метод доступен в пакете Gensim

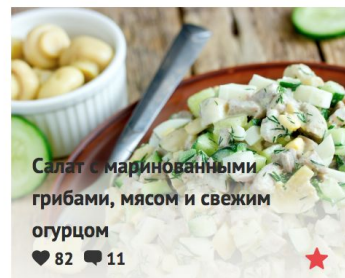
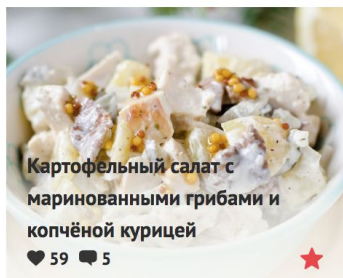
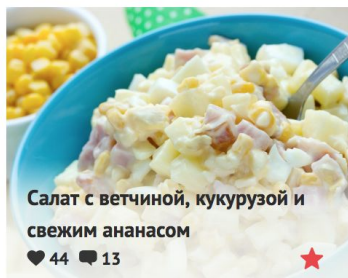
```
from gensim.similarities import WmdSimilarity
```

# Поиск близких текстов методом Word Mover's Distance

Ищем похожие рецепты для:



Похожие рецепты:



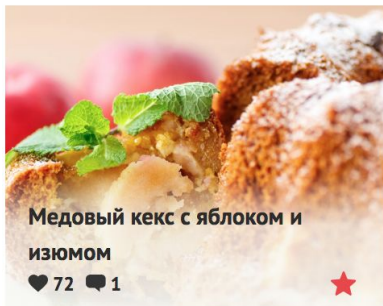


# Doc2vec для поиска близких текстов

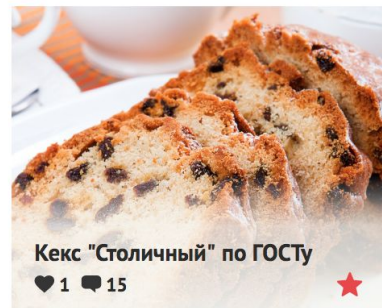
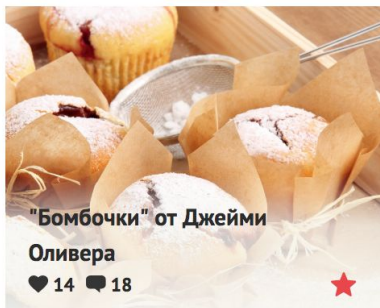
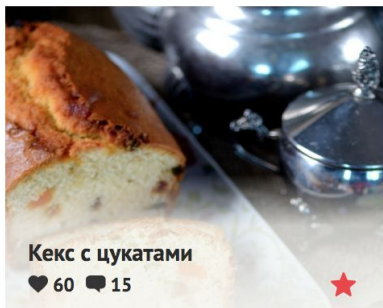
- Doc2Vec, можно использовать не только для классификации, но и для решения задачи поиска близкого текста

# Поиск близких текстов методом Doc2vec

Ищем похожие рецепты для:



Похожие рецепты:



# Использование модели рекомендаций

---

# Загрузка обученной модели word2vec

```
start = time.time()
d2v_model = Doc2Vec.load('recommendation.model')#13,5 MB
print('Время загрузки модели:', (time.time() - start)*1000, 'ms')
```

Время загрузки модели: 164.97302055358887 ms

# Поиск близких текстов

```
start = time.time()
y = d2v_model.infer_vector(content.split(" "))
res = d2v_model.docvecs.most_similar([y])
print('Время поиска близких текстов:', (time.time() - start)*1000, 'ms')
print(res[:3])
```

Время поиска близких текстов: 5.180835723876953 ms

```
[('372', 0.7513513565063477),
 ('1672', 0.7319275736808777),
 ('1090', 0.7106802463531494)]
```


# Что дальше или как улучшить рекомендательную систему?

- Сбор поведенческих данных пользователей;
- Построение рекомендательной системы на их основе.



# Выводы

1. Создан и внедрен классификатор на основе метода fastText
2. Разработана и внедрена рекомендательная система на основе метода doc2vec

 <https://github.com/smayluk/recipes-recommendation>

# Q&A

 smayluk@gmail.com

 @smayluk