Evaluating of word embeddings hyper-parameters of the master data in Russian-language information systems

Grinkina T. V. (tgreenkina@gmail.com), Dudnikov S. Y. (htserj@gmail.com),
Mikheev P. I. (petr_miheev@mail.ru)

Bauman Moscow State Technical University, Moscow, Russia

Evaluating of word embeddings hyper-parameters for master data quality support task is conducted in this work. We explore the structure and management of master data. Also, we describe a method of training the embeddings model for elements and methods for estimating the resulting vectors. Using a corpus of training and validation data sets, we are conducting an experiment on models with a different set of parameters, and get results. In conclusion, we present the main recommendations for the hyper-parameters setting.

Key words: master data, quality of master data, word embeddings, the Skip-Gram (Word2Vec) model, hyper-parameter settings

Исследование параметров настройки векторного представления текстовых элементов справочной информации русскоязычных информационных систем

Гринкна Т.В. (tgreenkina@gmail.com), Дудников С.Ю. (htserj@gmail.com),
Михеев П.И. (petr_miheev@mail.ru)

МГТУ им Баумана, Москва, Россия

Исследуется вопрос оптимального выбора векторного представления слов в задаче поддержки качества элементов справочной информации русскоязычных информационных систем. Приводится описание особенностей элементов нормативно справочной информации и методы их обработки. Приводится метод построения векторного пространства элементов и методы оценки, полученного пространства. Описывается основной корпус тренировочных и валидационных данных, приводится схема эксперимента и полученные результаты, на основании которых делается вывод о наиболее оптимальном наборе параметров настройки модели векторного представления слов. В заключении приводятся методы улучшения качества векторных представлений.

Ключевые слова: нормативно-справочная информация (НСИ), обработка НСИ, векторное представление слов, архитектура Skip-Gram (Word2Vec), параметры настройки skip-gram (Word2Vec)

1 Введение

Оперативная обработка справочной информации является значимой задачей и должна быть эффективно решена в рамках бизнес-процессов деятельности предприятия. Для удовлетворения этого требования системы обработки должны обладать пропускной способностью и мощностью, достаточной для принятия оперативных решений.

Такие системы должны поддерживать не только качество имеющихся в них данных, но и обновлять базу без потери качества информации. Важным условием является сохранение лексико-семантической специфики области деятельности предприятия, что не представляется возможным, используя только стандартные методы и подходы построения векторных представлений слов.

В данной статье исследуются набор параметров настройки модели векторных представлений текстовых элементов справочной информации для создания оптимального семантического пространства, способного достаточно точно решать основные задачи поддержки качества ведения справочной информации русскоязычных информационных систем.

2 Особенности элементов справочной информации и их векторных представлений

Нормативно-справочная информация по структуре представляет собой сущность с описательными или числовыми характеристиками. Например,

*Ключ гаечный комбинированный 41*41*, (1)

где *Ключ* – сущность, *гаечный, комбинированный* – описательные параметры, *41*41* – числовые характеристики.

Стандартный подход для поддержки качества такого типа информации состоит из двух этапов: разработка многоуровневого классификатора и создание шаблона элемента для каждой подгруппы. На базе созданного шаблона элементы вначале приводятся к единой форме, а затем сравниваются между собой по параметрам. Такой подход не позволяет сравнивать каждый элементы как самостоятельную единицу, в следствии чего качество справочных данных остается низким.

Гораздо более оптимальным является приведение элементов справочников к соизмеримым между собой представлениям, таким как вектор.

Основной проблемой при таком подходе является отсутствие открытых корпусов обученных векторов для такого типа текстовых данных. Это связано с содержанием в корпусе большого числа аббревиатур и слов специфичных для данной области. Существующие же корпуса покрывают от 30% до 40% массива данных и остальная группа слов нуждаются в самостоятельном обучении.

В данной работе исследуются параметры модели обучения векторных представлений слов, с целью подобрать наиболее оптимальный набор и дать необходимые рекомендации для обучения подобных корпусов справочной информации.

3 Исследование векторного представления элементов и настройка параметров

3.1 Корпус справочных элементов

Корпус справочных элементов представляет собой корпоративный справочник номенклатуры объемом 264 тысячи записей. В него входит перечень основных материалов, запасных комплектов и предоставляемых услуг на хранение.

Базовая предобработка включает в себя приведение слов к нижнему регистру, затем из элементов удаляются все символы пунктуации и специальные знаки, далее убираются все числовые значения и элемент разбивается на токены (слова). Алгоритм реализован с помощью Python-библиотеки NLTK(Bird, 2016).

3.2 Модель Word2Vec

В качестве модели для обучения векторов была выбрана архитектура Skip-gram [1] (Word2Vec), представленная на Рисунке 1. Во-первых, такая архитектура показывает

лучший результат для редких слов, по сравнению с архитектурой CBOW. Во-вторых, дальнейшее обучение глубоких контекстуальных моделей может базироваться на результатах, полученных в ходе исследования параметров skip-gram.

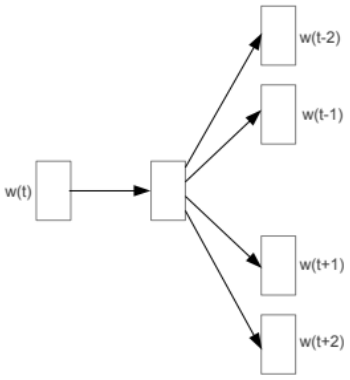

Рис. 1: Архитектура skip-gram

Принцип обучения рассматриваемой архитектуры заключается в прогнозировании контекста $C = \{w(t-2), w(t-1), w(t+1), w(t+2)\}$ по слову $w(t)$.

Первым этапом инициализируются две матрицы весов $W_{N\times V}$ и $W'_{V\times N}$, где N – размер словаря, а V – размерность векторного представления. Входной вектор равен $x_i = \{x_1, ..., x_k, ..., x_V\}$, где $x_k = 1$, если $k = i$ и $x_k = 0$, если $k \neq i$. Представление вектора на промежуточном слое равно по Формуле 2:

$$h_i = x^T \times W \tag{2}$$

Далее столбец v_j матрицы $W'_{V\times N}$ скалярно умножается на вектор промежуточного слоя, позволяя получить оценку для каждого слова в словаре – Формула 3:

$$u_j = v_j^T \cdot h_i \tag{3}$$

Апостериорное значение вероятности для каждого слова из контекста C рассчитывается по формуле софтмакса – Формула 4

$$p(w_j|w_I) = y_j = \frac{exp(u_j)}{\sum_{j'=1}^{k} exp(u'_j)}, \tag{4}$$

где w_I – входное слово, тогда w_O – выходное слово и функция потерь равна – Формула 5

$$L = -logp(w_{O1}, .., w_{OC}|w_I) \tag{5}$$

Обновление весов матриц $W_{N\times V} = \{w_{ij}\}$ и $W'_{V\times N} = \{w'_{ij}\}$ – Формула 6, 7

$$w_{ij}^{new} = w_{ij}^{old} - \eta \frac{\partial L}{\partial w_{ij}} \tag{6}$$

$$w_{ij}^{'new} = w_{ij}^{'old} - \eta \frac{\partial L}{\partial w'_{ij}} \tag{7}$$

где η – шаг обучения.

Дополнительно качество векторов повышается с помощью следующих двух методов.

Первый – метод сэмплирования, который применяется для балансирования корпуса тренировочных слов. Суть метода заключается в отбрасывании слов из обучающей выборки. Вероятность, что слово останется заданно Формулой 8

$$P(w_i) = \left(\sqrt{\frac{z(w_i)}{sample}} + 1\right) \cdot \frac{sample}{z(w_i)} \tag{8}$$

где $z(w_i)$ – доля слова в словаре, $sample$ – настраиваемый параметр сэмплирования.

Второй метод позволяет повышать скорость обучения – это метод негативного сэмплирования. Поскольку за каждый шаг обучения обновляются все веса, что является достаточно трудоемким вычислением, наиболее целесообразным является обновлять веса для части отрицательных примеров (ошибочные ответы). Количество таких примеров из корпуса равно $negative$. Вероятность попадание слова в подвыборку вычисляется по Формуле 9

$$P(w_i) = \frac{z(w_i)^{\frac{3}{4}}}{\sum_{i=0}^{V}(z(w_i))^{\frac{3}{4}}} \tag{9}$$

3.3 Параметры модели Word2Vec

Модель skip-gram обладает следующим набором ключевых параметров.

1. Size – размерность векторного представления слов.
2. Window – максимальный размер от целевого слова до прогнозируемого в архитектуре Skip-gramm.
3. Min count – минимальная частота употребления слов в корпусе. В случае, если слово встречается меньше заданного, оно игнорируется.
4. Negative – параметр негативного сэмплирования.
5. Learning rate – коэффициент скорости обучения.
6. Sample – коэффициент сэмплирования.

Тестовые значения параметров приведены в таблице, жирным выделены значения по умолчанию [3, 4].

Таблица 1: Тестовые значения параметров skip-gram

Параметры	Значения
Size	25 / 50 / **100** / 200 / 400 / 800
Window	1 / 2 / 3 / 4 / **5** / 6 / 7 / 8
Min count	0 / **5** / 10 / 20 / 50 / 100 / 200 / 400 / 800 / 1000 / 1200 / 2400
Negative	1 / 2 / 3 / **5** / 8 /10 / 15
Learning rate	0.0125 / **0.025** / 0.05 / 0.1
Sample	0 / 1e-1 / 1e-2 / **1e-3** / 1e-4 / 1e-5 / 1e-6 / 1e-7 / 1e-8 / 1e-9

4 Эксперимент

В качестве базового уровня качества векторов выбраны вектора обученные на архитектуре skip-gram с заданными параметрами по умолчанию в Таблице 1

4.1 Внутренняя оценка векторов

Обучение векторных представлений слов должно достаточно качественно решать задачу определения семантически близких элементов справочной информации. Под семантической близостью понимается определение косинусной близости между векторными представлениями – Формула 10

$$similarity(w_x, w_y) = \frac{w_x \cdot w_y}{\|w_x\| \cdot \|w_y\|} \tag{10}$$

Так как вектор обучается для токенов (слов), каждый элемент можно представить как набор из k токенов – Формула 11

$$el = \{w_1, w_2, ..., w_k\} \tag{11}$$

Тогда вектор элемента рассчитывается по Формуле 12

$$x_{el} = \frac{1}{k} \sum_{i=1}^{k} w_i \tag{12}$$

В качестве тестовой выборки используются размеченные элементы корпоративного справочника Номенклатура, в которой содержаться 379 пар элементов однозначно соответствующих друг другу по смыслу, то есть показатель $similarity$ для каждой из них равен 1. Оценкой качества в таком случае будет показатель среднего значения семантической близости каждой пары элементов тестовой выборки на различных векторных представлениях [3].

4.2 Внешняя оценка векторов

Справочный массив данных информационных систем применяется для широкого спектра задач, одна из которых является классификацией справочных элементов по категориям. Качество векторов так же будет влиять и на качество классификатора.

Для оценки эффективности модели [3] на каждой векторной модели обучается классификатор для 12 классов по методу опорных векторов (SVM) и далее подсчитывается показатель F1-score. Настройка классификатора не производится для более точного определения качества классификации от входных векторов.

4.3 Результаты эксперимента

4.3.1 Size

Оптимальный параметр size равен 25. Это объясняется тем, что размер корпуса слов равен 15 тысячам, тогда как величины размерности вектора такие как 200, 400 и т.д. приводят к переобучению модели. Следовательно классификатор на переобученных векторах показывает более лучший результат, по сравнению с векторами меньшей размерности.

Таблица 2: Результаты исследований параметра size

Size	Внутренняя оценка	Внешняя оценка
25	0.928	0.045
50	0.909	0.047
100	0.902	0.057
200	0.901	0.059
400	0.903	0.061
800	0.904	0.057

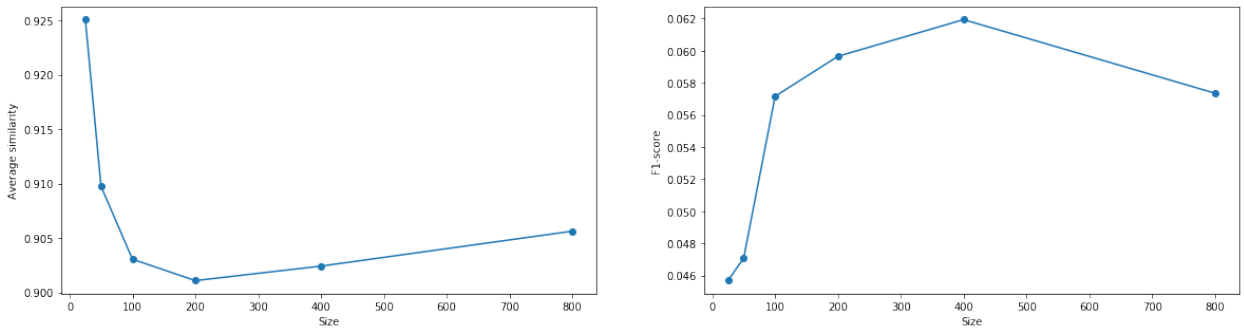

Рис. 2: Результаты исследований параметра size

4.3.2 Min count

Таблица 3: Результаты исследований параметра min count

Min count	Внутренняя оценка	Внешняя оценка
0	0.920	0.066
5	0.902	0.06
10	0.898	0.051
20	-	0.054
50	-	0.056
100	-	0.051
200	-	0.055
400	-	0.059
800	-	0.048
1000	-	0.045
1200	-	0.048
2400	-	0.054

В корпусе справочников встречается достаточно много уникальных слов (в среднем больше половины объема всего словаря), в следствии чего уже при отбрасывании слов, чья частота ниже 20 возникают неопределенности. В связи с эти среди обученных слов удается найти лишь небольшую часть и тест либо некорректен, как в случае внешнего испытания, либо не пройден, как в случае внутреннего.

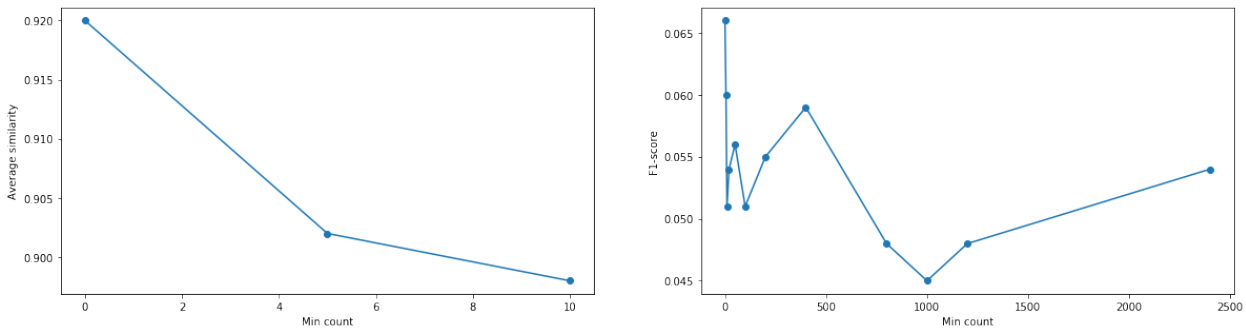

Рис. 3: Результаты исследований параметра min count

4.3.3 Window

В связи с тем что элементы справочника имеют достаточно ограниченную длину, в среднем один элемент содержит около 5 слов. Внутренняя оценка качества вектора ухудшается, когда размер окна превышает это среднее значение. В то время как более низкие значения не позволяют достаточно точно изучить тематическое сходство токенов.

Таблица 4: Результаты исследований параметра window

Window	Внутренняя оценка	Внешняя оценка
1	0.908	0.054
2	0.908	0.055
3	0.905	0.050
4	0.904	0.058
5	0.902	0.059
7	0.901	0.058
8	0.900	0.056

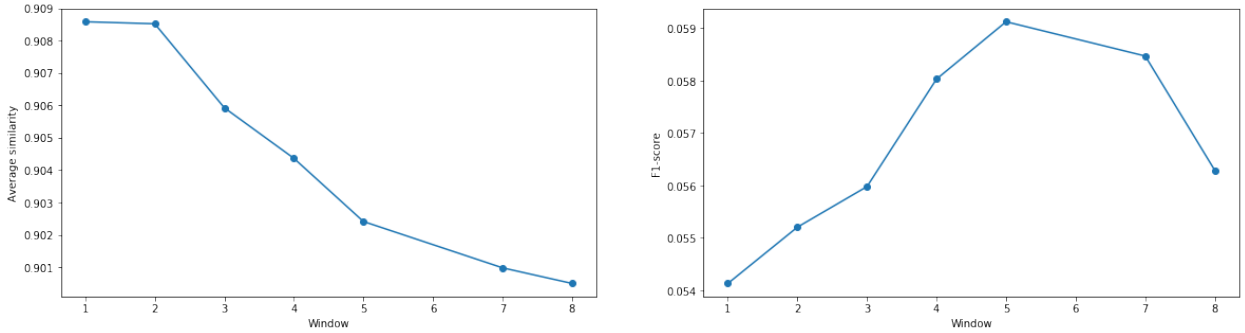

Рис. 4: Результаты исследований параметра window

4.3.4 Learning rate

При низких значениях шага обучения модель получается более точной, но скорость обучения довольно низкая, в то же время при высоких показателях шага процесс обучения нестабилен. Оптимальным вариантом являются значения 0.025 или 0.05.

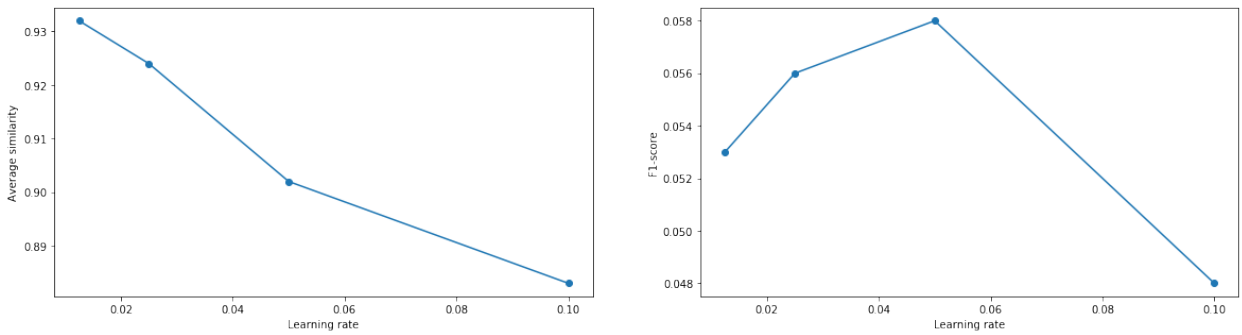

Рис. 5: Результаты исследований параметра learning rate

Таблица 5: Результаты исследований параметра learning rate

Learning rate	Внутренняя оценка	Внешняя оценка
0.0125	0.932	0.053
0.025	0.924	0.056
0.05	0.902	0.058
0.01	0.883	0.048

4.3.5 Sample

При низких порогах вероятность информативных слов быть отброшенными становится больше. Максимальное качество модели по обоим оценкам достигается в точке 1e-05, далее значения в обоих случаях резко падают. Это связано с тем что информативные высокочастотные слова так же как не информативные начинают с равной вероятностью отбрасываться.

Таблица 6: Результаты исследований параметра sample

Sample	Внутренняя оценка	Внешняя оценка
0	0.5	0.049
0.1	0.899	0.064
0.01	0.899	0.061
0.001	0.902	0.055
0.0001	0.944	0.049
1e-05	0.999	0.044
1e-06	0.587	0.046
1e-07	0.585	0.045
1e-08	0.585	0.042
1e-09	0.585	0.043

4.3.6 Negative

Чем выше параметр негативного сэмплирования, тем выше результаты как по внутренней, так и по внешней оценке

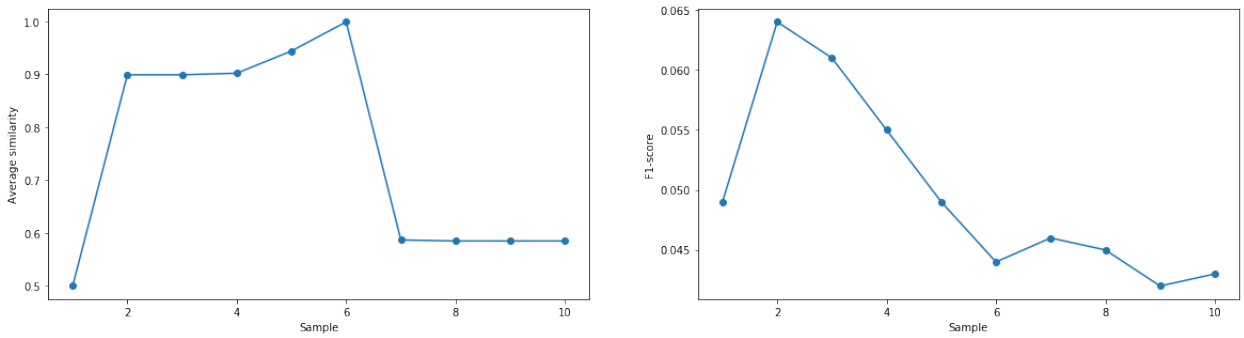

Рис. 6: Результаты исследований параметра sample

Таблица 7: Результаты исследований параметра negative

Negative	Внутренняя оценка	Внешняя оценка
1	0.895	0.057
2	0.904	0.058
3	0.905	0.059
5	0.903	0.060
8	0.901	0.063
10	0.901	0.065
15	0.902	0.068

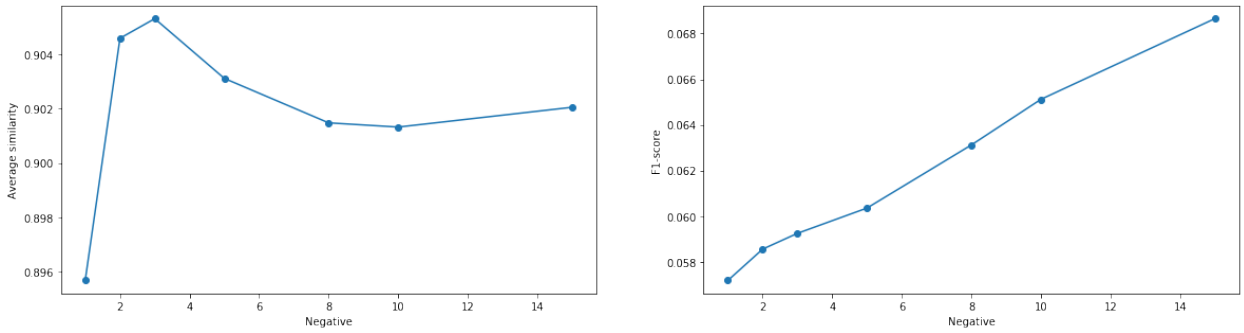

Рис. 7: Результаты исследований параметра negative

4.3.7 Сравнение с базовыми векторами

Оптимальный набор параметров в сравнении с базовыми векторами представлен в Таблице 8.

Таблица 8: Итоговый набор параметров

	Size	Window	Min count	Negative	Learning rate	Sample
Базовые значения	100	5	5	5	0.025	1e-03
Оптимальные значения	25	5	0	15	0.05	1e-05

При решении такого типа задач параметр Size, Window, Min count следует выбирать из свойств корпуса тренировочных данных.

Параметры обучения Learning rate, Negative, Sample в основном имеют одинаковое влияние и не зависят от параметров элементов справочника. Их значения следует выбирать из общих рекомендаций для решения прикладных задач.

5 Заключение

В результате работы была выбрана и исследована модель Skip-gram с различными комбинациями наборов параметров.

На базе проведенных экспериментов подготовлены выводы о влиянии композиции параметров модели на качество векторных представлений элементов справочников.

Рассмотренная методика подбора композиции параметров обучения векторного представления позволяет эффективным способом решать подобного рода задачи, связанные с применением нормативно-справочной информации в русскоязычных информационных системах.

Литература

1. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

2. Fanaeepour, M., Makarucha, A., & Lau, J. H. (2018). Evaluating Word Embedding Hyper-Parameters for Similarity and Analogy Tasks. arXiv preprint arXiv:1804.04211.

3. Wang, B., Wang, A., Chen, F., Wang, Y., & Kuo, C. C. J. (2019). Evaluating Word Embedding Models: Methods and Experimental Results. arXiv preprint arXiv:1901.09785.

4. Chiu, B., Crichton, G., Korhonen, A., & Pyysalo, S. (2016). How to train good word embeddings for biomedical NLP. In Proceedings of the 15th workshop on biomedical natural language processing (pp. 166-174).