ПОСТРОЕНИЕ АССОЦИАТИВНОГО РЯДА ХЕШТЕГОВ С ИСПОЛЬЗОВАНИЕМ СЕТИ СОВМЕСТНОЙ ВСТРЕЧАЕМОСТИ И ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ ХЕШТЕГОВ

Макрушин С.В., Блохин Н.В.

Финансовый университет при Правительстве Российской Федерации,
Россия, г. Москва Ленинградский проспект, д.49
SVMakrushin@fa.ru, blokhin.nv@bk.ru

Аннотация: В исследовании предложено формальное определение проблемы построения семантического пути в виде задачи многокритериальной оптимизации на графе, разработана методика решения задачи и предложены примеры автоматически построенных семантических путей на основе собранных данных о совместной встречаемости хештегов социальной сети Инстаграм.

Ключевые слова: семантическая навигация, оптимизация на графе, построение пути, рекомендация хештегов, сеть совместной встречаемости.

Хештеги являются важным механизмом семантической навигации в социальных сетях. В работе рассматривается задача построения ассоциативного ряда между двумя заданными хештегами в сети совместной встречаемости хештегов, построенной для сообщений из социальной сети Инстаграм.

В социальных сетях публикация часто дополнительно описывается набором хештегов, которые дополняют друг друга. Большое количество хештегов повышает популярность сообщения, но ручной подбор нескольких десятков адекватных сообщению хештегов является трудоемкой задачей. Наша цель – разработать алгоритм, который позволит расширить небольшое количество хештегов, придуманных автором. Целью исследования является создание алгоритма, который по паре заданных хештегов возвращает ассоциативный ряд хештегов, то есть последовательность семантически связанных хештегов, которая логически связывает начальный и конечный хештеги, заданные пользователем. Из полученного ассоциативного ряда автор может выбрать актуальные для него хештеги для расширения набора хештегов, сопровождающих публикацию.

В данном исследовании информационной базой для алгоритма построения ассоциативного ряда является сеть (граф) совместной встречаемости хештегов. В такой сети хештеги являются узлами

(вершинами), и факт употребления пары хештегов в одном сообщении Инстаграм порождает связь (ребро) между ними. В качестве веса связи выступает количество совместных упоминаний хештегов.

Для построения сети совместной встречаемости был собран корпус из 14,6 млн сообщений Инстаграм. Для исключения случайных совпадений связь между хештегами создавалась только в случае, если они употреблялись вместе хотя бы дважды. В результате была получена сеть, состоящая из 1,7 миллионов узлов-хештегов и 63,9 миллионов взвешенных связей. Сбор и предварительная обработка данных были реализованы на языке программирования Python, хранение сети – в графовой базе данных ArangoDB.

Анализ построенной сети показал, что в ней имеется значительное количество «хабов», обладающих десятками и сотнями тысяч связей. В результате исследования хештегов-хабов выяснилось, что 80% всех случаев совместных упоминаний хабов обычно обеспечивается всего несколькими десятками хештегов, сильно связанных с данным хабом. Именно такие ключевые связи в сети демонстрируют наибольшую смысловую взаимосвязь между хештегами. Избавившись от большого количества слабых смысловых связей в сети, можно как снизить вычислительную сложность реализации построения ассоциативной цепочки, так и гарантировать наличие сильной смысловой связи между соседними узлами.

В рассматриваемой сети хештегов мы отбросили у каждого узла наименее важные связи, которые в сумме дают не более 20% от общей суммы весов среди соседей данного узла. Такая процедура фильтрация приводит к тому, что связи становятся ориентированными. В результате 63,9 млн неориентированных связей превратились в 61,9 млн ориентированных, при этом модификация позволила снизить количество исходящих связей для «хабов» почти на порядок.

Классический подход поиска оптимального пути в сети – поиск кратчайшего пути – широко изучен, однако для построения ассоциативных рядов более ценными является не просто короткие, но короткие и интуитивно понятные пути [1]. В частности, исследование [2] показывает, что в сетях знаний люди обычно находят интуитивно понятные, но не обязательно кратчайшие пути. Согласно результатам данного исследования, кратчайшие пути между двумя вершинами оказываются намного менее понятными, чем более длинные пути, которые находит человек. Интуитивно понятный путь здесь – это путь, в котором логика каждого перехода очевидна человеку, при этом желательно, чтобы путь был относительно коротким.

Данные результаты можно перенести и на задачу, рассматриваемую в исследовании: ассоциативный ряд хештегов не должен иметь больших семантических разрывов и при этом должен быть достаточно коротким. Чтобы построить ассоциативный ряд хештегов, используя сеть совместной встречаемости, необходимо найти такой путь в сети, который удовлетворяет данным критериям. Далее, такие пути будем называть семантическими.

Формально задачу построения семантического пути можно описать следующим образом. Пусть $P = (n_0, n_1, ..., n_{L-1}, n_L)$ – путь длины L между начальной вершиной $a = n_0$ и конечной вершиной $b = n_L$ и имеется возможность вычислять семантическое расстояние между вершинами: $d(n_i, n_{i+1}) = d_{i+1}$.

Определим семантический разрыв пути как максимальное расстояние между соседними вершинами $D(P) = \max\limits_{i \in [0, L-1]} d(n_i, n_{i+1}) = \max\limits_{i \in [0, L-1]} d_i$. С учетом того, что семантический путь не должен содержать большой семантический разрыв и при этом должен быть коротким, задача построения семантического пути ставится как задача многокритериальной оптимизации (1).

$$(1) \qquad \begin{cases} D(P) \xrightarrow[P]{} \min \\ L(P) \xrightarrow[P]{} \min \end{cases}$$

Здесь оба критерия важны, но в то же время они противоречат друг другу: без первого критерия пути могут быть короче, но будут иметь больший семантический разрыв, а без второго критерия узлы в пути могут быть более тесно связаны, но это будет достигаться за счет слишком длинных путей.

Чтобы эффективно решить задачу (1), в рамках исследования был сконструирован объединенный критерий, включающий в себя оба критерия из задачи (1). Для этого были совершены следующие шаги. Во-первых, функция максимума в определении $D(P)$ была заменена функцией сглаженного максимума, то есть такой функцией S_α с параметром α, что $S_\alpha \xrightarrow[\alpha \to \infty]{} \max$. В нашем случае мы

выбрали функцию LogSumExp (LSE): $S_\alpha = LSE_\alpha(d_1,...,d_L) = \frac{1}{\alpha}\log\left(\sum_{i=1}^{L}\exp(\alpha d_i)\right)$. Чтобы учесть второй критерий задачи, был добавлен штраф *d* за каждый дополнительный шаг. С учетом того, что множитель $\frac{1}{\alpha}$ и логарифм не влияют на выбор оптимального пути, новый объединенный критерий имеет вид (2) (здесь γ - параметр масштаба).

$$W_{\alpha,d,\gamma}(P) = \gamma\sum_{i=1}^{L}\left(\exp(\alpha d_i) + d\right) \xrightarrow[P]{} \min \quad (2)$$

В описанных выше критериях предполагается, что имеется возможность вычислять семантическое расстояние d_i между вершинами. При необходимости метрику для узлов сети можно построить, опираясь только на топологию самой сети [3, 4]. Например, алгоритм node2vec [4] при помощи случайного блуждания по сети порождает последовательность упоминаний узлов сети и применяет к ней алгоритм word2vec [5] для получения векторного представления узлов сети. Однако для случая сетей совместной встречаемости векторное представление узлов легко получить напрямую, применив алгоритм word2vec к исходному множеству сообщений, по которым была построена сеть. С помощью библиотеки gensim для собранного корпуса сообщений из Инстаграм нами было построено векторное представление для каждого хештега. Благодаря этому каждый хештег характеризуется 300-мерным вектором, что позволяет, используя косинусную меру, очень просто определять расстояние между любыми двум хештегами в сети.

Параметры функции $W_{\alpha,d,\gamma}$ выбирались следующим образом. Были посчитали расстояния между всеми связанными узлами и выбрали параметр *d*, равный медиане распределения расстояний. Далее случайным образом были выбраны несколько пар хештегов и рассмотрены два типа путей: путь $P_{a,b}^{greedy}$, построенный простым жадным алгоритмом, учитывающим расстояние до конечного узла, и путь $P_{a,b}^{dijkstra\_hops}$, полученный при помощи алгоритма Дейкстры [6], минимизирующего количество переходов. Затем мы выбрали параметры α,γ таким образом, что $W_{\alpha,d,\gamma}(P_{a,b}^{greedy}) < W_{\alpha,d,\gamma}(P_{a,b}^{dijksta\_hops})$ для большинства пар. Такой выбор параметров приводит к тому, что алгоритм Дейкстры с учетом функции $W_{\alpha,d,\gamma}$ начинает предпочитать пути первого типа путям второго типа.

Построение путей с помощью алгоритма Дейкстры, использующего глобальную информацию о сети, может быть вычислительно затратным. Вместо этого можно применить алгоритм, который использует только локальную информацию и функцию эвристической оценки от рассматриваемой вершины до конечной. В рамках исследования была проведена модификация алгоритма из [7], позволившая учитывать текущее расстояние от начального узла и добавившая возможность хранить *k* лучших путей. Этот алгоритм был применен на модифицированной сети совместной встречаемости хештегов с метрикой, порожденной полученным векторным представлением хештегов, и с использованием функции $W_{\alpha,d,\gamma}$. В Таблице 1 представлен пример, демонстрирующий различие кратчайшего пути и семантического пути, построенного с помощью разработанного алгоритма для одной пары начального и конечного хештега.

Таблица 1. Примеры ассоциативных рядов, построенных с помощью глобального кратчайшего пути на графе совместной встречаемости и семантического пути, построенного с помощью жадного локального алгоритма с использованием векторного представления хештегов

Глобальный кратчайший путь	Семантический путь
#эконометрика	#эконометрика
#алматы	#экономика
#лень	#советыпредпринимателя
	#таймменеджмент
	#лень

Таким образом, в результате исследования была дана формальная постановка задачи построения семантического пути и предложен оригинальный подход для поиска вычислительно эффективного

решения задачи построения ассоциативного ряда на основе совместного использования техники получения векторного представления слов и сети совместной встречаемости терминов, приведен пример работы алгоритма для набора данных из социальной сети Инстаграм.

Литература

1. He L. et al. Neurally-Guided Semantic Navigation in Knowledge Graph. // IEEE Transactions on Big Data. 2018.

2. West R., Pineau J., Precup D. Wikispeedia: an online game for inferring semantic distances between concepts. // IJCAI, 2009. – P.1598-1603.

3. Goyal P., Ferrara E. Graph Embedding Techniques, Applications, and Performance: A Survey. // Knowl.-Based Syst. 2018, №151. – P.78-94.

4. Grover A., Leskovec J.. node2vec: Scalable feature learning for networks // Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, ACM. 2016. – P.855–864.

5. Mikolov, T., Chen, K., Corrado, G. K., Dean, J. Efficient estimation of word representations in vector space // CoRR, abs/1301.3781 (2013)

6. Dijkstra, E. A note on two problems in connexion with graphs // Numerische Mathematik 1(1), 269–271 (1959). doi: 10.1007/BF01386390

7. Capitan J.A. et al. Local-Based Semantic Navigation on a Networked Representation of Information // PLoS ONE 7(8): e43694 / 2012 *Sdf*