Петрова А. Н., Фролов Д. О.
A. N. Petrova, D. O. Frolov

РАЗРАБОТКА НЕЙРОННЫХ СЕТЕЙ ДЛЯ СКВОЗНОГО ОБУЧЕНИЯ МОДЕЛЕЙ ПОИСКА ИНФОРМАЦИИ

DEVELOPMENT OF NEURAL NETWORKS FOR END-TO-END TRAINING OF INFORMATION RETRIEVAL MODELS

Петрова Анна Николаевна – кандидат технических наук, заведующая кафедрой «Проектирование, управление и развитие информационных систем» Комсомольского-на-Амуре государственного университета (Россия, Комсомольск-на-Амуре). E-mail: PetrovaAN2006@yandex.ru.
Anna N. Petrova – PhD in Engineering, Head of Design, Management and Development of Information Systems Department, Komsomolsk-na-Amure State University (Russia, Komsomolsk-on-Amur). E-mail: PetrovaAN2006@yandex.ru.

Фролов Дмитрий Олегович – аспирант Комсомольского-на-Амуре государственного университета (Россия, Комсомольск-на-Амуре). E-mail: optcompanys@mail.ru.
Dmitriy O. Frolov – Graduate Student, Komsomolsk-na-Amure State University (Russia, Komsomolsk-on-Amur). E-mail: optcompanys@mail.ru.

Аннотация. Нейронные сети для сквозного обучения моделей поиска информации довольно часто применяются в крупных компаниях. Эти сети отличаются различными аспектами, включая архитектуру, данные обучения, представления данных и функции потерь. Однако при обучении сквозных моделей нейронного ранжирования без функций, созданных человеком, используются только точечные и парные функции потерь. Эти функции потерь не учитывают ранги документов при оценке потерь обучающих данных. В результате традиционные модели обучения ранжированию, использующие функции точечных или парных потерь, обычно показывают более низкую производительность по сравнению с моделями, использующими функции списочных потерь. Исходя из этого наблюдения предлагается использовать функции потерь по спискам для обучения моделей нейронного ранжирования. Разработано несколько нейронных сетей для сквозного обучения моделей поиска информации. В результате выполнения работы было доказано, что списочный нейронный ранжировщик превосходит модель парного нейронного ранжирования. Кроме того, были достигнуты улучшения производительности списочных моделей нейронного ранжирования за счёт выборки обучающих данных на основе запросов.

Summary. Neural networks for end-to-end learning of information retrieval models are quite common in large companies. These networks differ in various aspects including architecture, training data, data representations, and loss functions. However, when training end-to-end neural ranking models without human-generated features, only point and pairwise loss functions are used. These loss functions do not consider document ranks when estimating the loss of training data. As a result, traditional ranking learning models using point or pairwise loss functions usually show poorer performance compared to models using list loss functions. Based on this observation, it is proposed to use list loss functions to train neural ranking models. Several neural networks have been developed for end-to-end training of information retrieval models. Because of the work, it was proved that the list-based neural ranker outperforms the pairwise neural ranking model. In addition, performance improvements of list-based neural ranking models were achieved by query-based training data sampling.

Ключевые слова: концепция нейронных сетей для ранжирования документов, использование функции потерь по спискам, оптимизация обучающих данных на основе запросов.

Key words: concept of neural networks for ranking documents, using the list loss function, optimizing training data based on queries.

УДК 517.95

Введение. Модели нейронных сетей широко применяются в различных областях, включая поиск информации, благодаря их способности автоматически обучать функции на разных уровнях абстракции. Отличительной особенностью глубоких нейронных сетей является возможность со-

здания собственных функций или представлений данных без необходимости вручную определять их, как это делается в традиционных моделях ранжирования. Одной из основных концепций при использовании нейронных сетей является представление запросов и документов на основе различных признаков, а затем обучение алгоритмов на этих представлениях, например с помощью методов градиентного спуска и списочного перехода. В отличие от традиционных моделей обучения ранжированию, которые работают с необработанными данными, такими как текст, нейронные модели ранжирования используют созданные представления текста для обучения. Например, алгоритм списочного подхода может использовать нейронную сеть для обучения функции ранжирования, но при этом считается традиционным алгоритмом ранжирования из-за использования предварительно созданных представлений запросов и документов.

Параметры моделей обучения ранжированию настраиваются в соответствии с выбранной функцией потерь. Далее эти модели обычно классифицируются как точечные, парные и списочные подходы в зависимости от применяемых функций потерь. В то время как модели точечного и парного обучения рассматривают проблему ранжирования как задачу классификации, списочный подход к обучению ранжированию анализирует модель ранжирования более естественным образом. В результате традиционные алгоритмы обучения ранжированию, использующие списочные функции потерь, продемонстрировали более высокую эффективность по сравнению с точечными и парными алгоритмами на большинстве наборов данных, особенно на документах с высоким рейтингом. Это обусловлено в основном тем, что функции потерь точечных и парных алгоритмов не учитывают порядок документов в окончательных ранжированных списках и, следовательно, не соответствуют типичным метрикам оценки информационного поиска.

Хотя традиционные модели обучения ранжированию с использованием списочных функций потерь показали многообещающие результаты, ни одна из существующих нейронных моделей ранжирования, обучаемых полностью на входных данных без использования предварительно разработанных функций, не использует списочную функцию потерь для обучения. Поэтому в ходе данной работы исследуется потенциал использования списочных функций потерь для обучения дискриминационных моделей нейронного ранжирования. Конкретно рассматривается модель сопоставления с глубокой релевантностью, которая использует комбинацию попарной шарнирной потери и списочной функции потерь для определения возможных различий между изученными моделями ранжирования.

Несмотря на то что списочные функции потерь соответствуют сущности проблем ранжирования, обучение нейронных моделей ранжирования с использованием таких функций является более сложным процессом. При наличии набора размеченных данных алгоритмы точечного или парного обучения могут иметь больше обучающих примеров по сравнению с алгоритмами, использующими списочные функции. Однако нейронные сети требуют обучения на достаточно больших объёмах данных. Для решения этой проблемы предлагается случайная выборка документов, связанных с каждым запросом, перед каждой эпохой обучения. Оценка показывает, что перетасовка и выборка улучшают производительность списочного нейронного ранжирования по двум причинам: во-первых, создаётся немного изменённый набор обучающих примеров для каждой эпохи обучения, что помогает сети избежать запоминания обучающих данных; во-вторых, обеспечивается более плоский минимум функции потерь, что способствует лучшему обобщению обученной модели на тестовых данных.

Нейронная модель ранжирования. Сетевая архитектура. Использовалась модель глубокого соответствия релевантности (DRMM) для специфического поиска документов, направленного на оценку их релевантности. Эта модель работает путём создания гистограмм точного и семантического соответствия между запросом и документом фиксированного размера, которые затем подаются на вход нейронной сети. Далее используется предсказание для нелинейного сопоставления между терминами запроса и документа. В завершение для оценки каждого документа применяется сеть-шлюз, которая определяет важность каждого термина запроса. Обучение сети проводится с использованием функции попарных потерь:

$$L(q, d^+, d^-;\ \theta) = \max\big(0{,}1 - s(q, d^+) + s(q, d^-)\big),$$

где d^+ и d^- являются документами, релевантными и нерелевантными запросу q соответственно, а θ указывает параметры модели.

Функция потерь по спискам. Была использована функция потерь в рамках метода списочного подхода для обучения модели нейронного ранжирования. Эта функция потерь опирается на вероятностное распределение для списка оценённых документов, что указывает на вероятность различных позиций документов в рейтинге. Распределение вероятностей может быть оценено с помощью перестановок:

$$p(d_j) = \frac{\exp(s(q, d_j))}{\sum_{k=1}^{n} exp\,(s(q, d_k))},$$

где из-за вычислительной сложности перестановочных вероятностей мы используем вероятности топ-1, следуя первоначальной модели и её последующему применению.

Истинное распределение вероятностей $y^{(i)}$ оценивается с использованием человеческих суждений о релевантности. Затем перекрёстная энтропия используется для измерения расстояния между двумя распределениями вероятностей, которое оценивается на основе прогнозируемых оценок для документов ($z^{(i)}$) и на основе суждений о релевантности как

$$L\big(y^{(i)}, z^{(i)};\ \theta\big) = -\sum_{j=1}^{n} p_{y^{(i)}}(d_j) \log p_{z^{(i)}}(d_j).$$

Образцы обучения. При равном объёме размеченных данных алгоритмы обучения ранжированию с точечными или парными функциями потерь имеют больше обучающих примеров, чем алгоритмы с использованием списочных функций потерь. Это обусловлено тем, что каждый документ или каждая пара документов могут быть использованы как обучающий пример в первом случае, в то время как во втором каждый запрос является единственным обучающим примером. Однако для успешного обучения нейронных сетей требуется большой объём данных. Для компенсации ограниченного объёма обучающих данных для моделей списочного нейронного ранжирования мы предлагаем случайную выборку документов для каждого запроса перед каждой новой эпохой обучения. Этот подход не увеличивает количество обучающих примеров, но обновляет сеть новыми данными, что помогает избежать запоминания данных. Использование различных данных способствует более эффективному обучению модели ранжирования, предотвращая переобучение. Глубокие нейронные сети могут быстро запоминать обучающие данные, поэтому избегание этого помогает улучшить качество обученной модели.

Эксперимент. Для проведения экспериментов используются два набора данных:

1. набор данных Robust04, содержащий более 600 тысяч информационных материалов, использованных в TREC Robust Track 2004, и включающий 400 тем;

2. набор данных ClueWeb09 Category B, который включает более 60 миллионов информационных страниц и применялся в TRECWeb.

Используются названия тем в качестве запросов. Для каждого запроса извлекаются первые 1950 документов в качестве обучающих данных или документов-кандидатов, которые будут повторно ранжированы обученными моделями во время вывода. В экспериментах применяется модель правдоподобия запроса с параметрами по умолчанию для получения исходных документов. Слова формируются с использованием стеммера. Доступные помеченные данные разделяются на обучающие, проверочные и тестовые наборы случайным образом на 40, 30 и 30 % соответственно. Результаты тестовых наборов затем оцениваются с помощью средней точности среднего значения MAP, точности верхних *k* документов P@k и нормализованного дисконтированного совокупного выигрыша, рассчитанного для первых *k* документов nDCG@k, где *k* равно 1, 3, 5 и 10. Статистически значимые тесты выполняются с использованием двустороннего парного *t*-критерия на уровне значимости 0,05.

Экспериментальная настройка проводится с использованием публичной реализации модели DRMM. Входные данные для сети формируются с использованием предварительно обученных вложений слов размером 300 из глобального векторного представления слов. В экспериментах параметры модели оптимизируются с помощью оптимизатора AdaDelta и алгоритма обратного распространения ошибки для вычисления градиентов. Все гиперпараметры модели подбираются в наборе проверки. Для каждой модели количество скрытых слоёв и их размеры выбираются из диапазонов {2,3} и {50,100,150,200} соответственно. Начальная скорость обучения выбирается из множества {0.2,0.4,0.6,0.8,1}.

Для использования функции потерь по списку мы проводим перетасовку запросов и связанных с ними документов в обучающих данных перед каждой эпохой обучения. Затем случайным образом выбирается x из 2000 доступных документов для каждого запроса, где x выбирается из {500, 1000, 1500}, чтобы получить обучающие данные. Для тестовых запросов все 1950 документов используются для ранжирования, поэтому выборка во время вывода не выполняется.

Сравниваются три модели нейронного ранжирования:

1. DRMMpl – это исходная модель DRMM, которая использует функцию парных потерь;
2. DRMM11 – это модель DRMM, обученная с использованием функции списочных потерь в уравнении;
3. DRMM11-WS – это модель DRMM, обученная с использованием функции списочных потерь, но без случайной выборки документов.

Поскольку цель не заключается в достижении максимальной производительности с использованием списочных функций потерь, обучение модели DRMM с использованием функций потерь традиционных алгоритмов LambdaMart или LambdaRank остаётся без изменений, как показано на рис. 1.

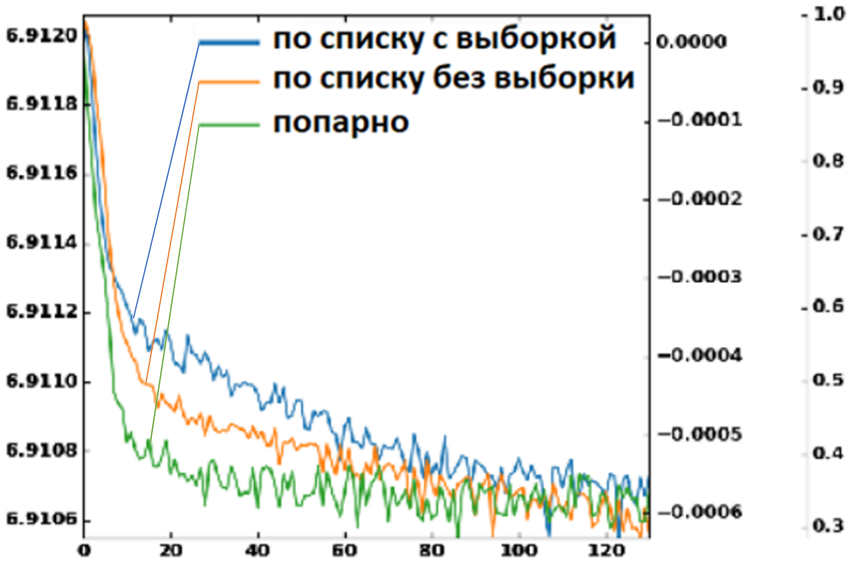

Рис. 1. Обучение модели DRMM

Заключение. Продемонстрировано, как использование функций списочных потерь способствует улучшению производительности поиска в моделях нейронного ранжирования. Более подробно было исследовано, как обучение модели сопоставления с глубокой релевантностью, представляющей собой парную модель с использованием списочной функции потерь, влияет на производительность поиска. Было выявлено, что перетасовка и случайная выборка документов, связанных с каждым запросом, перед каждой эпохой обучения приводят к улучшению производительности поиска. В дальнейшем исследовании могут быть рассмотрены различные подходы к выборке, т. к. было показано, что эта стратегия способствует повышению эффективности поиска. Важным

направлением для будущей работы является изучение влияния других функций списочных потерь на производительность моделей нейронного ранжирования.

ЛИТЕРАТУРА

1. Nazarov, S. V. Architecture and design of software systems: monograph / S. V. Nazarov. – 2nd ed., revised. – M.: INFRA-M, 2018. – 374 p. // ZNANIUM.COM: electronic library system. – Access mode: http://znanium.eom/catalog.php#, restricted. – Zagl. from the screen.
2. Huang, P. S., He, X., Gao, J., Deng, L., Acero, A., & Heck, L. (2013). Learning deep structured semantic models for web search using clickthrough data. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management (p. 2333-2338).
3. Shen, Y., He, X., Gao, J., Deng, L., & Mesnil, G. (2014). Learning semantic representations using convolutional neural networks for web search. In Proceedings of the 23rd International Conference on World Wide Web (p. 373-374).
4. Grbovic, M., Djuric, N., Radosavljevic, V., Silvestri, F., & Bhamidipati, N. (2018). Context-aware event recommendation in event-based social networks. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (p. 235-244).
5. Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (p. 1412-1421).
6. Dai, Z., Yang, Z., Yang, F., Cohen, W. W., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860.
7. Guo, J., Fan, Y., Ai, Q., Croft, W. B., & Yates, A. (2016). A deep relevance matching model for ad-hoc retrieval. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (p. 55-64).
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (p. 5998-6008).
9. Guo, J., Fan, Y., Ji, X., & Croft, W. B. (2017). A deep relevance ranking approach for ad-hoc retrieval. In Proceedings of the 26th International Conference on World Wide Web (p. 551-560).