Сравнение плоского и локального подходов к иерархической классификации текстов

А. А. Сайгин[1], С. А. Федосин[2]

Национальный исследовательский Мордовский государственный университет им. Н. П. Огарёва

[1]andrexsai@mail.ru, [2]fedosinsa@mrsu.ru

Аннотация. **В статье формируется задача иерархической классификации текстов, описываются подходы к иерархической классификации и метрики оценки их работы, подробно рассматриваются плоский и локальный подходы к иерархической классификации, проводится серия экспериментов по обучению иерархических классификаторов с различными методами векторизации, сравниваются результаты оценки работы обученных классификаторов.**

Ключевые слова обработка естественного языка; классификация; иерархическая классификация; плоская классификация; локальная классификация; векторизация

I. Введение

Классификация представляет собой процесс разделения множества объектов на группы в соответствии с определённым признаком или критерием. Она активно используется в различных сферах деятельности, из-за чего существует потребность в постоянном определении классов объектов [1].

Некоторые задачи классификации в различных прикладных областях можно рассматривать как задачи иерархической классификации. Они имеют место для объектов, организованных в иерархическую структуру классов. В данной структуре между классами можно построить отношения, в которых одни классы будут являться подклассами других. Визуально иерархию можно представить в виде графа или дерева (рис. 1).

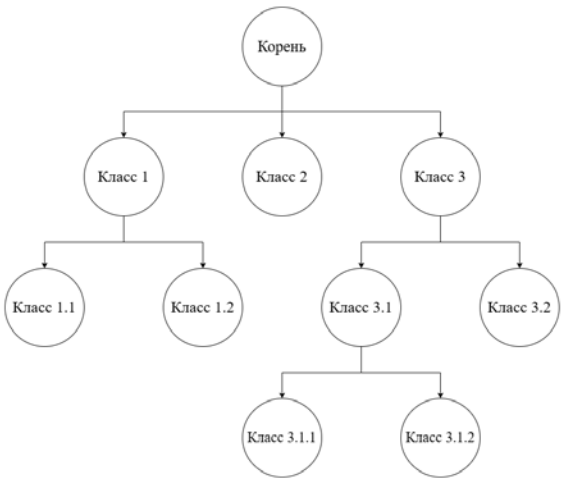

Рис. 1. Пример иерархии классов

Примерами таких задач являются прогнозирование функции белка, категоризация текста, классификация музыкальных жанров. По этой причине растёт число исследований, посвящённых разработке новых алгоритмов, способных генерировать модели классификации для задач иерархической классификации [2].

Классифицировать объекты становится значительно сложнее, когда речь идёт об иерархической структуре, поскольку итоговый результат может относиться как к конечным категориям, так и к промежуточным уровням. В связи с этим при построении системы классификации необходимо дополнительно анализировать взаимосвязи между различными категориями.

II. Подходы к иерархической классификации

Существующие подходы к иерархической классификации можно разделить на подходы плоской классификации, подходы иерархической классификации на основе локальных моделей и подходы иерархической классификации на основе глобальных моделей.

В подходе плоской классификации (Flat Classification, FC) рассматриваются только конечные классы иерархии. Фактически, обучается обычный алгоритм классификации. Информация о промежуточных классах иерархии игнорируются [3].

Подход локальной классификации используются несколько классификаторов, каждый из которых работает на своем уровне иерархии. Существует несколько видов локальной классификации: локальная классификация для вершины (Local Classifier per Node, LCN), локальная классификация для родительской вершины (Local Classifier per Parent Node, LCPN) и локальная классификация для уровня (Local Classifier per Level, LCL). В LCN в каждом узле создаются отдельные классификаторы, которые определяют принадлежность данных к текущему узлу. В LCPN обучаются классификаторы для каждого родительского узла, которых предсказывают дочерний класс по отношению к текущему. В LCL обучаются классификаторы для каждого уровня иерархии [3].

В подходе глобальной классификации обучается единая модель классификации, которая анализирует всю существующую иерархию. Прогнозирование происходит на любом уровне иерархии [3].

III. Описание эксперимента

В данном исследовании предлагается сравнить плоский и локальные подходы к иерархической классификации на примере задачи классификации текстов.

Для классификации иерархически распределенных текстов необходимо выполнить предобработку текстовых данных, представляющую из себя вычистку текста, избавление от стоп-слов, стэмминг, лемматизацию, и векторизацию, после выполнить обучение классифицирующей модели.

A. *Модели векторизации*

Важным этапом предобработки текста является векторизация. Это процесс преобразования текстовых данных в числовые векторы-эмбеддинги. Данный этап необходим для удобства обработки текста вычислительной техникой, которая может оперировать только числовыми данными. При этом итоговый вектор должен учитывать, что слова могут употребляться с разным значением в зависимости от контекста.

Для преобразования текстов в числовые векторы разработаны разные алгоритмы векторизации, в том числе на основе методов машинного обучения. В данном исследовании были использованы алгоритмы FastText, USE и BERT. FastText – метод векторного представления слов на базе нейронной сети прямого распространения, использующий символьные n-граммы [4]. USE – это модель, основанная на глубокой нейронной сети, с использованием двунаправленного кодера и декодера с механизмом внимания [5]. BERT – это модель, представляющая собой набор последовательно соединённых энкодеров Transformer с механизм внимания [6].

B. *Архитектуры классификаторов*

В роли классификатора используется нейронная сеть прямого распространения. Она содержит три слоя. Во входном слое количество нейронов зависит от метода векторизации, поскольку каждый алгоритм на выходе дает векторы разной длины. Скрытый слой содержит 512 нейронов и функцию активации ReLU. Выходной слой выдает предсказания классов, поэтому количество нейронов равно количеству классов в обучающей выборке. К выходу модели применяется функция LogSoftmax. Функции потерь при обучении – CrossEntropyLoss, оптимизатор – AdamW, количество эпох – 100. Пример архитектуры изображен на рис. 2.

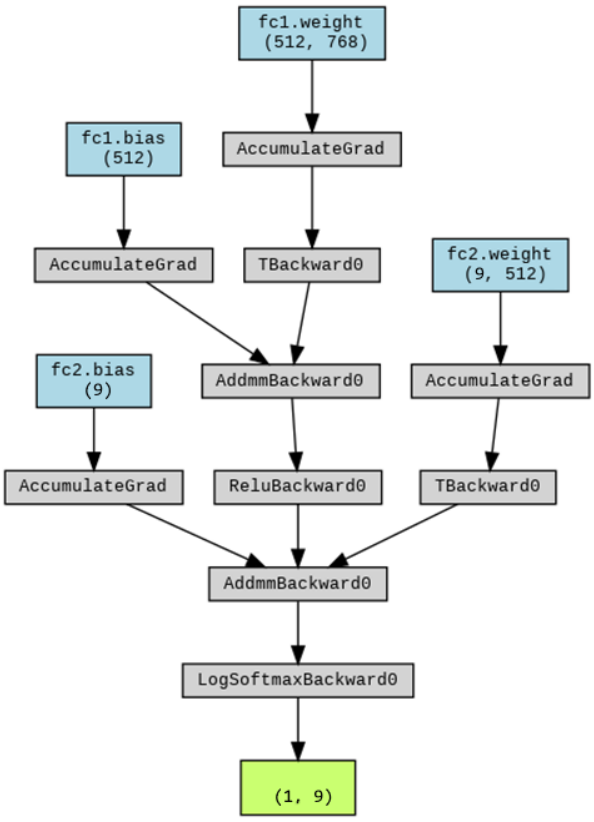

Рис. 2. Пример архитектуры классификатора

На рис. 2. изображена нейронная сеть, которая используется в локальной классификации по уровням на первом уровне иерархии. Базовым векторизатором в данном примере является BERT, размер вектора которого равен 768. Количество выходов равно 9, поскольку столько классов находится на первом уровне иерархии выбранного набора данных.

Описанная нейронная сеть используется в плоском подходе как классификатор и в локальном подходе в каждом узле иерархии.

C. *Данные*

Обучение производилось на наборе данных DBPedia Classes. Это набор структурированных данных, извлеченный из Википедии и публикации её в формате, пригодном для машинной обработки. Датасет предоставляет таксономические иерархические категории для 342782 статей Википедии, распределенных по трем уровням: по 9, 70 и 219 классов соответственно. Этот набор данных является популярной базой для задач обработки естественного языка и классификации текстов [7].

D. *Метрики*

Для оценки точности иерархической классификации используются отдельные метрики. Это связано с тем, что необходимо оценить точность предсказания на всех уровнях иерархии, так как конечный класс может быть предсказан неверно, но при этом иметь общего родителя с истинным классом. Такими метриками являются иерархическая точность, иерархическая полнота и иерархическая F-мера, которые вычисляются по следующим формулам:

$$hR = \frac{|A(C_t) \cap A(C_p)|}{A(C_t)}$$

$$hP = \frac{|A(C_t) \cap A(C_p)|}{A(C_p)}$$

$$hF = \frac{2 * hP * hR}{hP + hR}$$

где $A(C_t)$ – множество истинных классов, к которым относится объект в иерархии, $A(C_p)$ – множество предсказанных классов, к которым относится объект в иерархии [8].

Используем для оценки иерархическую F-меру и сравним ее с обычной F-мерой.

IV. Результаты

Результаты обучения моделей представлены в табл. 1.

ТАБЛИЦА I. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ КЛАССИФИКАТОРОВ

Классификатор	**FastText**		**USE**		**BERT**	
	F	*hF*	*F*	*hF*	*F*	*hF*
FC	0,91	0,94	0,91	0,94	0,90	0,94
LCN	0,89	0,92	0,89	0,93	0,89	0,93
LCPN	0,92	0,94	0,90	0,93	0,90	0,94
LCL	0,90	0,93	0,90	0,93	0,89	0,93

Наилучшие результаты показали плоский подход и локальный подход для родительской вершины на базе

векторизатора FastText. По F-мере видна высокая точность итоговой классификации, иерархическая F-мера показывает, что у части объектов были верно определены вышестоящие в иерархии классы. Худшие результаты показал локальный подход для вершины, н даже они находятся на достаточно высоком уровне.

V. Заключение

Все подходы показали примерно одинаковую высокую точность классификации. Из-за того, что плоский подход проще и быстрее обучить и размер итоговой системы намного меньше в сравнении с локальным подходом, то можно предположить, что целесообразнее на практике использовать его. В нашем примере в классификации для вершин обучалось 298 моделей, в классификации для родительских вершин – 80 моделей, а в классификации для уровней – 3 модели. В то время как для плоской классификации требуется всего 1 модель. Тем не менее, локальный подход обладает большей гибкостью. Это значит, что в ситуациях, когда настройка параметров при плоском подходе не дает требуемых результатов, то можно прибегнуть к локальному подходу, в котором можно настраивать классификацию на каждом отдельном узле, за счет этого повышая точность всей системы. Также, за счет этого можно регулировать размер итогового набора моделей.

Дальнейшие исследования могут быть связаны со сравнением описанных подходов с глобальным подходом к иерархической классификации.

Список литературы

[1] Айвазян С.А., Бухштабер В.М., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: классификация и снижение размерности. М. : Финансы и статистика, 1989. 607 с.

[2] Silla C.N., Freitas A.A. A survey of hierarchical classification across different application domains // Data mining and knowledge discovery. 2011. Т. 22. С. 31-72.

[3] Borges H.B., Silla JrC.N., Nievola J.C. An evaluation of global-model hierarchical classification algorithms for hierarchical classification problems with single path of labels // Computers & Mathematics with Applications. 2013. Т. 66. №. 10. С. 1991-2002.F

[4] Grave E., Bojanowski P., Gupta P., Joulin A., Mikolov T. Learning word vectors for 157 languages // arXiv preprint arXiv:1802.06893. – 2018. URL: arxiv.org/abs/1802.06893 (дата обращения: 26 июля 2025).

[5] Cer D., Yang Y., Kong S.Y., Hua N., Limtiaco N., John R.S., Constant N., Guajardo-Cespedes M., Yuan S., Tar C., Sung Y.-H., Strope B., Kurzweil R. Universal sentence encoder // arXiv preprint arXiv:1803.11175. 2018. URL: arxiv.org/abs/1803.11175 (дата обращения: 26 июля 2025).

[6] Devlin J., Chang M.W., Lee K., Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding // arXiv preprint arXiv:1810.04805. – 2018. URL: arxiv.org/abs/1810.04805 (дата обращения: 26 июля 2025).

[7] DBPedia Classes: Kaggle: сайт – 2010. URL : https://www.kaggle.com/ datasets/danofer/dbpedia-classes (дата обращения: 26.07.2025). – Режим доступа: свободный.

[8] Kiritchenko S. et al. Learning and evaluation in the presence of class hierarchies: Application to text categorization // Conference of the Canadian society for computational studies of intelligence. Berlin, Heidelberg : Springer Berlin Heidelberg, 2006. С. 395-406.