О многоклассовой классификации слов рекуррентной нейронной сетью с памятью (LSTM) применительно к задаче распознавания именованных сущностей

В.С. Вакурин[1], А.В. Копылов[1], О.С. Середин[1], К.С. Мерцалов[2]

[1]Тульский государственный университет, Ленина 92, Тула, Россия, 300012
[2]Rensselaer Polytechnic Institute, Troy, NY, USA

Аннотация. В работе исследуются вопросы обучения нейронных сетей с обратным распространением ошибки распознаванию именованных сущностей при использовании многослойных архитектур нейронных сетей и различных признаковых пространств, образованных на символьных цепочках. В статье приводятся результаты экспериментов, показывающих зависимость прогностических свойств от пересечения множества именованных сущностей между обучающим и тестовым набором при стандартной постановке задачи поиска именованных сущностей. Также предлагается способ улучшения прогностических свойств моделей для обнаружения именованных сущностей, ранее не предъявленных при обучении.

1. Введение

В данной работе предлагается новый метод, проводится исследование и уточняется природа основных недостатков решений по распознаванию именованных сущностей (Named Entities, далее по тексту статьи – NE) в тексте. Распознавание именованных сущностей является известной задачей, относящейся к одной из технологий интеллектуального анализа текстов на естественном языке (Text Minig) [1].

Распознавание именованных сущностей служит составной частью решения важной проблемы Text Mining – поиска и выделения единых информационных объектов, явно или косвенно указанных по тексту. Общая задача распознавания именованных сущностей (Named Entity Recognition, NER) заключается в выделении слов или последовательностей слов в тексте, принадлежащих заданной специфической группе слов, например таких как наименования организаций, географические названия, имена собственные и др., и включает в себя множество конкретных постановок, имеющих важное значение в системах автоматизированной обработки текстовой информации. В литературе обычно упоминается распознавание имён собственных, распознавание назначенных пациенту лекарственных препаратов (bio-NER, drug-NER) [2], распознавание специфической терминологии - химических формул (chem-NER) [3]. Ввиду сложности составления синтаксических правил для таких задач, сложности составления словаря и частые ошибки в написании имён собственных и формул такие задачи обычно решаются с помощью машинного обучения [3]. Современные способы распознавание именованных сущностей в течение последних трёх-четырёх лет пополнились существенным арсеналом новых способов, основанных на применении новых и новейших архитектур нейронных сетей с использованием долгой краткосрочной памяти [4] и стали темой

исследования для многих коллективов. В отечественной литературе использование такой архитектуры нейронной сети описывается в работе [5].

Часто используемым методом оптимизации для обучения нейронной сети является метод стохастического градиента (stohastic gradient descend, SGD) [6] который итеративно управляется числовым значением - величиной функции потерь [7]. С одной стороны, этот метод в своей основе имеет случайное распределение вносимых в коэффициенты нейронной сети изменений, в том смысле, что вектор параметров модели случайно колеблется вокруг общей траектории поскольку обновляется при каждом предъявлении нового объекта (с шумом относительно обобщенного портрета, “on-line update” см. источник [22]) и за счёт этого предполагаемый глобальный минимум ошибки находится быстрее [22]. С другой стороны, значения вектора истинных значений (ground truth) и алгоритм вычисления функции потерь должен быть адекватен задаче обучения.

Целью настоящей работы является апробация предложенного нами метода использования многоклассовой функции потерь совместно с вероятностным представлением цепочек конкретных именованных сущностей с целью улучшения обученности моделей для распознавания именованных сущностей, не предъявляемых при обучении.

В настоящей статье мы предлагаем способ улучшить качество моделей для обнаружения NE, ранее не предъявленных при обучении. Также мы приводим результаты экспериментов, показывающих зависимость прогностических свойств от пересечения множества именованных сущностей между обучающим и тестовым набором при стандартной постановке задачи поиска NE, крайне слабую предиктивную способность таких, стандартно обученных моделей, при применении к текстам содержащим новые, еще неизвестные NE, что часто случается при практическом (промышленном) применении этой задачи распознавания.

2. Сопутствующие работы

Известны несколько основных подходов к определению именованных сущностей: на основе грамматических шаблонов [8]; с использованием классификатора, построенного при применении метода опорных векторов [9]; с использованием статистических моделей - скрытых марковских моделей [10] условных случайных полей [11, 12], и различных моделей глубокого обучения на основе нейронных сетей [13-16]. Выявленные ограничения использования рекуррентных нейронных сетей для предсказания последовательности [13] послужили поводом для появления [4] ячеек нейронной сети с регулируемым запоминанием и горизонтальными связями т.н. долгой краткосрочной памяти (Long short-term memory, LSTM). Тенденция последних лет в этой области - комбинирование различных архитектур нейронных сетей, как слоёв более общей, многослойной нейронной сети, то, что в последнее время понимается в качестве “deep learning”. Такой принцип описан в работе [14], первые результаты для свёрточных сетей описаны в [15], применительно к современным архитектурам нейронных сетей в [16].

Несмотря на неплохие достигнутые и сравнимые с более традиционными методами показателями качества в задаче NER, исследователи отмечают и недостаток, который связан со случайным характером искажений значений признаков поступающего на вход системы объекта. Как отмечено в [17], слабо улучшают ситуацию и расширения признакового пространства за счёт признаков заглавных букв, найденных частей речи. Решением, выводящем нейронные сети с памятью на уровень “the state of the art” являются архитектуры, не требующие вручную составленных дополнительных признаков (Feature Engineering), предобработки, а использующие сквозные (end-to-end) архитектуры, базирующиеся на обработке символьных последовательностей непосредственно, и формирующих для верхних слоёв вида LSTM, распознающих цепочку (с наличием NE), признаковое пространство достаточной размерности [17, 18]. В пользу такого подхода свидетельствует и работа [19], в которой отмечается, что признаковое пространство, формируемое такой моделью, позволяет различать для каждого слова суффиксы, написание слов с заглавной буквы, префиксы и автоматически производить токенизацию. При таком подходе к проблеме задача обучения представляется аналогичной обучению человека словам - производится сопоставление непосредственного изображения

цепочек символов и проверочного, скрытого от непосредственного наблюдения обучаемым, списка слов: абстрактного, неочевидного для учащегося на первых этапах обучения, но по окончании обучения содержащего множество элементов-слов и закономерности последовательного употребления этих элементов. В настоящей работе мы экспериментально выясним, насколько состоятелен такой подход. Также мы экспериментально проверим и регулируемое вертикальное наращивание слоёв нейронной сети. В условиях такой, архитектурно обусловленной "глубины", возникла проблема зависимости о представлении линейного оператора для множества слоёв нейронной сети (применяемого к слоям нейронной сети как к элементам - так, как бы он применяется для элементов конкретного слоя нейронной сети в обычной постановке). В таких архитектурах это было успешно решено в [20, 21] и привело к появлению нейронных сетей с виртуальной (переменной) глубиной (Highway Networks).

3. Общая архитектура предлагаемой нейронной сети

3.1. Архитектура кодировщика

Формирование признакового представления выполняется на свёрточном кодировщике [22] на вход которого предъявляются закодированные натуральными числами признаки букв [18]. Каждое слово кодируется вектором некоторой длины самого длинного слова алфавита l (в нашем случае - 21 буква), элементами этого вектора являются числа - номер буквы в алфавите, пустое знакоместо кодируется цифрой 1.

Как указано в [18], в задачах обработки текстов на естественном языке используется свёртка по последовательности (обычно употребляется термин свёртка по времени), а не пространственная свертка как в случае изображений, поэтому признаковое описание $\mathbf{f}^k \in R^{l-w+1}$ промежуточного слоя нейронной сети для слова k будет образовано следующим образом:

$$\mathbf{f}^k[i] = \tanh\left(\left\langle \mathbf{C}^k[*, i:i+w-1], \mathbf{H}\right\rangle + b\right), i = 0...1000,$$

где $\mathbf{C}^k[*, i:i+w-1]$ представляет собой столбцы матрицы $\mathbf{C}^k$ от i до $i+w-1$, $\left\langle \mathbf{A}, \mathbf{B}\right\rangle = \mathrm{Tr}\left(\mathbf{A}\mathbf{B}^T\right)$ скалярное произведение Фробениуса.

Из вектора признаков $\mathbf{f}^k$ необходимо отобрать наиболее важные признаки для каждого слова k: $\mathbf{y}^k = \max_i \mathbf{f}^k[i]$ (max-over-time) для k, находящегося в центре окна из букв шириной w [18].

Наиболее эффективным способом представить сформированные из последовательности символов n-gram для сверточной сети можно с помощью одновременного использования нескольких таких фильтров, ширина которых будет различной и будет пропорциональна длине ожидаемого n-gram (слова в символах). Мы использовали те же параметры, что и в указанной статье [18] семь фильтров размерностью [50, 100, 150, 200, 200, 200, 200]. Как указывают авторы - основная идея в том, чтобы зафиксировать наиболее важные признаки для данного входа n-gram для каждого фильтра разной размерности.

Для фильтров $\mathbf{H}_1, \mathrm{K}, \mathbf{H}_h$ (в нашем случае $h = 7$), выход сверточной нейронной сети для символьного представления будет $\mathbf{y}^k = \left[y_1^k, \mathrm{K}, y_h^k\right]$ - для входного представления слова k максимальной длинной 21. Как указано в цитируемой статье [18], для многих приложениях обработки текстов на естественном языке h - размерность выходного промежуточного слоя обычно выбираются так, чтобы быть в диапазоне от 100 до 1000. В нашем эксперименте это значение было выбрано равным 650.

При предъявлении новых предложений в окне обучения длинной 100 предложений возможно проявление внутреннего сдвига переменных (internal covariance shift [22]). Для минимизации этого эффекта и ускорения обучения мы будем использовать нормализацию по порциям данных обучения (mini-batch normalization) [23].

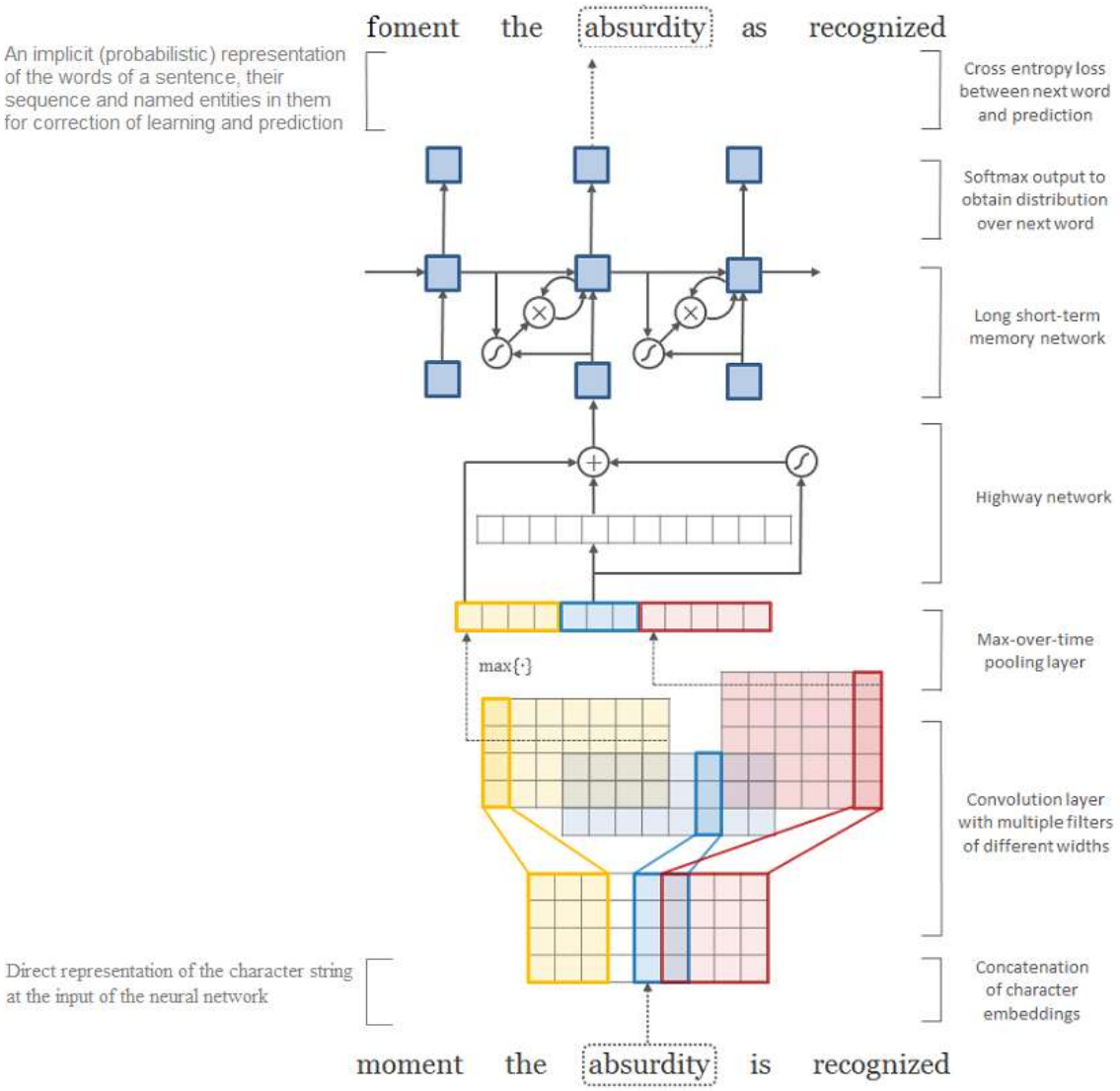

Рисунок 1. Общая схема char-cnn-lstm кодировщика, за основу взята схема представлена в статье [19].

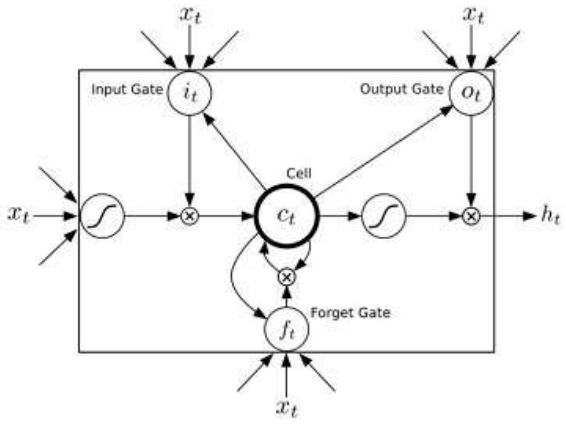

Рисунок 2. Структурная схема ячейки долгосрочно-краткосрочной памяти, рисунок взят из источника [4].

Выход свёрточного кодировщика после нормализации может быть дополнен слоями с линейными передаточными функциями с функцией динамического обхода - исключения нескольких линейных слоёв по значению функции G [20, 21]:

$$\mathbf{y} = H(\mathbf{x}, \mathbf{W}_H) \cdot G(\mathbf{x}, \mathbf{W}_G) + x \cdot (1 - G(\mathbf{x}, \mathbf{W}_G)),$$

где x - вход, $H(\mathbf{x}, \mathbf{W}_H)$ - функция обработки (transform gate), $G(\mathbf{x}, \mathbf{W}_G)$ - функция обхода (carry gate): $H(\mathbf{x}) = (\mathbf{W}_H \mathbf{x} + b_H)$, $G(\mathbf{x}) = \sigma(\mathbf{W}_G \mathbf{x} + b_G)$, где σ - сигмоидальная функция.

Для выполнения экспериментов мы использовали два таких слоя.

Распознавание последовательности организовано на ячейках LSTM. Слой с ячейками LSTM [5] представляет собой замену коэффициентов скрытого слоя ($\mathbf{W}$) нейронной сети на систему уравнений, позволяющей соединить элементы LSTM горизонтально и организовать хранение информации с регулируемым запоминанием (см. рис. 2).

4. Декодер. Алгоритм использования вектора несовпадения вычисленных классов по сравнению с эталонными для обратного распространения ошибки

Переход к формированию языковой модели, вычисляющей вероятность появления следующего слова w_{t+1} (именованной сущности или другого слова) на основании предъявления последовательности символов языка $\mathbf{w} = [w_1, \text{K}\ w_t]$ выполнен следующим образом. При каждом обновлении весов нейронной сети, при предъявлении новых признаков-цепочек символов, происходит вычисление функции ошибки, основанной на совпадении или несовпадении индекса класса (номера слова в словаре) в наборе для обучения и вычисленного индекса класса (номера слова в словаре) для каждой цепочки символов, представляющей слово:

$$y^* = \arg\max_{y \in Y(z)} p(y | z; \mathbf{W}, b).$$

В качестве одного из результатов, в случае успешного обучения, будет сопоставление участков символьных цепочек, представляющих слово, в качестве самостоятельных элементов [19].

Вычисление класса слова (или класса NE) в предложении (скрытое от входа нейросети представление текстовых данных) при предъявлении некоторой цепочки символов, содержащей, это слово или, в случае ошибочного предсказания, набор символов к предполагаемому слову не относящийся, организовано путём добавления двух слоёв к выходу рекуррентной нейронной сети - слой отбрасывания коэффициентов (Dropout Layer) с вероятностью отбрасывания 0,5 и т.н. линейный слой с размерностью, равной длине словаря:

$$P(\mathbf{x}) = \mathbf{W}_P \mathbf{x} + b_P.$$

Другими словами, происходит умножение выхода нейронной сети - матрицы размерности $S \times N$ на матрицу размерности $N \times T \times P$, где S - количество предложений (100), N - размерность выхода нейронной сети, T - количество слов в предложении (35), P - размер словаря.

Результирующая матрица представляет значения ненормированных принадлежностей слов из словаря к распознаваемым классам в массиве предложений, который нейросеть (не получая на вход “правильных” номеров термов непосредственно) получает в виде последовательности символов, и в процессе оптимизации будет учиться распознавать цепочки этих символов как неделимые фрагменты - слова, предсказывая каждое такое слово, и в том числе, предсказывая правильно или с ошибкой класс 0 - именованную сущность.

Сократить размерность P можно вычислив индекс максимальной логистической вероятности (softmax) присвоив соответствующему элементу массива $S \times T$ этот индекс - предполагаемый номер слова (класса) в словаре с целью сравнения текущего выхода нейронной сети с образцовым.

Оптимизация коэффициентов слоёв нейронной сети выполняется применением метода градиентного спуска (SGD), аргументом для которого является значение ошибки: вычисленной функции кросс-энтропии для вероятности принадлежности к каждому слову языка:

$$H(p,q) = -\sum_y p(y)\log q(y),$$

которую предстоит преобразовать обратно к значению (с некоторой ошибкой) коэффициентов рекуррентной нейронной сети с памятью LSTM.

5. Методика проведения экспериментов

Для выполнения экспериментов были использованы два корпуса текстовых данных: Penn Treebank [24] и English NER task CoNLL2003 [25], их статистические данные приведены в таблице 1. Для корпуса CoNLL2003 в качестве NE использовались NE-PER (Personal, персона, человек). Для оценивания качества распознавания именованных сущностей мы использовали типовые показатели: общая точность всех классов, точность, полнота, F1 для первого класса, представленного NE [26], см.также [27].

Таблица 1. Статистические характеристики экспериментальных наборов.

Набор	Вид текстовой единицы	Penn Treebank	CoNLL2003
Обучающий	Предложения	42068	14987
	Слова	887521 (45020-NE)	204567 (3432-NE)
Валидация	Предложения	3370	3466
	Слова	70390 (3485-NE)	51578
Тест	Предложения	3761	3684
	Слова	78669 (4794-NE)	46666

6. Результаты экспериментов

6.1. Эксперимент 1 - Стандартная задача распознавания NE

Результаты распознавания тестовой совокупности с использованием многоклассовой функции потерь представлены на рисунке 3.

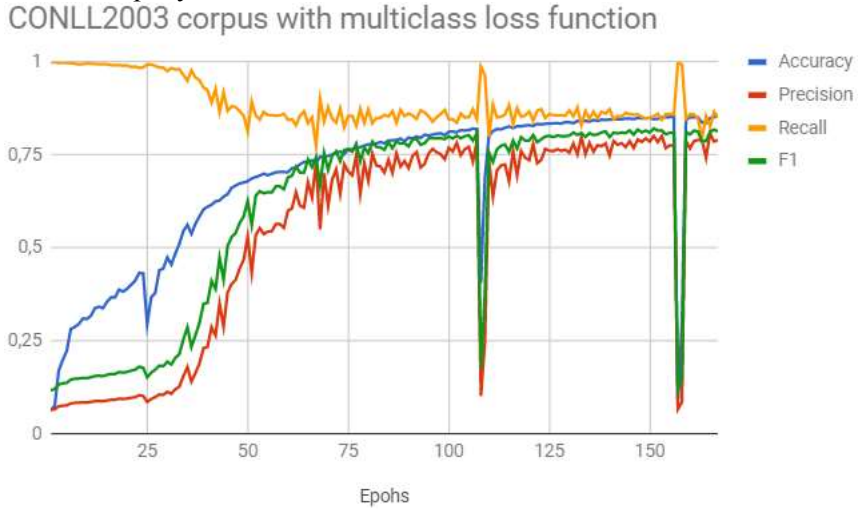

Рисунок 3. Результат распознавания тестовой совокупности CoNLL2003.

6.2. Эксперимент 2 - Распознавание NE, сформированные случайным образом

Для корпуса CoNLL2003составим признаковое пространство таким образом, чтобы символьные цепочки, представляющие именованные сущности были случайной длины 3..20 и

представлены случайными символами и на обучении, и на тесте. Результаты представлены на рисунке 4.

Рисунок 4. Результат распознавания тестовой совокупности корпуса Conll2003 со случайно искаженными символьными признаками NE при обучении и на тесте.

Далее мы проверим, не является ли этот экспериментальный результат ошибкой.

6.3. Эксперимент 3 - Уточнённая постановка задачи распознавания уникальных NE

Используя описание проблемы распознавания Chem-NER [3], поставим задачу распознавания именованных сущностей на корпусе CoNLL2003 следующим образом: обучение и распознавание именованных сущностей, уникальных для обучающей и с обычным написанием для тестовой совокупности. Мы усложним задачу, и заставим обучаться на символьных признаках NE, которые не встретятся у NE в тестовой совокупности, представив каждую именованную сущность корпуса CoNLL2003 цепочкой случайной длины 3..20 символов, каждый символ которой будет сгенерирован также случайно. В сущности, это постановка задачи transfer learning [27] применительно к задаче распознавания именованных сущностей.

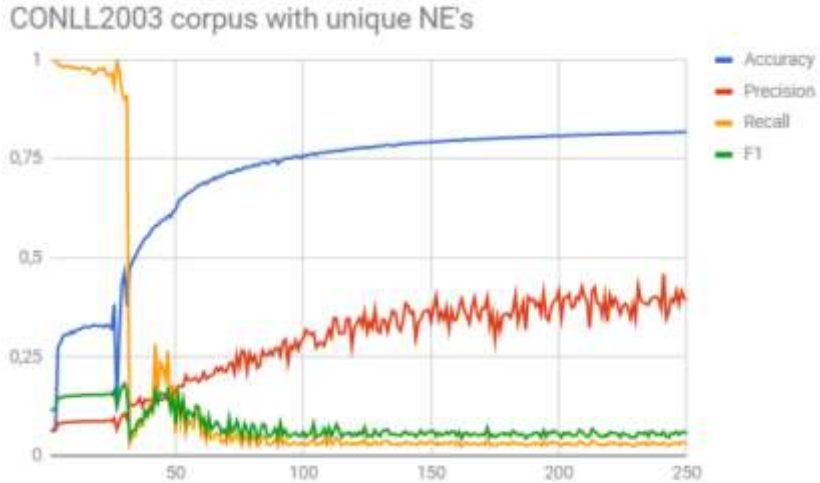

Рисунок 5. Результат распознавания тестовой совокупности корпуса CoNLL2003 нейросетью, обученной со случайно искаженными символьными признаками NE.

Очевидно, что имеет место противоречие результатов данного эксперимента и предыдущего.

6.4. Эксперимент 4 - Адаптация алгоритма для распознавания уникальных NE

Использовав условия формирования признакового пространства из эксперимента 3 заменим функцию предсказания класса softmax таким образом, чтобы степень доверия в пользу любого

класса была больше 50%, либо предсказывался класс 0 - именованная сущность, уникальное написание, то, чего нейросеть распознать не может с хорошей вероятностью:

$$y^* = \mathrm{ROUND}\left(\arg\max_{y \in Y(z)} p(y|z;\mathbf{W},b) \right).$$

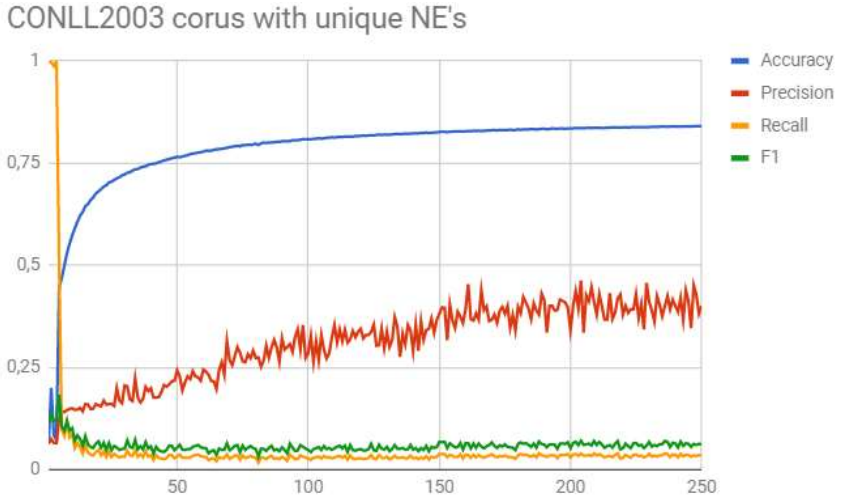

Рисунок 6. Результат распознавания валидационной совокупности корпуса CoNLL2003 нейросетью, обученной со случайно искажёнными символьными признаками NE.

В этом случае функция ошибки не будет при обучении учитывать ошибки распознавания случайно искажённых символов NE.

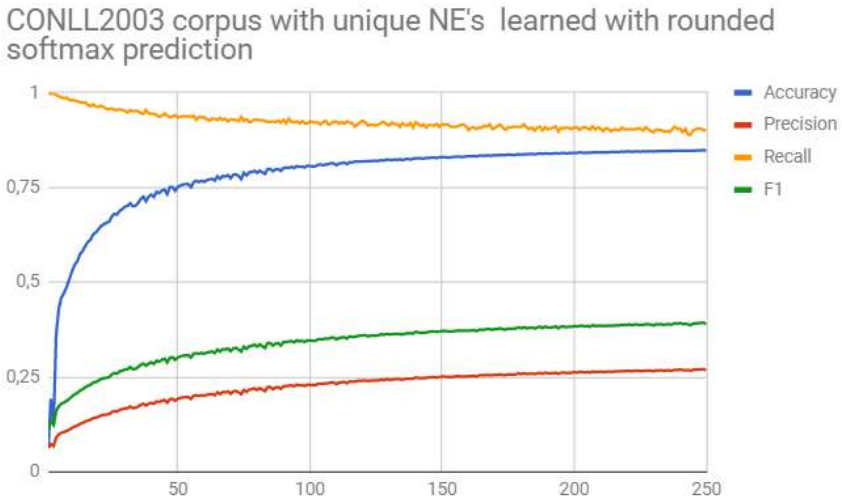

Рисунок 7. Результат распознавания тестовой совокупности корпуса CoNLL2003 нейросетью с модифицированными функциями предсказания и потерь, обученной со случайно искажёнными символьными признаками NE.

6.5. Эксперимент 5 - Проверка решения на корпусе Penn TreeBank

Выполним эксперимент 4 на корпусе Penn TreeBank - исходя из предположения, что, исказив написание каждой именованной сущности, мы исключим известную проблему узнавания нейросетью символов <UNK> (от слова unknown), которыми закодирована каждая именованная сущность в этом корпусе. Однотипность состава этого корпуса текстов (биржевые сводки и финансовые новости) и его объем позволят узнать, насколько точно позволяет узнавать абсолютно уникальные именованные сущности в ситуации большого обучающего набора методом, предложенным для эксперимента 4.

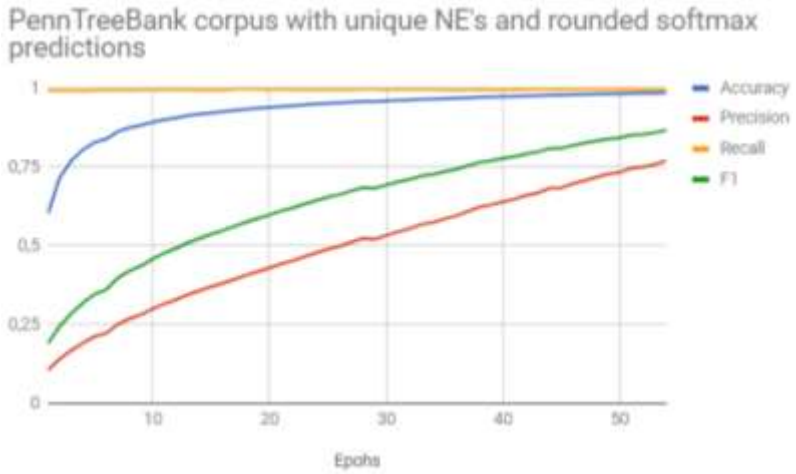

Рисунок 8. Результат распознавания тестовой совокупности корпуса PennTreeBank нейросетью с модифицированными функциями предсказания и потерь, обученной со случайно искажёнными символьными признаками NE.

6.6. Эксперимент 6 - Доработка метода и получение сравнительных показателей

В наших экспериментах мы выделили и подтвердили проблему, также заявленную в статье [29]. К сожалению, наша группа узнала об этих результатах в то время, когда основная экспериментальная работа (эксперименты 1-5) была уже завершена, что является независимым подтверждением наличия этой проблемы для промышленного применения. Однако, первоначально, для выделения этой проблемы мы использовали более радикальную постановку задачи и предложили своё решение, которое безразлично к написанию NE, пусть и не с выдающимся качеством распознавания. Теперь, для получения сравнительных характеристик функцию потерь оставим как в экспериментах 5, 6, а на вход сверточного кодировщика предъявим символьные цепочки NE. Для оценки качества распознавания полностью исключим предложения с известными на момент обучения NE на тесте, как это сделано в статье [29]. Поскольку в работе [29] были использованы газетиры (географические словари), мы тоже будем их использовать. Сравнительная характеристика этого метода с газетирами и без них представлена в таблице 2 для 1500 эпох обучения такой модели. Для этого эксперимента, как и для статьи [29] целевым классом NE для распознавания будут Person, Organisation, Location.

Таблица 2. Характеристики качества с использованием газетиров и без них.

	no gazetters			with gazetters		
	precision	recall	F1	precision	recall	F1
Conll test A	0.5637	0.78099	0.649261	0.59	0.7536	0.6573
Conll test B	0.436264	0.870625	0.5715	0.4485	0.8525	0.5788

Очевидно, что качество распознавания выше, если некоторый обобщенный портрет NE всё-таки будет составлен в процессе обучения. Сравнительные результаты к источнику [29] представлены в таблице 3. Полученные нами показатели сравниваются с приведенным в таблице 14 (Table 14: Out of domain performance: F1 of NERC with different models).

Таблица 3. Сравнительные результаты к источнику [29].

Our method			Memorization			CRF Suite			SENNA		
Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0.59055	0.75364	0.65737	53.14	22.36	31.48	67.12	38.57	48.99	68.62	58.68	63.26
0.44853	0.85251	0.57881	55.85	22.49	32.07	67.94	36.41	47.41	64.61	51.94	57.58

Численные результаты экспериментов представлены в таблице 4, качество полученных моделей естественного языка зафиксировано на указанную в таблице эпоху обучения.

Таблица 4. Результаты экспериментов.

Эксперимент	Рисунок статьи	Эпоха обучения	Общая точность	Точность NER	Полнота NER	F1 measure
1	3	150	0.849	0.7859	0.8495	0.8102
2	4	44	0.9214	0.8825	0.9950	0.9344
3	5	250	0.8174	0.3921	0.0302	0.05432
3	6	250	0.8401	0.4003	0.0346	0.0616
4	7	250	0.8466	0.2681	0.9023	0.39002
5	8	54	0.9852	0.7708	0.9943	0.8668
6	--	1500	--	0.5637	0.78099	0.649261

7. Обсуждение результатов

Интерпретация результатов эксперимента 2 (хорошее распознавание уникальных именованных сущностей) как успешного - ошибочна, это противоречит результатам эксперимента 3. Обоснованием такого противоречия может быть следующая программная особенность обработки выхода нейронной сети функцией из программного пакета tensorflow softmax:

- вычисления вероятности класса P из значений выхода нейронной сети, дающим признак класса 0 - типового индекса класса для класса NER-Person, пусть с низкой вероятностью, но большей чем у остальных n классов, представляющих слова;

-либо дающую индекс класса 0 (Person) для равновероятной оценки принадлежности терма ко всему множеству классов.

Тем не менее, встретив в символьной цепочке написание именованной сущности не случайного характера (см. эксперимент 3), такой классификатор присвоит ей индекс класса (слова) отличный от именованной сущности, более похожий на другое не случайное слово. Тривиальным примером подобной ситуации может быть задача: распознать имя собственное - фирменное название молочного напитка “Снежок” при применении модели, которая училась на корпусе, содержащем описания других напитков, а в качестве контрпримера для обучения служили тексты детских сказок, в которых слово Снежок упоминалось в качестве погодного явления, и никогда не упоминалось в качестве имени собственного.

Указанная проблема свидетельствует о том, что существующие методики обучения распознаванию именованных сущностей имеют в своей постановке существенный недостаток - качество распознавания будет зависеть от пересечения списка именованных сущностей между наборами для обучения и распознавания. Противоречием в такой постановке является потенциально уникальный характер написания именованной сущности и статистический метод её распознавания.

Эти результаты означают, что можно ставить задачу распознавания именованных сущностей как поиск символьной цепочки, ранее не присутствовавшей в обучении.

Очевидным решением этого противоречия служит повышение порога чувствительности нашего классификатора, например, до вероятности 50% - точного определения ранее известных, типовых, слов в предложении. Эксперименты 4 и 5 показывают, что эта цель достижима, а в случае большого обучающего набора (эксперимент 5) позволяет получить качество распознавания, приближающееся к качеству распознавания не уникальных именованных сущностей.

8. Заключение и развитие этой работы

Проведенные эксперименты свидетельствуют о возможности применения многослойных нейронных сетей для распознавания именованных сущностей, в том числе и тех, что сильно отличаются от обучающего набора: для относительно сложных текстов набора CoNLL2003 распознавание уникальных NE возможно с точностью 0.5637; полнотой 0.7809; F1-score 0.6492.

Тем не менее, исследователям стоит разделять постановку задачи опознавания известных и похожих по написанию на них именованных сущностей и постановку, в которой требуется найти ранее не известные именованные сущности, совсем не похожие на те, которые были в обучении. Об этой проблеме свидетельствует и работа [29], в которой получены показатели, сравнимые с представленными в данном исследовании. Наши эксперименты показали, что делая обычную постановку, и, тем более, улучшая её статистическими подсказками (газетиры, дополнительные признаки) исследователь всего лишь существенно улучшает задачу распознавания именованных сущностей известных по словарю, включая, разумеется, более сложные современные алгоритмы, улучшающие точность. В эксперименте 6 статистические подсказки улучшили F1-score на 0.7%...0.8% за счёт снижения полноты распознавания. Достижимые характеристики очередного метода при обычной постановке задачи будут зависеть от степени пересечения именованных сущностей в обучающем и тестовом наборе. Более естественная постановка этой задачи - вторая, распознавание типовых участков текста, расположенных между именованными сущностями. В нашем исследовании также выявлена проблема, связанная с применением функции softmax (в частности tensorflow tf.nn.softmax) применительно к коэффициентам выходного слоя нейросети, представляющих именованные сущности. В нашей работе мы обратили внимание на вероятностную природу в существенной степени уникальных последовательностей символов в противопоставлении её типовым (высоко вероятным) и хорошо распознающимся признакам последовательностей других связующих слов.

9. Литература

[1] Kao, A. Natural Language Processing and Text Mining / A. Kao, S. Poteet – London: Springer-Verlag, 2007.
[2] Patrick, J. High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge / J. Patrick, M. Li // Journal of the American Medical Informatics Association. – 2010. – Vol. 17. – P. 524 -527.
[3] Krallinger, M. The CHEMDNER corpus of chemicals and drugs and its annotation principles // Journal of cheminformatics. – 2015. – Vol. 7(1). – P. 1-17. DOI: 10.1186/1758-2946-7-S1-S.
[4] Hochreiter, S. Long Short-Term Memory / S. Hochreiter, J. Schmidhuber // Neural Comput. – 1997. – Vol. 9(8). – P. 1735-1780.
[5] Anh, L. Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition / L. Anh, M. Arkhipov, M. Burtsev // Proc. AINL, 2017.
[6] Robbins, H. A Stochastic Approximation Method / H. Robbins, S. Monro // The Annals of Mathematical Statistics. – 1951. – Vol. 22(3). – P. 400-407.
[7] Wald, A. Statistical Decision Functions – Wiley, 1950.
[8] Jing, J. Information extraction from text. Mining Text Data – Springer, 2012. – 524 p.
[9] Isozaki, H. Efficient support vector classifiers for named entity recognition / H. Isozaki, H. Kazawa // Proceedings of the 19th international conference on Computational linguistics. – 2002. – Vol. 1. – P. 1-7.
[10] Zhou, G.D. Named entity recognition using an hmm-based chunk tagger / G.D. Zhou, J. Su. // Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 2002. – P. 473-480.
[11] Klinger, R. Automatically selected skipedges in conditional random fields for named entity recognition // Proceedings of the 8th International Conference on Recent Advances in Natural Language Processing, 2011. – P. 580-585.
[12] Chen, W. Chinese named entity recognition with conditional random fields / W. Chen, Y. Zhang, H. Isahara // Proceedings of the 5th Special Interest Group of Chinese Language Processing Workshop, 2006. – P. 118-121.
[13] Bengio, Y. Learning long-term dependencies with gradient descent is difficult / Y. Bengio, P. Simard, P. Frasconi // IEEE Transactions on Neural Networks. – 1994. – Vol. 5. – P. 157-166.
[14] Ивахненко, А.Г. Метод группового учёта аргументов в задачах прогнозирования. – Автоматика. – 1976. – Vol. 6. – С. 24-33.

[15] LeCun, Y. Handwritten Digit Recognition with a Backpropagation Network / Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel // Proceedings of NIPS, 1989.
[16] Bengio, Y. Learning Deep Architectures for AI // Foundations and Trends in Machine Learning. – 2009. – Vol. 2(1). – P. 1-127. DOI:10.1561/2200000006.
[17] Ma, X. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF / X. Ma, E. Hovy // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. – 2016. – Vol. 1. – P. 1064-1074.
[18] Kim, Y. Character-Aware Neural Language Models / Y. Kim, Y. Jernite, D. Sontag, A.M. Rush // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016. – P. 2741-2749.
[19] Jason, P.C. Named entity recognition with bidirectional lstm-cnns / P.C. Jason, E. Nichols // Transactions of the Association for Computational Linguistics. – 2016. – Vol. 4. – P. 357-370.
[20] Srivastava, R.K. Highway networks / R.K. Srivastava, K. Greff, J. Schmidhuber // arXiv preprint arXiv:1505.00387, 2015.
[21] Pundak, G. Sainath: Highway-LSTM and Recurrent Highway Networks for Speech Recognition / G. Pundak, N. Tara // Proc. Interspeech, 2017.
[22] LeCun, Y. Gradient based learning applied to document recognition / Y. LeCun, L. Bottou, Y. Bengio, P. Haffner // Proceedings of the IEEE, 1998. – P. 2278-2324.
[23] Ioffe, S. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift / S. Ioffe, C. Szegedy // Procedings 32nd ICML, 2015. – P. 448-456.
[24] Marcus, M. Building a large annotated corpus of English: the Penn Treebank / M. Marcus, B. Santorini, M.A. Marcinkiewicz // Computational Linguistics. – 1993. – Vol. 19(2). – P. 313-330.
[25] Tjong, E.F. Introduction to the conll-2003 shared task: Language independent named entity recognition / E.F. Tjong, K. Sang, M. Fien De // Proceedings of CoNLL. – 2003. – Vol. 4. – P. 142-147.
[26] Van Rijsbergen, C.J. Information Retrieval – Butterworth-Heinemann, 1979.
[27] He, H. Learning from imbalanced data / IEEE Transactions on Knowledge and Data Engineering, 2009. – P. 1263-1284.
[28] Pratt, L.Y. Discriminability-based transfer between neural networks // NIPS Conference: Advances in Neural Information Processing Systems 5. Morgan Kaufmann Publishers, 1993. – P. 204-211.
[29] Augenstein, L. Generalisation in Named Entity Recognition: A Quantitative Analysis / L. Augenstein, L. Derczynski, K. Bontcheva // Computer Speech & Language, 2017. DOI:10.1016/j.csl.2017.01.012. 2017.

On the multiclass classification of words by a recurrent neural network with memory (LSTM) as applied to the problem of recognition of named entities

V.S. Vakurin[1], A.V. Kopylov[1], O.S. Seredin[1], K.S. Mertsalov[2]

[1]Tula State University, Lenina str. 92, Tula, Russia, 300012
[2] Rensselaer Polytechnic Institute, Troy, NY, USA

Abstract. The paper addresses the issues of training back propagation neural networks for recognition of named entities using multilayer architectures and various feature spaces formed on symbolic chains. The article presents the results of experiments showing the dependence of predictive properties on the intersection of a set of named entities between the training and test set in the standard formulation of the named entity search problem. We also propose a way to improve the predictive properties of models for detecting named entities that were not previously presented during training.