«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Центр непрерывного образования
Факультета компьютерных наук

АТТЕСТАЦИОННАЯ РАБОТА

Использование Seq2Seq архитектур в задаче модернизации языка (генерации современной формы слова по его древней форме)

Выполнил:
Арк Михаил Юрьевич
Научный руководитель:
Дереза Оксана Владимировна
Insight Centre for Data Analytics,
NUI Galway, Ireland

Москва 2019

Введение

Медленно, но верно растет интерес к обработке исторических документов современными методами. На это указывают профессионалы с обеих сторон: специалисты по машинному обучению [Bollmann, Søgaard 2016] и лингвисты [Черванёва 2012]. Подтверждением этому может служить увеличивающееся количество оцифрованных и вновь составленных ресурсов с историческими документами и их анализом. В качестве примера можно привести сайт, на котором собраны исторические корпусы разных языков [CLARIN 2019]. Приведение слов к единому виду часто является шагом, предшествующим анализу и дальнейшему изучению исторических документов. Качество обработки текста на этом этапе может оказать сильно влияние на результаты всего цикла анализа исторических документов.

Возможность использования глубоких нейронных сетей для модернизации языка под вопросом: эффективное обучение подразумевает наличие большого количества данных, и как раз в этой части домен не изобилует богатством. Рекуррентные сети символьного уровня позволяет создать компактную архитектуру, которой может и не потребоваться большого количества примеров для качественной работы.

Словари древних языков обычно очень небольшие, поэтому перед исследователями стоит задача не только в применении нейронных сетей непосредственно к данным, но и расширение самого словаря сопоставлений. Кроме того, существующие словари напрямую непригодны для применения в данной задаче, так как словарная статья не является модернизированной версией древнего слова, а часто состоит из нескольких слов. Недостаточное количество данных может стать серьезной проблемой при решении задачи модернизации языка.

Целью проекта является оценка возможности использования Seq2Seq архитектур для решения задачи модернизации языка.

В связи с этим, можно выделить три основных задачи работы.

1. Найти общий метод приведения серии слов в одно, который покажет достаточную эффективность в данной задаче.
2. По возможности максимально обогатить словарь сопоставлений для повышения эффективности обучения.
3. Протестировать различные архитектуры и выбрать ту, что дает максимальное качество в решении данной задачи.

Для этого была изучена литература по теме, проведена программная обработка словарей с помощью языка Python и поставлены эксперименты с архитектурой нейронной сети с помощью библиотеки Keras.

Работа проводилась в октябре 2019 года.

Обзор литературы

Проанализированные материалы логично разделить на две части: те, что относятся к используемым в работе технологиям, и те, что касаются домена.

Предположение, что для представления словарной статьи в виде одного слова может пригодиться представление слова в векторном пространстве привело к статье о Word2Vec [Mikolov, Chen, Corrado, Dean 2013]. Помимо самой технологии векторизации в ней приведены результаты экспериментов, где векторы сопоставлялись не только словам, но и фразам. Вектора в пространстве расположены таким образом, что близкие семантически слова имеют маленькое косинусное расстояние между соответствующими им векторами.

Эффективность символьных рекуррентных сетей для генерации текста способна поразить даже опытных исследователей аналитических систем [Karpathy 2015].

LSTM (Long Short-Term Memory) [Hochreiter, Schmidhuber 1997] — популярная архитектура рекуррентной сети. Основная идея этой архитектуры — выделение ячейки памяти, ответственной за хранение информации, полученной в предыдущие моменты времени. Некоторые исследователи [Ponomareva, Milintsevich, Artemova 2019] указывают на эффективность применения двунаправленных LSTM в задачах, связанных с символьным уровнем языка.

GRU (Gated Recurrent Unit) [Chung, Gulcehre, Cho, Bengio 2014] — еще одна рекуррентная архитектура, позволяющая выполнять задачи не менее эффективно чем с использованием LSTM, уменьшая при этом сложность модели и количество обучаемых параметров.

Задача модернизации часто сравнивается с задачей перевода, поскольку схожи применяемые методы и структура данных [Piotrowski, 2012]. Одной из наиболее популярных архитектур в машинном переводе являются модели

Seq2Seq [Sutskever, Vinyals, Le 2014]. Такие модели состоят из двух рекуррентных сетей: кодировщика и декодировщика. Кодировщик строит представление входной последовательности в виде тензора своего внутреннего состояния. Тензор копируется в декодировщик. По полученному представлению декодировщик пытается восстановить целевую последовательность.

Для преобразования символов во входные вектора используется матрица представлений. Каждая строка соответствует векторному представлению соответствующего символа. Каждый символ перед подачей на вход LSTM сети заменяется на соответствующую строку матрицы представлений. Достаточно эффективным подходом к созданию матрицы представлений является One-Hot Encoding, возможность его использования является отличительно чертой символьных архитектур.

Проблемой нейросетевых моделей является необходимость сжать всю информацию в векторе представления [Bahdanau, Cho, Bengio 2014].
Seq2Seq с вниманием [Luong, Pham, D Manning 2015] значительно улучшает качество работы системы. Механизмы внимания — это подход в машинном обучении, заключающийся в выделении части входных данных для более детальной обработки. В качестве объекта внимания в таких моделях используются выходы последнего слоя кодирующей части для каждого элемента последовательности. В качестве ключа выбирается выход последнего слоя декодирующей части. Для генерации последовательности вектор контекста конкатенируется с ключом и пропускается через еще один рекуррентный слой.

Говоря о литературе ближе к предметной области, следует упомянуть исследование [Bollmann, Søgaard 2016] о нормализации исторических написаний. В статье проведено исчерпывающее описание преимуществ использования LSTM перед некоторыми другими методами, приведен подробный анализ результатов, благодаря которому можно дать оценку

эффективности выбранного подхода для задачи модернизации. Также описаны основные проблемы, стоящие перед специалистами, работающими с историческими текстами, в том числе упомянуто влияние когнитивного искажения перевода отдельными авторами на данные, и вместе с этим на результат исследования.

Одной из сложностей работы с историческими текстами является большая вариабельность написания слов от текста к тексту [Hendrickx, Marquilhas 2011]. Приведение к единой форме рассматривается специалистами как отдельная проблема.

Задачу модернизации языка частично можно воспринимать как задачу перевода с древнего языка на современный. Символьный уровень модели, выбранный в качестве главного подхода, не типичен для перевода с одного языка на другой [Costa-Jussa, Fonollosa 2016], но по некоторым исследованиям решается не менее успешно [Lee, Cho, Hofmann 2017]. В статьях рассмотрены примеры эффективных моделей символьного уровня, примеры архитектур и указание на положительное влияние выравнивания размеров входящего тензора [Zhao, Zhang 2016].

Отношения древнего языка и современного можно воспринимать как близкородственные, если такое определение вообще уместно. Существует утверждение, подкрепленное экспериментами, что перевод машинными методами осуществляется проще в паре близкородственных языков, чем для языков, не обладающих такой связью [Altintas, Cicekli 2002].

Оценка эффективности NLP моделей признается сообществом непростой задачей [Toldova, Lyashevskaya, Bonch-Osmolskaya, Ionov 2015], особенно для морфологически сложных языков, поэтому для оценки качества была выбрана простая в подсчете и понимании метрика средней доли точности по всем символам выходной последовательности.

Методы исследования

По причине того, что однословных переводов слов с древнерусского на русский в исходном виде пригодных для решения задачи в открытых источника найти не удалось, необходимо было собрать и подготовить данные. Данные свободных источников – это древние слова с сопоставленными им словарными статьями. В работе предполагается, что с помощью Word2Vec можно получить для каждого древнего слова какое-то однословное соответствие. Часть этих соответствий будет истинной модернизированной версией древнего.

На следующем этапе необходимо было провести эксперименты с архитектурой нейронных сетей и настройкой гиперпараметров для наилучшего результата. Предполагается, что Seq2Seq архитектура будет достаточно эффективной в решении задачи модернизации языка; в частности, ожидается, что двунаправленная LSTM с вниманием покажет наилучший результат.

Третьим этапом была проверка генерализующей способности всего пайплайна на примере английского и древнеанглийского.

Эксперименты

Начнем с обзора существующих словарей древнерусского языка. Для консистентности данных выбор ресурсов был ограничен письменным вокабуляром 11-15 вв. Из ресурсов в открытом доступе были выбраны четыре (по два на язык), которые показались наиболее подходящими для автоматической обработки:

1. Словарь древнерусского языка (XI-XIV вв.) [АН СССР ИРЯ 1988].
2. Онлайн версия перевода словаря древнерусского языка И. И. Срезневского [Балыберин 2013].
3. Словарь древнеанглийского языка 1150 – 1580 гг. [Mayhew, Skeat 2004]
4. Онлайн словарь древнеанглийского [Lewis, Arbor 2018].

Текст словарных статей был нормализован, из них же взята информация о принадлежности частям речи.

Словарная статья обращается в одно слово по следующему алгоритму:

1. Если определение состоит из одного слова - ставим в соответствие.
2. Слова, входящие в определение, транформируются и векторизуются с помощью предобученной Word2Vec модели [Kutuzov, Kuzmenko 2017].
3. Векторы слов взвешенно усредняются. Вес определен как повышенный коэффициент для существительных по сравнению со словами других частей речи, а если существительных в определении нет, то коэффициент повышается для вторых по порядку слов в определении. Эффективность подбора данной эвристики подтверждается минимумом по метрике: расстояние Левенштейна, нормированное на длину словарного слова.
4. В векторном пространстве ищутся кандидаты на однозначное соответствие. Отбираются пять ближайших векторов и выбирается слово, совпадающее со словарным по части речи, либо наиболее близкое по описанной метрике.

Из всего списка полученных слов выделены особо слова, имеющие значение метрики меньше 0.65. Они составили примерно треть от всего списка (четверть для английского). Из них составляется отдельный короткий датасет для схожих слов.

Информация о получившихся датасетах представлена в таблице:

Название	ORUS	OENG
Общее количество пар	38435	47521
Количество пар, метрика < 0.65	13057 (~34%)	12596 (~26.5%)
Максимальная длина слова (in)	15	15
Максимальная длина слова (out)	15	15
Количество уникальных символов (in)	36	49
Количество уникальных символов (out)	33	26

Словари превращаются в три трехмерных тензора, размером (количество строк) × (максимальная длина слов) × (количество символов словаря) с некоторым различием в размерностях для входного тензора, таргета и тензора для форсирования обучения.

Эксперименты с архитектурой сети начались с пробы GRU-32 на полном русском датасете. Затем на коротком словаре последовательно GRU-32, LSTM-64, BiLSTM-64, LSTM-64 + ATT, BiLSTM-128 + ATT. Таким образом Baseline модель была опробована на полном русском, а основной поиск наиболее подходящей архитектуры проходил на коротком словаре.

Результаты представлены в таблице (Accuracy посимвольно):

Архитектура	ORUS_ALL	ORUS_CLOSE	OENG_ALL	OENG_CLOSE
GRU-32	0.7242	0.7892	-	-
LSTM-64	-	0.8259	-	-
BiLSTM-64	-	0.8501	-	-
LSTM-64 + Att	-	0.8566	-	-
BiLSTM-128 + Att	0.8032	**0.8648**	0.7589	0.8248

Лучший результат показала последняя из списка проверенных, представленная на рис. 1. Эта архитектура была использована при обучении моделей на остальных датасетах.

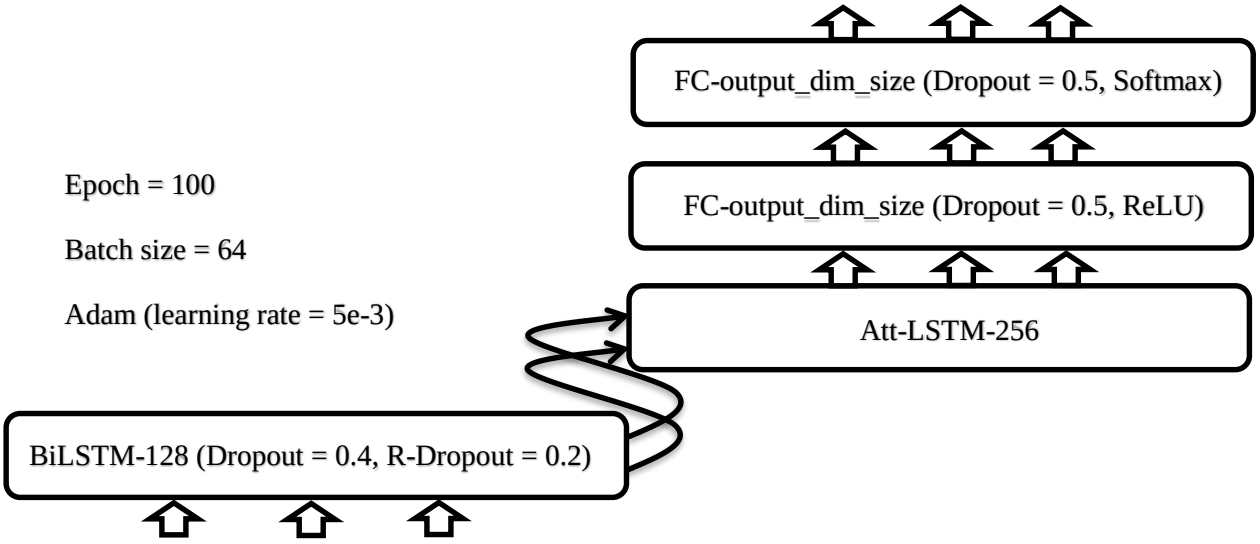

Рис. 1 Архитектура модели BiLSTM-128 + Attention

Полученные 86% точности на символах могут показаться невпечатляющим результатом, особенно в свете того, что в рассмотренных статьях исследователи достигают 80% точности на словах, но важно понимать, что́ берется в качестве данных. Часто в задаче нормализации используют "золотой стандарт" нормализованного языка как таргета. В нашем случае одной из задач было нахождение генерализованного метода нахождения переводов без ручной обработки и таргетом служили слова, ближайшие в пространстве взвешенно-усредненному вектору слов определения. Это значит, что несмотря на минимизированную метрику при поиске перевода данные остаются сильно зашумленными. Из-за малого размера словаря на первой итерации экспериментов для каждого древнего слова ставилось свое однословное соответствие. В реальных переводах это не всегда возможно.

Далее была предпринята попытка отсеять шум по порогу метрики. Это помогло оставить только похожие по написанию слова. Этот подход сократил

шум в данных, но и сместил задачу к более простой, что и отразилось на результате.

Accuracy пословно для BiLSTM-128 + Att:

ORUS_ALL	ORUS_CLOSE	OENG_ALL	OENG_CLOSE
0.1569	0.2903	0.1656	0.2891

Из-за смещенности данных нельзя напрямую сравнивать метрики на разных рассмотренных датасетах. В тоже время, факт повышения результата на сокращенной выборке позволяет сделать вывод о том, что увеличением степени обработки входящих данных можно добиться лучших результатов.

Усложнения архитектуры, как и ожидалось, позволяют повысить точность на каждом шаге, что даёт повод для продолжения экспериментов с устройством сети.

Проверка обобщающей силы всего пайплайна для решения задачи модернизации была проведена для пары древнеангийского - английского, которая не лишена своих особенностей, и особенности эти не были учтены ни при поиске переводов, ни при построении архитектуры. При должной адаптации можно добиться гораздо более впечатляющих результатов. Нашей же целью было проверить, что схема работает и без пристального внимания исследователя ко входящим данным, что и подтвердилось на практике.

Проведенный анализ позволяет сделать вывод: представленная схема способна решить задачу модернизации языка с достаточной точностью.

Заключение

Во процессе работы над проектом была изучена литература, проведены эксперименты и получены следующие результаты:

1. Выбран подход, позволяющий собрать данные и трансформировать их в вид, пригодный для использования в нейронных сетях архитектуры Seq2Seq для решения задачи модернизации языка.

2. Найдена подходящая для решения задачи архитектура, посредством применения и тестирования предложенных в рассмотренной литературе примеров алгоритмов, с некоторыми адаптациями к поставленной задаче.

3. Проверена возможность генерализации описанных методов на другие пары языков.

4. Собраны данные, написан код для их обработки и для дальнейших исследований в этой области. Информацию можно найти в открытом доступе в репозитории проекта[1].

В результате проведенной работы можно сделать вывод: задачу модернизации языка можно решить с достаточной точностью с помощью описанного подхода.

Работу по проекту машинного обучения логично вести в рамках итеративной модели жизненного цикла программного продукта. Для дальнейшего улучшения качества решения в первую очередь необходимо вернуться к обработке данных, как к наиболее важной части проекта, особенно в условиях дефицита и зашумленности. На второй итерации работы с данными можно детальнее подойти к выбору алгоритма сопоставления, попробовать другие варианты, либо просто вручную, насколько это возможно почистить их. Можно попробовать увеличить словарь, разнообразив его разными окончаниями, примеры которых приведены в источниках, но не

[1] https://github.com/Mikhail-Ark/seq2seq-word-modernization

использовались на этапе сбора данных. В качестве вариантов расширения словаря можно попробовать использовать аугментации, либо сгенерировать новые данные.

Усложнение архитектуры на каждом шаге приводило к увеличению качества модели, поэтому я считаю, что на следующей итерации необходимо провести дополнительные эксперименты с архитектурой трансформер, как наиболее современной в рассматриваемой области [Al-Rfou, Choe, Guo 2018].

Список литературы

1. Marcel Bollmann, Anders Søgaard; Improving historical spelling normalization with bi-directional LSTMs and multi-task learning. // arXiv:1610.05157.

2. Черванёва В. А. Вводный курс по чтению и переводу старославянских текстов: учебно-методическое пособие / В.А. Черванёва. – Воронеж: Воронежский государственный педагогический университет, 2012.

3. Historical corpora. CLARIN - European Research Infrastructure for Language Resources and Technology. https://www.clarin.eu/resource-families/historical-corpora

4. Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean; Efficient Estimation of Word Representations in Vector Space. // arXiv:1301.3781.

5. Andrej Karpathy, The Unreasonable Effectiveness of Recurrent Neural Networks; 2015. http://karpathy.github.io/2015/05/21/rnn-effectiveness/

6. Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling; 2014. // arXiv: 1412.3555v1.

7. Sepp Hochreiter and J¨urgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

8. Maria Ponomareva, Kirill Milintsevich, Ekaterina Artemova. Char-RNN for Word Stress Detection in East Slavic Languages; 2019. // arXiv:1906.04082v1.

9. Michael Piotrowski. Natural Language Processing for Historical Texts. Synthesis Lectures on Human Language Technologies; 2012, Vol. 5, No. 2.

10. Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Advances in neural information processing systems, pages 3104–3112, 2014.

11. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. 2014. // arXiv:1409.0473.

12. Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation; 2015.
// arXiv:1508.04025, 2015.

13. Iris Hendrickx, Rita Marquilhas. From old texts to modern spellings: an experiment in automatic normalization; 2011. Annotation of Corpora for Research in the Humanities Volume 26 - Number 2.

14. Marta R. Costa-Jussa, Jose A. R. Fonollosa; Character-based Neural Machine Translation, 2016. // arXiv:1603.0081.

15. Jason Lee, Kyunghyun Cho, Thomas Hofmann. Fully Character-Level Neural Machine Translation without Explicit Segmentation; 2017.
// arXiv:1610.03017v3.

16. Shenjian Zhao, Zhihua Zhang; An Efficient Character-Level Neural Machine Translation. 2016. // arXiv:1608.04738v2.

17. Kemal Altintas, Ilyas Cicekli. A Machine Translation System Between a Pair of Closely Related Languages; 2003. Proceedings of the 17th International Symposium on Computer and Information Sciences (ISCIS 2002).

18. S. Toldova, O. Lyashevskaya, A. Bonch-Osmolovskaya and M. Ionov. Evaluation for morphologically rich language: Russian NLP; 2015. Int'l Conf. Artificial Intelligence.

19. Словарь древнерусского языка (XI-XIV вв.)1 / АН СССР. Институт русского языка. — М.: Русский язык. 1988.

20. Сергей Балыберин. Перевод словаря древнерусского языка И. И. Срезневского; 2013. http://oldrusdict.ru/dict.html

21. A. L. Mayhew and Walter W. Skeat. A Concise Dictionary of Middle English From A.D. 1150 To 1580; 2004.

22. Middle English Dictionary. Ed. Robert E. Lewis, et al. Ann Arbor: University of Michigan Press, 1952-2001. Online edition in Middle English Compendium. Ed. Frances McSparran, et al. Ann Arbor: University of Michigan Library, 2000-2018. http://quod.lib.umich.edu/m/middle-english-dictionary/

23. Kutuzov A., Kuzmenko E. (2017) WebVectors: A Toolkit for Building Web Interfaces for Vector Semantic Models. In: Ignatov D. et al. (eds) Analysis of Images, Social Networks and Texts. AIST 2016. Communications in Computer and Information Science, vol 661. Springer, Cham. https://rusvectores.org/ru/

24. Rami Al-Rfou, Dokook Choe, Noah Constant, Mandy Guo, Llion Jones. Character-Level Language Modeling with Deeper Self-Attention; 2018. // arXiv:1808.04444v2.