Cross-Language Document Extractive Summarization with Neural Sequence Model*

Захаров П. С., Сельницкий И. С., Кваша П. А., Дьячков Е. А, Петров Е. Д.
Московский физико-технический институт

В статье исследуется модель, которая осуществляет сокращение исходного текста с сохранением начального смысла, кроме того это происходит на языке, который отличается от языка исходного документа. В статье происходит кртакое изложение текста, которое изменяет язык с помощью машинного перевода. Качество проделанной работы оценивается 2 характеристиками: сокращение исходного текста и осуществленный перевод укороченного текста. Работа данной модели осуществляется при помощи спроектированных нейронных сетей. Статья основана на базовой модели, которая учитывала малое количество набора данных, поэтому в данной статье мы планируем исследовать исходную базовую модель на более широкий спектр набора данных, который позволит обеспечить более корректную и улучшенную версию сущетвующей модели в будущем.

Ключевые слова: *Сокращение текстов, машинный перевод, нейронные сети.*

Введение

В настоящее время получили развитие различные базы данных, хранящие большое количетсво документов того или иного рода (научные или журналистские статьи, статистическая информация и т.д.). Эти документы требуется представлять в удобном для быстрой оценки виде, т.е. создать изложение текста. Большое количество данных привело к развитию машинного сокращения текстов. При решении задач подобного рода можно выделить два подхода: абстрактное и экстрактивное изложение. В первом случае сокращенный текст генерируется "с нуля в то время как второй подход отбирает предложения из исходного текста. В данной работе рассматривается в основном экстрактивное сокращение, т.к. оно проще в реализации и в целом на настоящий момент показывает лучшие результаты .

Необходимость делать сокращения текстов на других языках и развитие технологий машинного перевода подтолкнуло создание моделей, реализующих межъязыковое сокращение текстов (англ. Cross-Language Automatic Text Summarization). Наиболее простым решением проблемы является последовательное применение двух техник. Такие модели называются LateTrans и EarlyTrans - в первом случае сначала идет сокращение на языке оригинала, а затем перевод, во втором - наоборот. Обе концепции показали себя не лучшим образом в связи с несовершенством обеих технологий: неидеальный выход первой модели еще сильнее искажался второй. Wan и др.[5] предложили идею усовершенстованной LateTrans модели: при сокращении на языке оригинала учитывались не только информативность предложения, но и предсказание качества перевода. Помимо этого, Wan и др. [6] реализовал систему, создающую изложения-кандидаты, полученные, разными способами, и отбирающую лучшие из них. Pontes и др.[4] использовали кластеризацию и сжатие исходных предложений для получения более информативных кандидатов.

Помимо совершенствования систем в целом, ведутся дальнейшие исследования в моноязыковом сокращении текстов, [7][3]. Можно также отметить прогресс в, к примеру,

---

Научный руководитель: Стрижов В. В. Задачу поставил: Хританков А. С. Консультант: Романов А. В.

задаче векторизации предложений [8]. Модульная архитектура позволяет использовать эти наработки в содании более эффективных CLATS-моделей.

В данной работе предлагается развить идею Wan[5] в приложении к сокращению с переводом с английского языка на русский, используя более совершенные составляющие: в качестве базовой модели используется SummaRunner2016[3], для перевода - openNMT, описанная в [1]. Ставится задача решить проблемы модели сокращения текста, связанные с переносом на другую выбору документов, а также определить необходимость дополнительной предобработки текстов и границы применимости модели.

Для обучения SummaRunner используется исходная выборка - CNN/DailyMail corpus, а для обучения openNMT - параллельный корпус OPUS. Кроме того, имеются данные на русском языке для оценки ошибки. ...

Постановка задачи

В основе реализуемой модели лежит объединение модели моноязыкового аннотирования и модели предсказания качества машинного перевода. В следующих 2 подразделах по отдельности поставлены задачи для каждой из двух моделей, в третьем описано их объединение.

Извлечение

Для реферирования используется трехслойная двухсторонняя рекуррентная нейронная сеть. Пусть $\mathfrak{D} = (\mathfrak{V}, \mathbf{Y})$ - выборка (предложения в документе и бинарный целевой вектор), где $\mathfrak{V} = \{\mathbf{V}_i\}, \mathbf{V}_i \in \mathbb{R}^{N_i \times n}$ - набор предложений, $i \in \{1, .M\}$. При этом $\mathbf{V}_i = = [\mathbf{v}_{ij}] \in \mathbb{R}^{N_i \times n}$ - предложения, состоящие из векторных представлений слов длиной n. Слои первых двух слоев нейронной сети состоят из нейронов, которые описываются двумя $\mathbf{u}_j$ и $\mathbf{r}_j$ по следующим формулам:

$$\begin{aligned} \mathbf{u_j} &= \sigma\left(\mathbf{W}_{ux}\mathbf{x}_j + \mathbf{W}_{uh}\mathbf{h}_{j-1} + \mathbf{b}_j\right) \\ \mathbf{r}_j &= \sigma\left(\mathbf{W}_{rx}\mathbf{x}_j + \mathbf{W}_{rh}\mathbf{h}_{j-1} + \mathbf{b}_r\right) \\ \mathbf{h}'_j &= tanh\left(\mathbf{W}_{hx}\mathbf{x}_j + \mathbf{W}_{hh}\left(\mathbf{r}_j \odot \mathbf{h}_{j-1}\right) + \mathbf{b}_j\right) \\ \mathbf{h}_j &= (1 - \mathbf{u}_j) \odot \mathbf{h}'_j + \mathbf{u}_j \odot \mathbf{h}_{j-1} \end{aligned} \tag{1}$$

На первом слое строится две цепочки нейронов для каждого предложения в тексте. Для одной цепочки $\mathbf{x}_j = \mathbf{v}_{ij}, j \in \{1..N_i\}$, для другой $\mathbf{x}_j = \mathbf{v}_{iN_i-j}, j \in \{1..N_i\}$ - во второй цепочке слова подаются в обратном порядке. Эту цепочку будем называть обратной, а первую = прямой. Здесь N_i - количетсво слов в i-ом предложении

На втором слое строятся такие же цепочки нейронов, для прямой:

$$\mathbf{x}_j = \frac{1}{N_j}\sum_{k=1}^{N_j}\left[\mathbf{h}_j^f, \mathbf{h}_j^b\right], \tag{2}$$

где $\mathbf{h}_j^f$ - скрытое состояние нейронов прямой цепочки для j-ого предложения, а $\mathbf{h}_j^b$ - обратной. Квадратные скобки означают конкатенацию векторов. Для обратной цепочки:

$$\mathbf{x}_j = \frac{1}{N_{M-j}}\sum_{k=1}^{N_{M-j}}\left[\mathbf{h}_{M-j}^f, \mathbf{h}_{M-j}^b\right], \tag{3}$$

где M - число предложений в документе Представление документа $\mathbf{d}$ формируется следующим образом:

$$\mathbf{d} = tanh\left(W_d\frac{1}{M}\sum_{j=1}^{M}\left[\mathbf{h}_j^f, \mathbf{h}_j^b\right] + \mathbf{b}_j\right), \tag{4}$$

где $\mathbf{h}_j^f$ и $\mathbf{h}_j^b$ - скрытые состояния прямой и обратной цепочек на втором слое.

Для классификации используется логистический слой:

$$P\left(y_j = 1|\mathbf{h}_j, \mathbf{s}_j, \mathbf{d}\right) = \sigma\left(\mathbf{W}_c\mathbf{h}_j + \mathbf{h}_j^T\mathbf{W}_s\mathbf{d} - \mathbf{h}_j^T\mathbf{W}_r tanh\left(\mathbf{s}_j\right) + \mathbf{W}_{ap}\mathbf{p}_j^a + \mathbf{W}_{rp}\mathbf{p}_j^r + \mathbf{b}\right) \quad (5)$$

Здесь $\mathbf{s}_j$ - динамическое представление аннотации на j-ом шаге, а $\mathbf{p}_j^a$ и $\mathbf{p}_j^r$ - абсолютные и относительные положения в документе. Члены, обозначенные $\mathbf{W}$ и $\mathbf{b}$ с индексами, являются праметрами модели. Представление аннотации определяется следующим образом:

$$\mathbf{s}_j = \sum_{i=1}^{j-1} \mathbf{h}_i P\left(y_i = 1|\mathbf{h}_j, \mathbf{s}_j, \mathbf{d}\right) \quad (6)$$

Ставится задача минимизовать логистическую функцию правдоподобия:

$$\begin{aligned} l\left(\mathbf{W}, \mathbf{b}\right) = -\sum_{k=1}^{D}\sum_{j=1}^{M_k}\left(y_j^k log P\left(y_j^k = 1|\mathbf{h}_j^k, \mathbf{s}_j^k, \mathbf{d}_k\right) + \right. \\ \left. + \left(1 - y_j^k\right) log\left(1 - P\left(y_j^k = 1|\mathbf{h}_j^k, \mathbf{s}_j^k, \mathbf{d}_k\right)\right) \rightarrow min \right. \end{aligned} \quad (7)$$

Полученное мягкое предсказание в дальнейшем используется для формирования конечного прогноза.

Предсказание качества машинного перевода

Пусть $\mathfrak{D} = (\mathfrak{V}, \mathbf{Y}) \subset \mathbb{R}^{m\times M} \times \mathbb{R}^M$ - выборка (объекты и целевой вектор), $\mathfrak{V} = [\mathbf{V}_i] \in \mathbb{R}^m$ - объекты (предложения). Подчеркнем, что ввиду решения другой задачи в этом подразделе представление предложений отличается - здесь они сами являются объектами, в то время как в предыдущем объектами были слова предложений.

Для предсказания качества машинного перевода используется ε-SVR метод. Требуется найти гладкую функцию $\mathbf{f}$ такую, что:

$$\underset{\mathbf{w},b,\xi,\xi^*}{min} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\xi_i + C\sum_{i=1}^{n}\xi_i^* \quad (8)$$

при условии, что

$$\begin{aligned} \mathbf{w}^T\mathbf{f}(\mathbf{V}_i) + b - y_i &\leqslant \varepsilon + \xi_i \\ y_i - \mathbf{w}^T\mathbf{f}(\mathbf{V}_i) - b &\leqslant \varepsilon + \xi_i^* \\ \varepsilon, \xi_i, \xi_i^* \geqslant 0, i = 1, .., M & \end{aligned} \quad (9)$$

Метриками качества являются

$$\begin{aligned} MSE &= \frac{1}{M}\sum_{i=1}^{M}\left(\hat{y}_i - y_i\right)^2 \quad \text{– среднеквадратичная ошибка и} \\ \rho &= \frac{\sum_{i=1}^{M}\left(y_i - \overline{y}\right)\left(\hat{y}_i - \overline{\hat{y}}\right)}{M s_y s_{\hat{y}}} \quad \text{– коэффициент Пирсона,} \end{aligned}$$

где $\overline{\hat{y}}, \overline{y}$ - средние предсказанных и данных значений соответственно, а $s_y, s_{\hat{y}}$ - их среднеквадратичные отклонения.

После обучения выданные оценки качества нормализуются максимальным значением в документе: $\tilde{y_i} = \frac{\hat{y_i}}{\underset{i}{max}\hat{y_i}}$

Получение итогового результата

На выходе моделей для каждого предложения $\mathbf{V}_i$ получены результаты $y_{MT,i}, y_{ES,i} \in [0, 1]$. Итоговое предсказание строится по правилу

$$y_{final,i} = y_{MT,i}\,(1 - \lambda) + y_{ES,i}\lambda, \tag{10}$$

где λ - эмпирически подбираемый параметр, отражающий важность информативности предложения по сравнению с предполагаемым качеством перевода. К примеру, при $\lambda = 1$ качество перевода вообще не учитывается. После этого отбирается несколько предложений с наивысшими оценками. Их количество зависит от настроек используемой метрики ROUGE. После этого происходит перевод, описание см. в [2][1].

Литература

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[2] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. Opennmt: Open-source toolkit for neural machine translation. *CoRR*, abs/1701.02810, 2017.

[3] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *CoRR*, abs/1611.04230, 2016.

[4] Elvys Linhares Pontes, Stéphane Huet, Juan-Manuel Torres-Moreno, and Andréa Carneiro Linhares. Cross-language text summarization using sentence and multi-sentence compression. In *Natural Language Processing and Information Systems*, pages 467–479. Springer International Publishing, 2018.

[5] Xiaojun Wan, Huiying Li, and Jianguo Xiao. Cross-language document summarization based on machine translation quality prediction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 917–926, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[6] Xiaojun Wan, Fuli Luo, Xue Sun, Songfang Huang, and Jin ge Yao. Cross-language document summarization via extraction and ranking of multiple summaries. *Knowledge and Information Systems*, jan 2018.

[7] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[8] Chi Zhang, Shagan Sah, Thang Nguyen, Dheeraj Peri, Alexander Loui, Carl Salvaggio, and Raymond Ptucha. Semantic sentence embeddings for paraphrasing and text summarization. In *GlobalSIP*, pages 705–709. IEEE, 2017.