Impact Factor:

ISRA (India)	= **3.117**	**SIS** (USA)	= **0.912**	**ICV** (Poland)	= **6.630**
ISI (Dubai, UAE)	= **0.829**	**РИНЦ** (Russia)	= **0.156**	**PIF** (India)	= **1.940**
GIF (Australia)	= **0.564**	**ESJI** (KZ)	= **5.015**	**IBI** (India)	= **4.260**
JIF	= **1.500**	**SJIF** (Morocco)	= **5.667**		

SOI: 1.1/TAS DOI: 10.15863/TAS

International Scientific Journal

Theoretical & Applied Science

p-ISSN: 2308-4944 (print) **e-ISSN:** 2409-0085 (online)

Year: 2018 **Issue:** 12 **Volume:** 68

Published: 28.12.2018 **http://T-Science.org**

SECTION 4. Computer science, computer engineering and automation

QR – Issue QR – Article

Marina Vladimirovna Shkurina
Master's Student
Institute of Computer Science and Technology, Peter the Great St. Petersburg Polytechnic University

Oleg Yurievich Sabinin
Candidate of Engineering Sciences, Docent
Institute of Computer Science and Technology, Peter the Great St. Petersburg Polytechnic University

AN OVERVIEW AND ANALYSIS OF AUTOMATIC TEXT SUMMARIZATION METHODS

Abstract: *With the amount of data online growing each day, automatic text summarization methods are needed to help people navigate through all the information that is available to them. This article provides an overview of automatic text summarization methods. It starts with a brief description of early methods and methods that are used today. Promising paths for future research are presented.*

Key words: *Automatic text summarization, natural language processing, machine learning, neural networks.*

Language: *Russian*

Citation: Shkurina, M. V., & Sabinin, O. Y. (2018). An overview and analysis of automatic text summarization methods. *ISJ Theoretical & Applied Science, 12 (68),* 282-286.

Soi: http://s-o-i.org/1.1/TAS-12-68-41 ***Doi***: crossref https://dx.doi.org/10.15863/TAS.2018.12.68.41

ОБЗОР И АНАЛИЗ МЕТОДОВ АВТОМАТИЧЕСКОГО АННОТИРОВАНИЯ ТЕКСТА

Аннотация: *В связи с тем, что количество данных онлайн растет каждый день, методы автоматического аннотирования необходимы, чтобы помочь людям ориентироваться в информации, которая им доступна. Данная статья предлагает обзор методов автоматического аннотирования. Она начинается с краткого описания первых методов и методов, которые используются сегодня. Представлены перспективные направления дальнейших исследований.*

Ключевые слова: *Автоматическое аннотирование текста, обработка естественного языка, машинное обучение, нейронные сети.*

Введение

На сегодняшний день объемы информации в Интернете и различных хранилищах данных увеличиваются с невероятной скоростью – согласно отчету аналитической фирмы IDC, проспонсированному Seagate, объем данных в 2025 году достигнет отметки в 175 зеттабайт (для сравнения, в 2018 году объем данных составил 33 зеттабайта). [1] При таких условиях становится невозможным детальный просмотр всех доступных источников информации.

Довольно часто для того чтобы определить, содержится ли в тексте действительно важная и необходимая информация, человеку нужно прочитать и изучить этот текст полностью. Если текст довольно длинный, то это может занять большое количество времени. Для некоторых документов составляются аннотации, которые кратко описывают основную суть содержимого, что может значительно помочь при оценке полезности данного текста. Но чаще всего тексты не сопровождаются аннотациями, а ручное аннотирование является довольно трудоемкой задачей. В связи с этим все больше интереса вызывает задача автоматического аннотирования текста. Решение данной задачи позволило бы значительно уменьшить количество информации, которую необходимо обрабатывать человеку, за счет выявления ключевых идей в тексте.

Несмотря на то, что исследования в этой области начались еще в 1950-ые годы, когда возник повышенный интерес к автоматизации

составления аннотаций к технической документации [2], все еще нельзя сказать, что задача полностью решена, в силу сложности и неоднозначности естественного языка.

В данной статье будут кратко рассмотрены ранние работы, посвященные данной проблеме, будет проведен обзор современных методов и перспективных направлений исследований.

Ранние работы

Большая часть ранних работ по автоматическому аннотированию посвящена аннотированию технической документации. Одна из основополагающих работ того времени – работа Луна [2], опубликованная в 1958 году. В этой работе Лун предполагает, что частота встречаемости каждого слова в тексте позволяет оценить его значимость в рамках данного документа. Составляется список встречаемости слов в тексте, отсортированный по убыванию, из которого выбирается некоторое количество наиболее часто встречающихся слов. На основе этого списка для каждого предложения рассчитывается степень его значимости, зависящая от количества значимых слов в предложении и линейного расстояния между ними. Все предложения ранжируются на основе степени значимости, и в аннотацию входят предложения с наиболее высоким рангом.

В ходе другого исследования, опубликованного в том же году, было обнаружено, что в 85% анализируемых параграфов текста предложение с основной мыслью было первым, в 7% - последним. [3] Таким образом, в качестве основного предложения можно выбрать одно из этих двух. Эта характеристика с тех пор используется во многих системах на основе машинного обучения.

Эдмундсон, опубликовавший работу [4] в 1969 году, расширил подход Луна, предположив, что на значимость предложения могут влиять несколько характеристик. Помимо частоты слов, использовавшейся в работе Луна, и позиции предложения в тексте, он также рассматривал наличие ключевых слов (таких как “значительно” и “невозможно”) и структуру документа (является ли предложение названием или заголовком). Для оценки предложений каждой характеристике был вручную присвоен вес. Результаты его исследования говорят о том, что для его конкретной задачи и набора документов частота встречаемости слов является наименее важной характеристикой из четырёх предложенных.

В 1982 году была разработана система FRUMP [5], которая была первой системой, не просто извлекающей предложения из исходного текста, а пытающейся понять документы, написанные на естественном языке. Вручную было составлено 60 сценариев, связанных с определенными событиями (например, теракт, землетрясение, дипломатический визит). Из текста извлекались ключевые слова, на основе которых отдельный модуль предлагал подходящий сценарий. Выбранному сценарию соответствовал заранее заданный шаблон, который заполнялся на основе исходного текста.

В 1995 году Купец, Педерсен и Чень в своей работе впервые применили машинное обучение в задаче автоматического аннотирования текста, разработав систему, которая рассматривала данную задачу как задачу классификации и использовала наивный байесовский классификатор. [6]

Классификация методов

Существуют разные способы классификации методов автоматического аннотирования в зависимости от выбранного критерия. Так, по обрабатываемому количеству документов можно выделить аннотирование одного документа и аннотирование массива документов. [8] По цели использования можно выделить общее аннотирование и аннотирование по запросу. По языку методы делятся на монолингвальные, мультилингвальные и кросслингвальные. Когда язык исходного документа и аннотации совпадает, то метод относится к монолингвальным. Если исходный документ написан на нескольких языках, так же, как и аннотация, то метод является мультилингвальным. Метод относится с кросслингвальным, когда язык исходного документа и язык аннотации различаются.

Наиболее значимая классификация основывается на способе построения текста. Выделяются две группы методов: извлекающие и генерирующие. При использовании извлекающих методов аннотирования из исходного текста выделяются наиболее важные предложения. При этом данные предложения не обрабатываются, и извлекаются в таком порядке, в котором они представлены в тексте. Результатом генерирующих методов является аннотация, включающая в себя слова и фразы, которые отличаются от тех, которые присутствовали в исходном тексте. Таким образом, такая аннотация состоит из идей и концепций, взятых из исходного документа, но они реинтерпретированы и представлены в другой форме.

В данной статье будут рассматриваться монолингвальные методы общего аннотирования одного документа.

Современные методы

Извлекающие методы можно разделить на две большие группы: поверхностные методы,

которые не прибегают к сложному лингвистическому анализу, и глубокие методы. [7]

К поверхностным методам относятся, например, методы из работ Луна и Эдмундсона, а также другие методы, которые для выбора предложений используют некоторые статистические характеристики. К поверхностным также относятся методы на основе графов – TextRank [8] и LexRank [9]. В этих методах документ представляется в виде графа, вершинами которого являются предложения или слова из текста. Веса ребер, соединяющих вершины, отражают степень сходства предложений. Такую связь можно рассматривать как рекомендацию. Предложение, которое связано с некоторой концепцией в тексте, рекомендует читателю другие предложения, которые имеют похожее содержание.

С ростом популярности машинного обучения появлялось все больше работ, использующих его для автоматического аннотирования. В этом случае задача рассматривалась как задача классификации – предложение из текста либо входит в аннотацию, либо не входит. К таким методам можно отнести методы, использующие деревья решений, метод опорных векторов и нейронные сети. В работе [10] использовались скрытые марковские модели, в которых при анализе предложения учитывалось, входит ли предыдущее предложение в аннотацию. Авторы предположили, что введение этой зависимости улучшит итоговый результат по сравнению с методами, использующими наивный байесовский классификатор.

К глубоким относятся, например, методы с использованием латентно-семантического анализа, которые анализируют взаимосвязь между предложениями текста и терминами, содержащимися в них, выявляют тематики, присутствующие в тексте, и в аннотацию выбирается определенное количество предложений от каждой тематики. [11]

К глубоким методам также относятся методы, использующие более сложные архитектуры нейронных сетей. Так, например, работы [12], [13], [14] используют в своих моделях сверточные или рекуррентные нейронные сети.

Для генерирующих методов можно выделить несколько направлений: использование шаблонов, сжатие предложений, полноценная абстракция.

Подходы на основе шаблонов используют заранее подготовленные шаблоны для представления документа. Лингвистические паттерны или правила извлечения используются для заполнения пропусков в этом шаблоне.

Сжимающие методы выполняют извлечение наиболее важных предложений из текста, но либо удаляют из них лишнюю информацию, либо объединяют несколько предложений, пытаясь при этом сохранить связность и смысл текста. Существующие работы на данную тему предлагают различные способы решения этой задачи: в работе [15] авторы используют условные случайные поля (Conditional Random Fields, CRF), чтобы каждому слову предложения присвоить метку 0 – оставить - или 1 – удалить. Работа [16] описывает алгоритм, основанный на построении словесного графа текста и применении алгоритма Дейкстры для поиска кратчайших путей с целью сократить или объединить существующие предложения. В работе [17] исходный документ представляется в виде вложенного дерева, которое состоит из двух видов структур: дерева документа и дерева предложения. Это дерево строится на основе теории риторической структуры.

Последние исследования показывают, что наиболее перспективно для полноценной абстракции выглядит модель кодировщик-декодировщик, которая основывается на использовании рекуррентных нейронных сетей. Такие модели использовали авторы в своих работах [18], [19], [20], [21].

Основные тенденции

В последние годы стали приобретать популярность подходы на основе более сложных видов нейронных сетей как для извлекающих, так и для генерирующих методов аннотирования.

Большинство систем, использующих такие методы, выполняют следующие шаги:

1. слова преобразуются в векторное представление;
2. предложения преобразуются в векторное представление на основе векторных представлений слов;
3. представления предложений передаются модели для выбора предложений (извлекающие методы) или генерации текста (генерирующие методы).

Нейронные сети могут применяться на каждом из этих шагов. На шаге 1 они могут использоваться для получения предобученных таблиц поиска (такие инструменты как Word2Vec, GloVe). На втором шаге нейронные сети могут использоваться в качестве кодировщиков для извлечения признаков предложения/документа. На третьем шаге сети могут применяться для ранжирования/отбора (в извлекающих методах) или как декодировщики (в генерирующих методах).

В задаче автоматического аннотирования используются два вида нейронных сетей: сверточные нейронные сети (СНС) и рекуррентные нейронные сети (РНС). [22]

Перспективы исследований

Лучшие результаты как для извлекающих, так и для генерирующих методов сегодня показывают модели, которые прибегают к использованию СНС или РНС с различными модификациями и улучшениями. Использование их в генерирующих методах представляет особенный интерес.

При этом у методов, в которых применяются РНС, есть существенные недостатки:

1. несмотря на то, что РНС позволяет моделировать «память», если входная последовательность относительно длинная, то на дальнейших итерациях будет происходить эффект «забывания» того, что было в начале последовательности;
2. они не справляются с документами, длина которых превышает несколько тысяч слов, в связи с высокими требованиями к памяти;
3. такие модели медленно обучаются в связи со сложностью архитектуры.

В связи с этими ограничениями открываются направления для дальнейших исследований:

1. использование механизмов внимания, что позволило бы нейронной сети сфокусировать внимание на определенных частях текста;
2. использование методов обучения с подкреплением для обучения модели, например, алгоритм актёр-критик [23], использование которого приводит к уменьшению времени обучения;
3. предварительное сжатие текста для упрощения его дальнейшей обработки. [22]

Оценка аннотации

Оценка автоматически созданной аннотации также является нетривиальной задачей, так как для документа или массива документов не существует идеальной аннотации. [24]

В 2004 году был предложен набор метрик ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [25], который на сегодняшний день и используется для сравнения различных методов и систем.

Наиболее часто для оценки используется метрика ROUGE-N из этого набора, которая зависит от отношения количества совпавших n-грамм для эталонной аннотации и оцениваемой аннотации к количеству n-грамм в эталонной аннотации. Она рассчитывается по формуле (1).

$$ROUGE - N = \frac{\sum_{S \in R} \sum_{g_n \in S} C_m(g_n)}{\sum_{S \in R} \sum_{g_n \in S} C(g_n)}, \qquad (1)$$

где R – множество эталонных аннотаций, g_n – n-грамм длины n, $C_m(g_n)$ – количество n-грамм g_n, совпавших для эталонной и оцениваемой аннотации, $C(g_n)$ – количество n-грамм g_n в эталонной аннотации.

Заключение

В работе были рассмотрены ранние работы в области автоматического аннотирования и современные методы. Были обозначены основные тенденции и перспективы дальнейших исследований для улучшения результатов и вычислительных затрат для реализации описанных методов.

Таким образом, можно сделать следующие выводы:

– существует меньше работ, посвященных генерирующим методам автоматического аннотирования, в силу их сложности;

– появляется тенденция к использованию сверточных и рекуррентных нейронных сетей для решения данной задачи;

– можно выделить направления для дальнейших исследований с целью улучшить модели с использованием СНС и РНС.

References:

1. Reinsel, D., Gantz, J., & Rydning, J. (2018). *The Digitization of the World From Edge to Core.*
2. Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development, Vol. 2, № 2,* 159-165.
3. Baxendale, P. B. (1958). Machine-Made Index for Technical Literature-An Experiment. *IBM Journal of Research and Development, Vol. 2, № 4,* 354–361.
4. Edmundson, H. P. (1969). New Methods in Automatic Extracting. *Journal of the ACM, Vol. 16, № 2,* 264–285.
5. DeJong, G. (1982). *An Overview of the FRUMP System Strategies for Natural Language Processing.* In: Lehnert W., Ringle M.H. (Eds.). (pp.149-176). Lawrence Erlbaum.

6. Kupiec, J., Pedersen, J., & Chen, F. (1995). *A trainable document summarizer*. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '95. (pp.68-73). New York, USA: ACM Press
7. Radev, D. R., Hovy, E., & McKeown, K. (2002). Introduction to the Special Issue on Summarization. *Computational Linguistics, Vol. 28, № 4*, 399–408.
8. Mihalcea, R., & Tarau, P. (2004). *TextRank: Bringing Order into Texts*. Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A Meeting of Sigdat, A Special Interest Group of the Acl, Held in Conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain.
9. Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research, Vol. 22*, 457–479.
10. Conroy, J. M., & O'leary, D. P. (2001). *Text summarization via hidden Markov models*. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '01. (pp.406-407). New York, USA: ACM Press.
11. Ozsoy, M. G., Alpaslan, F. N., & Cicekli, I. (2011). Text summarization using Latent Semantic Analysis. *Journal of Information Science, Vol. 37, № 4*, 405–417.
12. Nallapati, R., Zhai, F., & Zhou, B. (2016). *SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents.*
13. Cheng, J., & Lapata, M. (2016). *Neural Summarization by Extracting Sentences and Words.*
14. Cao, Z., et al. (2015). *Learning summary prior representation for extractive summarization.* Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). Vol. 2. (pp. 829–833).
15. Li, C., et al. (2013). *Document Summarization via Guided Sentence Compression*. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics. (pp. 490–500).
16. Lloret, E., & Palomar, M. (2011). *Analyzing the use of word graphs for abstractive text summarization*. Proceedings of the First International Conference on Advances in Information Mining and Management. (pp.61-66). Barcelona, Spain.
17. Kikuchi, Y., et al. (2014). *Single Document Summarization based on Nested Tree Structure*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). (pp.315-320). Stroudsburg, PA, USA: Association for Computational Linguistics.
18. Rush, A. M., Chopra, S., & Weston, J. (2015). *A Neural Attention Model for Abstractive Sentence Summarization.* Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. (pp.379-389). Stroudsburg, PA, USA: Association for Computational Linguistics.
19. Nallapati, R., et al. (2016). *Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond.*
20. See, A., Liu, P. J., & Manning, C. D. (2017). *Get To The Point: Summarization with Pointer-Generator Networks*. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). (pp.1073-1083). Stroudsburg, PA, USA: Association for Computational Linguistics.
21. Paulus, R., Xiong, C., & Socher, R. (2017). *A Deep Reinforced Model for Abstractive Summarization.*
22. Lecun, Y., et al. (1998). *Gradient-based learning applied to document recognition.* Proceedings of the IEEE. Vol. 86, № 11. (pp. 2278–2324).
23. Li, P., Bing, L., & Lam, W. (2018). *Actor-Critic based Training Framework for Abstractive Summarization.*
24. Das, D., & Martins, A. F. T. (2007). *A Survey on Automatic Text Summarization Eighth ACIS.* International Conference on Software Engineering Artificial Intelligence Networking and ParallelDistributed Computing SNPD 2007. Vol. 4. (pp.574–578).
25. Lin, C. Y. (2004). *ROUGE: A Package for Automatic Evaluation of summaries.* Conference: In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004).