СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ ЭКСТРАКЦИОННОГО РЕФЕРИРОВАНИЯ ОДИНОЧНЫХ ДОКУМЕНТОВ

Новосёлова А. М., студентка кафедры информационно-аналитических систем СПБГУ, novonastya96@gmail.com

Аннотация

В данной работе проведен обзор двух алгоритмов автоматического реферирования текстов, а также способов оценки их качества. Представлено сравнение этих алгоритмов по нескольким метрикам в применении к набору русскоязычных новостных текстов.

Введение

Задача автоматического реферирования текстов очень популярна среди исследователей. Существует большое количество публикаций, в которых описываются различные алгоритмы автоматического реферирования.

Однако, различные авторы используют различные метрики для оценки предложенных ими алгоритмов. Кроме того, оценка алгоритмов производится, в основном, на англоязычных наборах документов. В связи с этим применение алгоритмов автоматического реферирования текстов к русскоязычному набору документов и их сравнение является актуальной задачей.

В данной работе рассмотрены алгоритмы экстракционного реферирования. Этот тип реферирования основан на выделении из первичных документов ключевых предложений, которые добавляются в реферат без изменений в порядке их появления в тексте. Обобщенно схему экстракционного реферирования можно представить следующим образом:

1. Предварительная обработка изначального документа: удаление стоп-слов, стемминг, разбиение текста на предложения
2. Присвоение определенного веса каждому предложению с помощью какого-либо алгоритма
3. Формирование реферата из предложений с наибольшим весом

Описание алгоритмов

В данном разделе описаны алгоритмы, выбранные для сравнения. Результатом их работы является присвоение каждому предложению исходного текста определенного веса.

В работе [1] был предложен алгоритм TextRank, основанный на построении графа. Он заключается в следующем:

1. По тексту строится взвешенный неориентированный граф, вершины в котором обозначают предложения текста. Весом ребра между двумя вершинами является степень схожести двух предложений, соответствующих вершинам. Она вычисляется, как количество совпадающих слов в предложениях, нормированное суммарной длиной этих предложений.
2. С помощью итерационного процесса каждой вершине графа присваивается вес, исходя из весов ребер.

В работе [2] был предложен алгоритм LSA, основанный на сингулярном разложении матрицы. Он заключается в следующем:

1. По тексту строится матрица терм-предложение, размер которой равен $n \times m$, где n - количество уникальных слов текста, m - количество предложений текста. Элемент a_{ij} этой матрицы равен частоте встречаемости слова i в тексте, если слово i встречается в предложении j, и 0 в противном случае.
2. К полученной матрице применяется сингулярное разложение. По полученному разложению вычисляются веса предложений.

Оценка качества алгоритмов

Для оценки качества работы алгоритмов необходим набор документов, с приложенными к ним рефератами, называемыми образцовыми.

В работе [3] был предложен пакет оценки качества алгоритмов автоматического реферирования текстов ROUGE, метрики которого имеют высокую корреляцию с человеческими оценками.

Далее рассмотрены основные метрики данного пакета.

- ROUGE - N

$$ROUGE - N = \frac{Count_{match}(gram_n)}{Count_{ref}(gram_n)},$$

$Count_{match}(gram_n)$ - количество *n*-грамм, появляющихся и в полученном алгоритмом реферате, и в образцовом; $Count_{ref}(gram_n)$ - количество *n*-грамм в образцовом реферате.

Теперь пусть *X* - образцовый реферат длины *n*, *Y* - реферат длины *m*, полученный алгоритмом.

- ROUGE - L

  Пусть *LCS(X, Y)* - длина наибольшей общей подпоследовательности между *X* и *Y*, где *X*, *Y* рассматриваются как последовательности слов. Тогда:

$$P_{lcs} = \frac{LCS(X,Y)}{m}$$

$$R_{lcs} = \frac{LCS(X,Y)}{n}$$

$$ROUGE - L = \frac{2P_{lcs}R_{lcs}}{P_{lcs} + R_{lcs}}$$

- ROUGE - S

  Пусть *SKIP2(X, Y)* - количество совпадений биграмм с пропусками между *X* и *Y*. Тогда:

$$P_{skip2} = \frac{SKIP2(X,Y)}{C(m,2)}$$

$$R_{skip2} = \frac{SKIP2(X,Y)}{C(n,2)}$$

$$ROUGE - S = \frac{2P_{skip2}R_{skip2}}{P_{skip2} + R_{skip2}}$$

Эксперимент

Описанные алгоритмы и модуль оценки были реализованы на языке Python с использованием оптимизированных библиотек. Для стемминга

был использован Snowball stemmer, список русских стоп-слов был взят из библиотеки stop_words для Python.

Оценка была проведена на наборе, состоящем из 35300 новостей, собранных на различных новостных сайтах, с приложенными аннотациями.

Оценка производилась с помощью четырех метрик, каждая из которых была вычислена на полученном и образцовом рефератах в чистом виде, с проведением стемминга в рефератах, а также с проведением стемминга и удалением стоп-слов.

Были получены следующие результаты:

	TextRank			**LSA**		
	basic	stem	stem, no stop-words	basic	stem	stem, no stop-words
ROUGE-1	0.22	**0.29**	**0.25**	0.22	0.28	0.24
ROUGE-2	0.07	**0.09**	0.08	0.07	0.08	0.08
ROUGE-L	**0.11**	**0.14**	**0.12**	0.10	0.13	0.11
ROUGE-S	0.03	0.05	**0.04**	0.03	0.05	0.03

Из полученных оценок видно, что TextRank превосходит LSA на используемом наборе данных. Причем наибольшее превосходство наблюдается при оценках с проведением стемминга и удалением стоп-слов, которые являются более объективными с точки зрения человека.

Заключение

В данной работе было проведено сравнение двух алгоритмов (TextRank и LSA) автоматического реферирования текстов в применении к русскоязычному набору новостных документов. В ходе оценки алгоритм TextRank показал лучшие результаты.

Литература

[1] R. Mihalcea and P. Tarau, “TextRank: Bringing Order into Texts,” in *Proc. of the 9th Conf. on Empirical Methods in Natural Language Processing*, 2004, pp. 404–411.

[2] J. Steinberger and K. Jezek, “Using Latent Semantic Analysis in Text Summarization and Summary Evaluation,” in *Proc. of ISIM*, 2004, pp. 93–100.

[3] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Proceedings of ACL Text Summarization Branches Out Workshop*, 2004, pp. 74–81.