**Finding Keywords of an Article in a Corpus Using Log Word Rank Movement**

Yitong SUN1 and Siew Ann CHEONG2,3

1National Junior College, 37 Hillcrest Road, Singapore 288913

2Division of Physics and Applied Physics, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371

3Complexity Institute, Academic Building North Level 1 Section B Unit No. 7 (ABN-01B-07), 61 Nanyang Drive, Singapore 637335

**Abstract**

**Introduction**

The most common method to find keywords is the term frequency-inverse document frequency (TF-IDF) method (Salton, G., Wong, A. and Yang, C.S., 1975. A vector space model for automatic indexing. Communications of the ACM, 18(11), pp.613-620.). Similar to TF-IDF includes Jones, K.S., 2004. A statistical interpretation of term specificity and its application in retrieval. Journal of documentation. Andrade, M.A. and Valencia, A., 1998. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics (Oxford, England), 14(7), pp.600-607.

Other keyword finding methods?

All these methods require us to first filter out stop words, because these have high frequencies.

The performance will depend on the list of stop words

Here we introduce a simple method based on log word rank movement that does not require initial filtering of stop words

Methods that utilizes linguistic information include Hulth, Anette. "Improved automatic keyword extraction given more linguistic knowledge." In Proceedings of the 2003 conference on Empirical methods in natural language processing, pp. 216-223. Association for Computational Linguistics, 2003.

Methods that do not require a corpus include Matsuo, Y. and Ishizuka, M., 2004. Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools, 13(01), pp.157-169.

Not text summarization. Nenkova, A. and McKeown, K., 2012. A survey of text summarization techniques. In Mining text data (pp. 43-76). Springer, Boston, MA.

In this paper, we compare our word rank movement and 2-gram rank movement methods against TF-IDF (corpus-based) and RAKE (not corpus-based).

**Method**

What is the rank of a word in the corpus?

What is the rank of a word in a document?

From Zipf’s Law, we know that the probability of a word appearing is inversely proportional to its rank, P(n) = A/n.

We are sensitive to large changes in this probability.

So if P0(n) is the probability that word n appear in the corpus, and P1(n) is the probability that the word n appears in the document, P1(n)/P0(n) or log P1(n)/P0(n) will tells how large this change is.

proportional to rank of word in corpus minus rank of word in document. We call this difference the log word rank movement Delta.

If Delta < -1, then the word is at least e times more frequent in the document than in the corpus.

If it is more negative, then the change is even greater.

Compare against RAKE and TF-IDF.

RAKE from rake\_nltk module installed using pip, Rake class uses stopwords for English from NLTK, and all punctuation characters

Using functions extract\_keywords\_from\_text()

Output is a list of keyword phrases, ordered from the most important to the least important

We implemented TF-IDF on our own. See Python script in appendix. Keywords are individual tokens that have the largest TFIDF score.

**Data Sets**

*Reuters data set*

*ACM Transactions on the Web*

ACM Transactions …

Finding Keywords

A keyword should be one with very negative log word rank movement.

But how negative must this be in order for us to be sure that the word identified can be used as a keyword?

Statistical Test: Each document contains on the order of 100 distinct words, whereas the corpus contains on the order of 10,000 to 100,000 distinct words.

Suppose a document has n tokens, not all distinct. For example, ‘the’ appears between 10 and 20 times in a typical document. There are m distinct words in the n tokens.

If we randomly sample from the corpus n tokens, how many distinct words will we find, and what will their frequencies be?

This random sampling of n tokens from the corpus is our null model.

If we repeat this random sampling or times, we would end with a histogram of frequencies for each word in the corpus.

Instead of frequencies, we can also measure the word ranks = 10^6 or 10^9 ranks of all words in the corpus if we sample only n tokens.

The basic idea is that small log word rank movements can be by chance, but large ones cannot be!

Through statistical testing, we will know whether the rank of a word is statistically significant, or it appears in the document by chance.

The opposite of by chance is by choice = keyword.

Once we understand, we can relax this statistical testing procedure to more heuristic choices.

**Results**

How to show?

ACM Transactions on the Web = very technical. The same procedure works, but only for appropriate corpora. This shows how we should choose the corpus, against similar documents.

ACM-TRANS-TWEB/ACM-TRANS-WEB-WordRankPlot.pdf ACM-TRANS-TWEB/ACM-TRANS-WEB-BigramRankPlot.pdf

Figure XX. Frequency-rank plots of (left) words and (right) bigrams from the ACM Transactions on the Web corpus of abstracts. In these plots, we also show the best fits to power laws. In the case of bigrams, the fit used only the 10th to 10,000th bigrams.

When we plot the word frequency against the word rank for the 6,183 distinct words. 54,635 tokens in total in the abstracts. Close to Zipf’s Law. Simple linear regression gives the exponent to be , but this is too steep. The actual exponent if we neglect part of the tail is closer to . In contrast, the exponent of frequency-rank plot for bigrams of is significantly different from the Zipf’s law exponent of . If bigrams are constructed by randomly sampling pairs of words from the distribution of words, the exponent will remain close to . This shows how strong the correlation between words that make up actual bigrams must be, for the exponent to have drop by so much.

First abstract of ACM Transactions on the Web.

|  |  |  |  |
| --- | --- | --- | --- |
| lWRM1 () | lWRM2 | RAKE | TF-IDF () |
| tiers: , queues: , multitier: , provisioning: , workloads: , response: , times: , application: , average: , model: , caching: , behavior: , where: , applications: , performance: , Our: | ‘average response’: , ‘caching at’: , ‘tiers.’: , ‘the behavior’: , ‘response times’: , ‘our model’: , ‘the application’: , ‘Our experiments’: , ‘where the’: , | 'model using real multitier applications running', 'since many internet applications employ', 'request arrival rate increased', 'observed average response times', 'maintain response time targets', 'load imbalances across replicas', 'average response times predicted', 'dynamic provisioning technique employing', 'significantly different performance characteristics', 'queues represent different tiers', 'nearly 4200 requests', 'model successfully handles', 'model faithfully captures', 'linux server cluster', 'dynamic capacity provisioning', '95 confidence intervals', 'resource utilization ---', 'experiments also demonstrate', 'two separate tiers', 'multitier architecture', 'model based', 'experiments indicate', 'cpu ---', 'intermediate tiers', 'performance prediction', 'application tiers', 'tier replication', 'sufficiently general', 'near saturation', 'comprehensive range', 'bottleneck identification', 'applications', 'analytically modeling', 'application idiosyncrasies', 'session policing', 'one scenario', 'based workloads', 'model', 'queues', 'tiers', 'performance', 'two', 'capacity', 'application', 'workloads', 'session', 'one', 'within', 'variety', 'validate', 'utility', 'scenarios', 'respectively', 'problem', 'present', 'number', 'network', 'minute', 'less', 'increasing', 'including', 'ii', 'furthermore', 'focus', 'factors', 'configurations', 'capture', 'caching', 'behavior', 'article', 'able', '5', '3', '2', '1500', '0' | model  tiers  applications  response  performance  Our  application  provisioning  capacity  dynamic  times  average  one  two  --  experiments  caching  workloads  different  queues  We  behavior  multitier |

Second abstract of ACM Transactions on the Web.

|  |  |  |  |
| --- | --- | --- | --- |
| LWRM1 (count > 1) | LWRM2 (count > 1) | RAKE | TF-IDF |
| sponsored: , nonsponsored: , links: , relevance: , e-commerce: , analyzed: , %: , relevant: , engines: , campaigns: , major: , average: , business: , queries: , ratings: , search: | ‘sponsored search’: , ‘sponsored and’: , ‘sponsored links’: , ‘of sponsored’: , ‘business model’: , ‘and nonsponsored’: , ‘for sponsored’: , ‘e-commerce queries’: , ‘links for’: , ‘relevance ratings’: , ‘nonsponsored links’: , ‘analyzed the’: , ‘search campaigns’: , ‘ratings for’: , ‘links are’: , ‘%)’: , ‘the relevance’: , ‘search engines’: , ‘links from’: , ‘for Web’: , ‘Web search’: , | 'sponsored links providing online consumers', 'three major web search engines', 'used 108 ecommerce queries', 'large sponsored search campaigns', 'web search engines', 'major search engines', 'sponsored search campaigns', 'yahoo !, google', 'underlying information needs', 'term business model', 'predominant business model', 'deriving five categorizations', 'average sponsored link', '256 retrieved links', 'evaluating sponsored links', 'finding relevant information', 'average relevance ratings', 'sponsored search', 'sponsored links', 'nonsponsored links', 'relevance ratings', 'specific queries', 'commerce queries', 'yearly revenue', 'various viewpoints', 'statistically higher', 'results show', 'relevant choices', 'relevance measures', 'related issues', 'primary basis', 'generates billions', 'distant third', '62 %)', '48 %).', '33 %)', '2 %).', 'qualitatively analyzed', 'sponsored', 'links', 'queries', 'relevant', 'relevance', 'analyzed', 'url', 'title', 'summary', 'services', 'searchers', 'products', 'product', 'prevalent', 'practically', 'organizations', 'msn', 'mechanism', 'long', 'investigating', 'implications', 'gauge', 'effectiveness', 'e', 'discuss', 'appears', 'although', 'address', 'addition', '8' | sponsored  links  search  queries  relevance  engines  We  relevant  Web  campaigns  information  analyzed  ratings  average  major  e-commerce  nonsponsored  model  business  The |

**Compare LWRM1 and TFIDF**

LWRM1: 'sponsored': -2.85, 'nonsponsored': -2.07, 'links': -1.59, 'relevance': -1.48, 'e-commerce': -1.40, 'analyzed': -1.35, '%': -1.25, 'relevant': -1.18, 'engines': -1.17, 'campaigns': -1.14, 'major': -1.12, 'average': -1.03, 'business': -0.93, 'queries': -0.92, 'ratings': -0.86, 'search': -0.70

TF-IDF: sponsored, links, search, queries, relevance, engines, We, relevant, Web, campaigns, information, analyzed, ratings, average, major, e-commerce, nonsponsored, model, business, …

The simplest comparison would be the number of words in common, ignoring the ordering of the words. This would be 14 words in common, out of the first 16 words in LWRM1, and the first 19 words in TF-IDF.

However, we do not know a priori how many words from each list to compare. As this list grows longer, we may have more matches. One thing we can do would be to measure the proportion of matches as a function of the length of the list.

|  |  |  |
| --- | --- | --- |
| LWRM1 | TFIDF | Number of Matches |
| sponsored | sponsored | 1 |
| nonsponsored | links | 1 |
| links  relevance  e-commerce  analyzed: , %: , relevant: , engines: , campaigns: , major: , average: , business: , queries: , ratings: , search: | search  queries  relevance  engines  We  relevant  Web  campaigns  information  analyzed  ratings  average  major  e-commerce  nonsponsored  model  business  The | 2  2  3 |

From the figure shown below,

ACM-TRANS-TWEB/LWRM1vTDIDFabs1.pdf

we see that the proportion of matches start high (because the first words coincide), and then fluctuate between 0.5 and 0.6, before climbing to a maximum of 0.82, before starting to fall again. This maximum corresponds to 16 matching words.

Is the order important? If the order is important we can calculate correlations between the two lists.

If we compute the Pearson correlation between the two list, we find that it is .

If we compute the Kendall tau correlation between the two list, we find that it is .

Both correlations are between -1 and 1.

Generally correlations are low, which tells us that the ordering of the words by the different methods are different.

compareLWRM1TFIDFhist.pdf

Figure XX. Histogram of the maximum proportion of matching keywords between LWRM1 and TFIDF for the 224 abstracts from ACM Transactions on the Web.

As we can see from Figure XX, the maximum proportion of matching keywords between LWRM1 and TFIDF are between 0.7 and 1.0, if we ignore the orderings of the keywords by the two methods.

**Compare LWRM1 and LWRM2**

LWRM1: 'sponsored': -2.85, 'nonsponsored': -2.07, 'links': -1.59, 'relevance': -1.48, 'e-commerce': -1.40, 'analyzed': -1.35, '%': -1.25, 'relevant': -1.18, 'engines': -1.17, 'campaigns': -1.14, 'major': -1.12, 'average': -1.03, 'business': -0.93, 'queries': -0.92, 'ratings': -0.86, 'search': -0.70

LWRM2: ‘sponsored search’: , ‘sponsored and’: , ‘sponsored links’: , ‘of sponsored’: , ‘business model’: , ‘and nonsponsored’: , ‘for sponsored’: , ‘e-commerce queries’: , ‘links for’: , ‘relevance ratings’: , ‘nonsponsored links’: , ‘analyzed the’: , ‘search campaigns’: , ‘ratings for’: , ‘links are’: , ‘%)’: , ‘the relevance’: , ‘search engines’: , ‘links from’: , ‘for Web’: , ‘Web search’: ,

Two ways to compare: (1) word-based, and (2) bigram-based.

For word-based comparison, we compare two lists of the same length from LWRM1 and LWRM2. For LWRM2, we decompose the list of bigrams into a set of words. For example, {'sponsored', 'nonsponsored', 'links', 'relevance', 'e-commerce'} are the first five words from LWRM1. The first five bigrams from LWRM2 are {‘sponsored search’, ‘sponsored and’, ‘sponsored links’, ‘of sponsored’, ‘business model’}. This corresponds to the set of words {‘sponsored’, ‘search’, ‘and’, ‘links’, ‘of’, ‘business’, ‘model’}. For these lists of length 5, we find two matches, ‘sponsored’ and ‘links’.

If we now go to lists of length 10, we have {'sponsored', 'nonsponsored', 'links', 'relevance', 'e-commerce', 'analyzed', '%', 'relevant', 'engines', 'campaigns'} from LWRM1, and {‘sponsored search’, ‘sponsored and’, ‘sponsored links’, ‘of sponsored’, ‘business model’, ‘and nonsponsored’, ‘for sponsored’, ‘e-commerce queries’, ‘links for’, ‘relevance ratings’} for LWRM2, which gives the set of words {‘sponsored’, ‘search’, ‘and’, ‘links’, ‘of’, ‘business’, ‘model’, ‘nonsponsored’, ‘for’, ‘e-commerce’, ‘queries’, ‘relevance’, ‘ratings’}. For these lists of length 10, we find 5 matches.

LWRM1 agrees best with TFIDF at length 16. At this length, the LWRM1 list is {'sponsored', 'nonsponsored', 'links', 'relevance', 'e-commerce', 'analyzed', '%', 'relevant', 'engines', 'campaigns', 'major', 'average', 'business', 'queries', 'ratings', 'search'}. The LWRM2 list is {‘sponsored search’, ‘sponsored and’, ‘sponsored links’, ‘of sponsored’, ‘business model’, ‘and nonsponsored’, ‘for sponsored’, ‘e-commerce queries’, ‘links for’, ‘relevance ratings’, ‘nonsponsored links’, ‘analyzed the’, ‘search campaigns’, ‘ratings for’, ‘links are’, ‘%}’}, and the set of words is {‘sponsored’, ‘search’, ‘and’, ‘links’, ‘of’, ‘business’, ‘model’, ‘nonsponsored’, ‘for’, ‘e-commerce’, ‘queries’, ‘relevance’, ‘ratings’, ‘analyzed’, ‘the’, ‘campaigns’, ‘are’}. 11 matches. Proportion of match would be 11/16 = 0.69.

For bigram-based comparison of length 10, we have {'sponsored', 'nonsponsored', 'links', 'relevance', 'e-commerce', 'analyzed', '%', 'relevant', 'engines', 'campaigns'} from LWRM1, and {‘sponsored search’, ‘sponsored and’, ‘sponsored links’, ‘of sponsored’, ‘business model’, ‘and nonsponsored’, ‘for sponsored’, ‘e-commerce queries’, ‘links for’, ‘relevance ratings’} for LWRM2, we find 6 matches at the bigram level, and 1 bigram complete match, 6 bigrams partial match.

LWRM2: sponsored = (1 + 2 + 3 + 6)/4 = 3, links = (3 + 9 + 11 + 15 + 18)/5 = 11.2, relevance = (10 + 16)/2 = 13, engines = 17, queries = 8, search = (1 + 13 + 17 + 20)/4 = 12.75

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Length | 1 | 2 | 3 | 4 | 5 |
| LWRM2 | ‘sponsored search’ | ‘sponsored and’ | ‘sponsored links’ | ‘of sponsored’ | ‘business model’ |
| LWRM1 | 'sponsored' | 'nonsponsored' | 'links' | 'relevance' | 'e-commerce' |
| Cumulative Matches | 0.5 | 0.5 + 0.5 = 1.0 | 0.5 + 0.5 + 1.0 = 2.0 | 0.5 + 0.5 + 1.0 + 0.5 = 2.5 | 0.5 + 0.5 + 1.0 + 0.5 + 0.0 = 2.5 |

Pearson correlation for the average rank of LWRM2 against LWRM1, 0.479.

Kendall tau correlation for the average rank of LWRM2 against LWRM1, 0.333.

compareLWRM1LWRM2hist.pdf

Figure XX. Histogram of maximum proportion of matching words between LWRM1 and LWRM2.

The histogram extends towards lower maximum proportion of matching words, because LWRM2 is able to identify bigrams with significant log rank movement, but not containing words that have significant log rank movement.

compareLWRM1LWRM2bigramhist.pdf

Here we see that the histogram shifts towards lower proportion of matching bigrams. In fact, the histogram peaks around 0.5, which suggests that on average, one word in the bigrams of LWRM2 is matched by the keywords identified by LWRM1.

Summary of Word-Based Comparisons

|  |  |  |  |
| --- | --- | --- | --- |
|  | LWRM2 | TFIDF | RAKE |
| LWRM1 | compareLWRM1LWRM2hist.pdf | compareLWRM1TFIDFhist.pdf | compareLWRM1RAKEhist.pdf |
| LWRM2 |  | compareTFIDFLWRM2hist.pdf | compareRAKELWRM2hist.pdf |
| TFIDF |  |  | compareTFIDFRAKEhist.pdf |

LWRM1 very good agreement with TFIDF, which is a word-based algorithm, and reasonable agreement with LWRM2 and RAKE, which are substring-based algorithms (restricted to bigrams) in this comparison.

LWRM2 reasonable agreement with LWRM1 and TFIDF, which are word-based algorithms, but poor agreement with RAKE. Let us see what the agreement is like at the bigram level.

Summary of Bigram-Based Comparisons

|  |  |  |  |
| --- | --- | --- | --- |
|  | LWRM2 | TFIDF | RAKE |
| LWRM1 | compareLWRM1LWRM2bigramhist.pdf | - | compareLWRM1RAKEbigramhist.pdf |
| LWRM2 |  | compareTFIDFLWRM2bigramhist.pdf | compareRAKELWRM2bigramhist.pdf |
| TFIDF |  |  | compareTFIDFRAKEbigramhist.pdf |

Reasonable agreement LWRM1 and TFIDF with LWRM2 at the bigram level, less so with RAKE. Poor agreement between LWRM2 and RAKE even at the bigram level. Understandable, because RAKE is corpus-independent, while the other algorithms are corpus-dependent.

**Reuters News Dataset**

Reuters = keywords useful to classify the news into categories, business news, keyword set is like a summary of the news. Somewhat non-technical.

For this dataset, we find 74,634 unique words, and 2,670,066 occurrences. We also find 627,524 unique bigrams, and 2,651,964 occurrences.

ReutersWordRankPlot.pdf ReutersBigramRankPlot.pdf

Figure XX. Frequency-rank plots of (left) words and (right) bigrams from the Reuters News corpus. In these plots, we also show approximate fits to exponentially-truncated power laws, . For words, we find that , , and , whereas for bigrams, we find , , and .

Next, let us look at how well the LWRM method works for this news dataset. The first news item in this dataset reads:

'Showers continued throughout the week in the Bahia cocoa zone, alleviating the drought since early January and improving prospects for the coming temporao, although normal humidity levels have not been restored, Comissaria Smith said in its weekly review. The dry period means the temporao will be late this year. Arrivals for the week ended February 22 were 155,221 bags of 60 kilos making a cumulative total for the season of 5.93 mln against 5.81 at the same stage last year. Again it seems that cocoa delivered earlier on consignment was included in the arrivals figures. Comissaria Smith said there is still some doubt as to how much old crop cocoa is still available as harvesting has practically come to an end. With total Bahia crop estimates around 6.4 mln bags and sales standing at almost 6.2 mln there are a few hundred thousand bags still in the hands of farmers, middlemen, exporters and processors. There are doubts as to how much of this cocoa would be fit for export as shippers are now experiencing dificulties in obtaining +Bahia superior+ certificates. In view of the lower quality over recent weeks farmers have sold a good part of their cocoa held on consignment. Comissaria Smith said spot bean prices rose to 340 to 350 cruzados per arroba of 15 kilos. Bean shippers were reluctant to offer nearby shipment and only limited sales were booked for March shipment at 1,750 to 1,780 dlrs per tonne to ports to be named. New crop sales were also light and all to open ports with June/July going at 1,850 and 1,880 dlrs and at 35 and 45 dlrs under New York july, Aug/Sept at 1,870, 1,875 and 1,880 dlrs per tonne FOB. Routine sales of butter were made. March/April sold at 4,340, 4,345 and 4,350 dlrs. April/May butter went at 2.27 times New York May, June/July at 4,400 and 4,415 dlrs, Aug/Sept at 4,351 to 4,450 dlrs and at 2.27 and 2.28 times New York Sept and Oct/Dec at 4,480 dlrs and 2.27 times New York Dec, Comissaria Smith said. Destinations were the U.S., Covertible currency areas, Uruguay and open ports. Cake sales were registered at 785 to 995 dlrs for March/April, 785 dlrs for May, 753 dlrs for Aug and 0.39 times New York Dec for Oct/Dec. Buyers were the U.S., Argentina, Uruguay and convertible currency areas. Liquor sales were limited with March/April selling at 2,325 and 2,380 dlrs, June/July at 2,375 dlrs and at 1.25 times New York July, Aug/Sept at 2,400 dlrs and at 1.25 times New York Sept and Oct/Dec at 1.25 times New York Dec, Comissaria Smith said. Total Bahia sales are currently estimated at 6.13 mln bags against the 1986/87 crop and 1.06 mln bags against the 1987/88 crop. Final figures for the period to February 28 are expected to be published by the Brazilian Cocoa Trade Commission after carnival which ends midday on February 27.'

A quick human reading of this news suggests that the news is about improved production of cocoa in the Bahia region, after the drought ended in January, and rain expected through much of February and March. The keywords identified by the four methods are shown in Table XX.

Table XX. Keywords of the first Reuters news item identified by the four methods being compared.

|  |  |  |  |
| --- | --- | --- | --- |
| LWRM1 (count > 1) | LWRM2 (count > 1) | RAKE | TF-IDF |
| 'Comissaria': , 'June/July': 'Aug/Sept': , 'March/April': , 'Bahia': , 'Oct/Dec': , '2.27': ,  'Smith': , 'times': ,  'bags': ,  'ports': , 'cocoa': ,  '1.25': ,  'Dec': , | ('times', 'New'): , ('Comissaria', 'Smith'): ,  ('.', 'Comissaria'): ,  (',', 'Comissaria'): ,  (',', 'Aug/Sept'): ,  ('York', 'Dec'): ,  ('Aug/Sept', 'at'): ,  ('on', 'consignment'): ,  ('and', '1,880'): ,  ('1.25', 'times'): ,  ('open', 'ports'): ,  ('at', '2.27'): , (',', 'June/July'): ,  ('1,880', 'dlrs'): ,  ('2.27', 'times'): ,  ('Dec', ','): , ('Smith', 'said'): ,  ('Uruguay', 'and'): ,  ('June/July', 'at'): ,  ('York', 'Sept'): ,  ('and', 'Oct/Dec'): ,  ('Sept', 'and'): ,  ('currency', 'areas'): ,  ('Oct/Dec', 'at'): ,  ('bags', 'against'): ,  ('at', '1.25'): , ('to', 'how'): , (',', 'Uruguay'): ,  ('and', 'at'): , ('mln', 'bags'): ,  ('sales', 'were'): ,  ('as', 'to'): , ('New', 'York'): ,  ('how', 'much'): ,  ('May', ','): , ('were', 'the'): ,  ('is', 'still'): , ('dlrs', 'and'): , ('per', 'tonne'): ,  ('the', 'week'): ,  ('dlrs', 'for'): , ('U.S.', ','): , | [missing] | **'dlrs', 'New', 'York', 'times', 'sales', 'crop', 'mln', 'bags', 'said', 'Smith', 'Comissaria', 'cocoa',** '1.25', 'Dec', 'Oct/Dec', '2.27', 'March/April', 'Aug/Sept', 'June/July', 'ports', 'per', 'still', 'February', 'Bahia', |

As we can see, the set of LWRM1 keywords are largely similar to the set of TFIDF keywords. The most obvious differences are ‘dlrs’, ‘New’, and ‘York’ being absent from the set of LWRM1 keywords. This is understandable, because ‘dlrs’ is a Reuters-specific abbreviation for ‘delivers’, so it appears in the set of TFIDF keywords. However, because the word rank of ‘dlrs’ is high in the Reuters dataset, its log rank movement in the first news item is not significant. Probably similar reasons for ‘New’ and ‘York’.

We are now ready to compare the four methods, at the word-based and bigram-based levels. Only 5,956 news items, because most of them were too short for our maximum matching probability.

Word-based

|  |  |  |  |
| --- | --- | --- | --- |
|  | LWRM2 | TFIDF | RAKE |
| LWRM1 | NewsUpdated/compareReutersLWRM1LWRM2hist.pdf | NewsUpdated/compareReutersLWRM1TFIDFhist.pdf | NewsUpdated/compareReutersLWRM1RAKEhist.pdf |
| LWRM2 |  | NewsUpdated/compareReutersLWRM2TFIDFhist.pdf | NewsUpdated/compareReutersLWRM2RAKEhist.pdf |
| TFIDF |  |  | NewsUpdated/compareReutersTFIDFRAKEhist.pdf |

Word-based comparison, good agreement between LWRM1, LWRM2, and TFIDF. Decent agreement between LWRM1, TFIDF and RAKE. Poor agreement between LWRM2 and RAKE.

Bigram-based

|  |  |  |  |
| --- | --- | --- | --- |
|  | LWRM2 | TFIDF | RAKE |
| LWRM1 | NewsUpdated/compareReutersLWRM1LWRM2bigramhist.pdf |  | NewsUpdated/compareReutersLWRM1RAKEbigramhist.pdf |
| LWRM2 |  | NewsUpdated/compareReutersLWRM2TFIDFbigramhist.pdf | NewsUpdated/compareReutersLWRM2RAKEbigramhist.pdf |
| TFIDF |  |  | NewsUpdated/compareReutersTFIDFRAKEbigramhist.pdf |

Bigram-based comparison, good agreement between LWRM1 and LWRM2, and decent agreement between TFIDF and LWRM2. Generally poor agreement between LWRM1, LWRM2, TFIDF and RAKE.

ACM Transactions …

Medicine or Chemistry abstracts

**Test of Statistical Significance**

Second abstract of ACM Transactions on the Web: sponsored: , nonsponsored: , links: , relevance: , e-commerce: , analyzed: , %: , relevant: , engines: , campaigns: , major: , average: , business: , queries: , ratings: , search:

This abstract has 1,427 tokens.

Sample 10,000 sets of 1,427 tokens from the corpus. For each sample, we sample 1,427 tokens randomly from the total 54,635 tokens in all 224 abstracts without replacement. These represent our null model: that all tokens in the given abstract have no special meaning of their own, but are instead randomly selected from the corpus. We then count the number of times a keyword appears in each sample, and plot the histogram of these word frequencies, to compare against the observed frequency in the abstract.

ACM-TRANS-TWEB/TWEB001sigtest-sponsored.pdf ACM-TRANS-TWEB/TWEB001sigtest-nonsponsored.pdf

ACM-TRANS-TWEB/TWEB001sigtest-links.pdf ACM-TRANS-TWEB/TWEB001sigtest-relevance.pdf

ACM-TRANS-TWEB/TWEB001sigtest-e-commerce.pdf ACM-TRANS-TWEB/TWEB001sigtest-analyzed.pdf

ACM-TRANS-TWEB/TWEB001sigtest-percentsign.pdf ACM-TRANS-TWEB/TWEB001sigtest-relevant.pdf

ACM-TRANS-TWEB/TWEB001sigtest-engines.pdf ACM-TRANS-TWEB/TWEB001sigtest-campaigns.pdf

ACM-TRANS-TWEB/TWEB001sigtest-major.pdf ACM-TRANS-TWEB/TWEB001sigtest-average.pdf

ACM-TRANS-TWEB/TWEB001sigtest-business.pdf ACM-TRANS-TWEB/TWEB001sigtest-queries.pdf

ACM-TRANS-TWEB/TWEB001sigtest-ratings.pdf ACM-TRANS-TWEB/TWEB001sigtest-search.pdf

The p-value is computed using , which is the probability that the null-model word frequency is larger or equal to the observed word frequency .

**sponsored (13):** , **nonsponsored (2):** , **links (8)**: , **relevance (4):** , **e-commerce (2):** , **analyzed (2):** , **% (4):** , **relevant (3):** , **engines (4):** , campaigns (2): , major (2): , average (2): , business (2): , **queries (5):** , ratings (2): , search (9): .

Given an abstract abs, first tokenize it:

aw = nltk.tokenize.word\_tokenize(abs)

Then create 10,000 samples of the same size as aw:

f = []

for n in range(10000):

wn = np.random.choice(words, len(aw), replace=False)

wn = wn.tolist()

f.append(wn.count(keyword))

bb = np.arange(10)

plt.hist(f, bb, density=True)

Demonstrate the problem of wrong corpus = loss of sensitivity,

Using this method, we can compare corpora. We can define quantitative measure of how different corpora are based on log word rank movement.