Vol.2014-DBS-159 No.8
Vol.2014-IFAT-115 No.8
2014/8/1

Finding Co-occurring Topics in Wikipedia Article Segments

Renzhi Wang, Jianmin Wu, Mizuho Iwaihara

Graduate School of Information, Production and Systems
Waseda University, Kitakyushu 808-0135, JAPAN

Abstract: Wikipedia is the largest online encyclopedia, in which articles form knowledgeable and semantic resources. A number of researches about detecting topics and semantic similarity analysis are based on the Wikipedia corpus. Identical topics in different articles indicate that the articles are related to each other about topics. Finding such co-occurring topics is useful to improve the accuracy of querying and clustering, and also to contrast related articles. Existing topic alignment work and topic relevance detection are based on term occurrence. In our research, we discuss incorporating latent topics existing in article segments by utilizing Latent Dirichlet Allocation (LDA), to detect topic relevance. We also study how segment proximities, arising from segment ordering and hyperlinks, shall be incorporated into topic detection and alignment.

Keywords: LDA, MLE, Link, Wikipedia.

1. Introduction

Wikipedia articles are edited by different volunteers with different thoughts and styles. Articles in Wikipedia are structured by links. The concept name where a link is from and the article name of the link destination do not need to completely match. So the topic's relationship cannot be found by links when the same topic is expressed in a different way. The structure of Wikipedia itself cannot show the topic relationship. The method only use link information before is not accurate, finding topic's semantic relationship can help us to improve the accuracy of query and clustering, and to contrast related articles.

TF-IDF is a numerical statistic method that is intended to reflect how important a word is to a document in a static corpus.[7] TF-IDF only uses the statistical effect as the parameters. The semantic information about the word is missing. Also, if there are no common words (except for stop words) in different articles, there will be no relationship by the TF-IDF method, but it is apparently wrong with general recognition. Typically, articles can be modeled as a "bag of words", and each word is assumed to occur independently. A topic can be represented as a group of words. Latent Dirichlet Allocation (LDA) has been the most popular topic model, with more and more variants appearing after the original LDA model was proposed by David M. Blei [1]. LDA model is a generative model. It assumes topics to be multinomial distributions over words and assumes articles to be sampled randomly from those topics. The LDA model uses the Dirichlet priors for the articles over topics and the topic over words. It is usually used to cluster articles by semantic meaning. For Wikipedia structure, two methods above do not utilize the Wikipedia links. Explicit Semantic Analysis (ESA) is a good method to use Wikipedia links to assess the relatedness of articles, it cluster the articles into the Wikipedia-based concept space and evaluate the relatedness of articles. But it cost a lot. The Wikipedia Link-based Measure (WLM) is another method to obtain semantic relatedness from Wikipedia Links. It cost less than ESA, but accuracy is lower about 6% than the ESA.

For those advantages of each model, we carry out an LDA-based algorithm to find co-occurring topics in Wikipedia articles, and capture the topic meaning. The output of LDA is a sparse matrix. To improve the results, our algorithm combines LDA with Maximum Likelihood Estimation (MLE) to smooth the LDA result. Considering the effect of neighboring articles, we also utilize Wikipedia links to reflect network influence. Our experimental results show that when suitable parameters are given, it can achieve a high F1-score.In Section 2 we discuss cluster methods based on TF-IDF and LDA topic model. We present a new algorithm based on LDA and link information in Section 3, and describe our experimental results in Section 4. Finally, Section 5 is the conclusion and future work.

2. Related Work

2.1 LDA

LDA is a generative topic model that each document is viewed as a mixture of various topics and the topic distribution is assumed to have a Dirichlet prior. LDA model can provide us semantic topic by training the corpus. The LDA model is already proved to work well as a topic tracking, classification tool in many fields such as Facebook, newspaper, academic literature [2]. We can expect the LDA model will work well in the Wikipedia corpus, but there are issues that need to be resolved. If the corpus consists of the complete articles in Wikipedia, the result will not be good enough, because the articles are long on average and most of them having not only one topic. However, due to the corpus size, the result of LDA training becomes a sparse matrix, where a sparse matrix means that each article is only mapped to one or few main topics. The minor topic will not be obvious enough to be extracted. It leads to a low accuracy. Another reason is that all the articles in the corpus are seen as independent during training. But as we know, Wikipedia has a network structure of articles, where articles are connected by interlinks. If we just see the articles as independent, the structure information will not be reflected.

2.2 ESA and WLM

Explicit Semantic Analysis (ESA) [9] is a vector-space model, in which not only term weight vectors are compared, but also link weight vectors are compared to evaluate relatedness, such as linked documents like Wikipedia. The vector elements in ESA are Wikipedia-based concepts which are constructed by human, so it is costly. The method compares the text vectors which reflect on the concept space and calculate their similarity. WLM [10] utilizes the vector-space model and normalized Google distance to measure relatedness. Their link vectors are similar to TF-IDF vectors. They use link counts weighted by the probability of each link occurrence, and reasonable results can be easily calculated.

3. Link-weighed corpus

3.1 Wikipedia characteristics

As the world largest encyclopedia, Wikipedia creates a large, complex network, where articles are connected by interlinks. The link distance between two article nodes and other graph-theoretic information can be utilized for topic detection. One of assumption is that the links from a central article to neighboring articles assist complementing the content of the central article by incorporating the neighboring articles. In order to use this structure information, we propose to create a suitable corpus for target articles.

Definition(distance-based sphere)**:**Given a central article *A* and a distance *k*>0 which is measured by the number of links between two article nodes, a *k*-sphere $SP_k(A)$ is the set of article nodes that are connected to *A* by *k* or less links and *A* itself, where link directions are ignored.

We make the union of the terms in $SP_k(A)$ as the corpus. Thus, all the articles in the corpus are directly or indirectly connected to the central article.

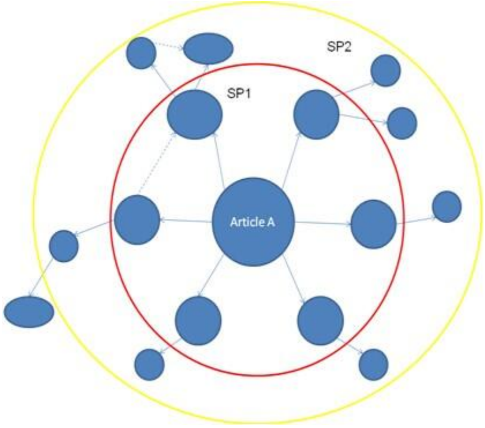

Fig. 1 The distance-based sphere of the article

Wikipedia articles are edited by many different online volunteers. Articles are usually long with multiple topics. The LDA model proved to work well when the articles in the corpus are relatively short like online news articles. But Wikipedia articles often have long, detailed tests, and additional contents can be found from its neighbor articles. So we divide one whole article into several segments based on its logical structure. The best situation is that each segment is short and only contains a few topics. As articles are paragraphed by editors when it was edited, we can just divide the whole article into segments by paragraphs. We see each segment as a document in the corpus. These documents (segments) fit the LDA model better than directly applying onto whole articles.

3.2 Algorithm

Wikipedia is a structured encyclopedia, in which a link connects a term with another article as reference. We assumed that linked articles bring additional information to the source article and affect the topic of the source article. We divide the information of the articles into three parts: the first is the obvious part, which can be observed by simple term occurrence. The second part is the latent part, and it needs to use a latent topic model to extract the latent part. The third part is the link structure information from the sub graph based on *k*-sphere.

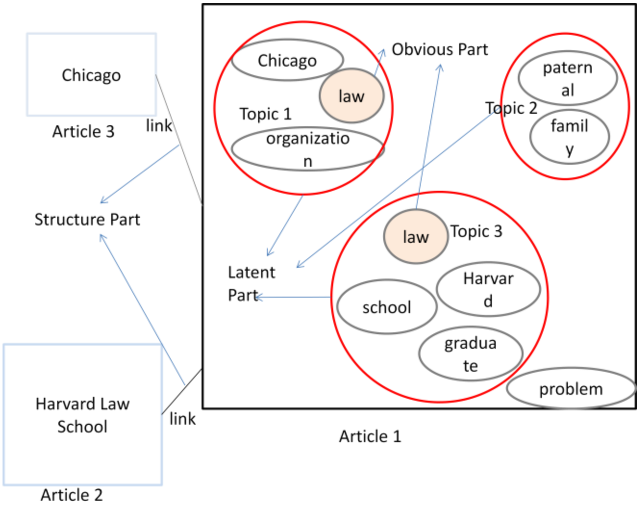

Fig. 2 Article feature consists of three parts

We construct the corpus as the union of SP_k*(Articles)*with given articles. Then divide all the articles into segments by paragraphs and regard each segment as a document in the training data. Then we find what topic each segment as and calculate their segment-wise similarities.

The simplest way to estimate the similarity of segments is to find the feature vector of each segment and calculate their cosine similarity. Since we assume that each segment is generated by a topic distribution, and one segment has a major topic, it is not suitable to use topic probabilities as the feature vector element. So we set term probabilities as the feature vector elements. In the LDA model, a topic is seen as a distribution on all the words, so one of method is to set all the words in the corpus as our vector elements.

The LDA model models a document over topics by topic distribution$\boldsymbol{\theta}$, and atopic over is represented by a distribution over words by word distributionϕ.

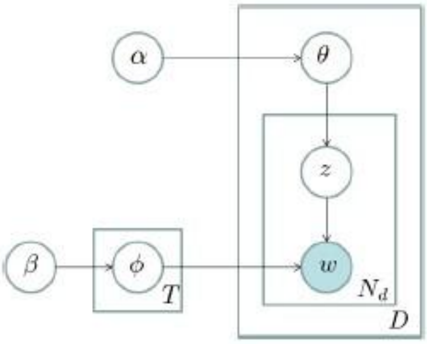

Fig. 3LDA generation process [1]

After training an LDA model, we can obtain the probability of a term in a document by the following formula:

$$\mathrm{P}_{\mathrm{lda}}\ (\mathrm{w} \mid \mathrm{d},\hat{\theta},\hat{\phi}) = \sum_{\mathrm{z}=1}^{\mathrm{K}} \mathrm{P}(\mathrm{w}|\mathrm{z},\hat{\phi})\, \mathrm{P}(\mathrm{z}|\hat{\theta},\mathrm{d}) (1)$$

Where $\hat{\theta}$ and $\hat{\phi}$ are the posterior estimates of θ and φ respectively. Because of the two distributions in the output of LDA are sparse, it is necessary to smooth the LDA result. The probability of terms can be specified by the document language model. We refer to [] for the obvious part of the probability, which is a linear combination of the document-level probability and collection-level probability:

$$P(w|D) = \lambda\left[\frac{N_d}{N_d+\mu}P_{ML}(w|D) + \left(1-\frac{N_d}{N_d+\mu}\right)P_{ML}(w|Coll)\right]$$
$$+(1-\lambda)P_{lda}(w|D)(2)$$

Here $\mathrm{N_d}$ is the number of terms appearing in the segment [3][4][5][6].The first part of formula (2) is the probability of word appeared in the segment by term frequencies. This segment-level probability is combined with the collection probability part by the smoothing parameter μ.We adjust the value of μ to optimize the obvious part. The third part is the potential part consisting of the probability of the word appeared in the segment estimated by LDA. Smoothing parameter $\boldsymbol{\lambda}$ is to adjust the ratio of the obvious part and potential part. The weight proportion of the obvious part and potential part also affect the similarity result. By the formulae (1) and (2) we can obtain the probability of all the words in the corpus being generated by the segment itself.

We describe the structure as follows:

$$\sum_{i=1}^{N_D} w(D,D_i)P_{LDA}(w|D_i)(3)$$

Where D_i is the set of articles of which have links from *D*, $w(D, D_i)$ is the weight of the link from *D* to D_i, $\mathrm{P_{LDA}}(w|\mathrm{D_i})$ is the probability by LDA of word *w* appearing in the segment of the link destination. N_D is the number of links from source segments. So our overall word probability is as follows:

$$P(w|D) =$$
$$\alpha\left\{\lambda\left[\frac{N_d}{N_d+\mu}P_{ML}(w|D) + \left(1-\frac{N_d}{N_d+\mu}\right)P_{ML}(w|Coll)\right] +\right.$$

1−λPldawD+1− αi=1NDw(D,Di)PLDA(w|Di)(4)

In (4), $\mathrm{D_i}$ in $\mathrm{P_{LDA}(w|D_i)}$ should be ranging the whole target article. Here α is the parameter to adjust the ratio of the weight of the segment itself and the structure information from links. Let *s* and *t* be the source article and target article of a link, respectively. We calculate the link weight (normalized LF-ICF) in a TF-IDF fashion as below. Link frequency represents how important a link in one segment, inverse corpus frequency represents how important a link in the corpus. The function *w*(*s*, *t*) is the weight of the link from *s* to *t*, calculated as the normalized *link frequency* (LF) multiplied by the *inverse corpus frequency*(ICF) defined as below:

$$\mathrm{LF(s,t)} = \frac{\mathrm{N(s,t)}}{\mathrm{N(s,*)}} \qquad (5)$$

$$\mathrm{ICF(t)} = \log\frac{\mathrm{N(all)}}{\mathrm{N(*,t)}} \qquad (6)$$

Where N(*s*, *t*) is the number of links from *s* to *t*. N(*s*,*) is the number of links from *s* to any target. N(*,*t*) is the number of links from any source to t. N(all) in the total number of all links in the corpus. The normalizing function is $\sum_{\mathrm{i}=1}^{\mathrm{N_S}}\mathrm{LF}\cdot\mathrm{ICF(s,i)}$, and we define w(*s* ,*t*)=$\frac{\mathrm{LFICF\ (s,t)}}{\sum_{\mathrm{i}=1}^{\mathrm{N_S}}\mathrm{LFICF\ (s,i)}}$, where$\sum_{\mathrm{i}=1}^{\mathrm{N_S}}\mathrm{w}(s,i)$.=1 . Here $\mathrm{N_S}$ is the number of links from source segment *s*.

But in the first two steps, we have already divided all articles into several segments, and Gibbs LDA’s result is different in every sample. It is Wikipedia’s edit principle that the first paragraph should be the summary of the article. So we can just use the first segment of the linked article to replace the complete article.

One way to assign the values of vector elements is to use all term probabilities of the segment. For representing the topics of each segment, top-*N* probability words can be used.

Mutual information is often used to measure variables’ mutual dependence. The mutual information between words and segments can be utilized as vector elements [8].In this case, a vector represents a distribution of mutual information between words and segments. Mutual information between a word and segment is defined as

$$MI(word,seg) = P(word,seg)\,log\left(\frac{P(word\ ,seg)}{P(word\ )P(seg)}\right), (7)$$

where *P(word, seg)* is from the formula (4).

For the meaning of the segment, the words with the top-N highest mutual information about the segment are expected to explain the segment. It is assumed that if there are co-occurring topics in two segments, there is probably a high cosine similarity between the vectors of two segments. So it is reasonable to assume that segments have co-occurring topics if their similarity is more than a threshold. In our assumptions below, there are two types of co-occurrence.

The first type of co-occurrence is that two segments describe one common thing or one common event. The event or thing is a set of words which contain one key word or one key phrase. All the other words are supplementing the keyword or key phrase. This set of words is a subset of one topic. For example: “He began his presidential campaign in 2007 and, after a close primary campaign against Hillary Rodham Clinton in 2008, he won sufficient delegates in the Democratic Party primaries to receive the presidential nomination.” from article *Barack Obama*. And “Running in the 2008 Democratic presidential primaries, Hillary Clinton won far more primaries and delegates than any other female candidate in American history, but narrowly lost the nomination to U.S. Senator Barack Obama, who went on to win the national election.” from article *Hillary Rodham Clinton*. The two segments pair belongs to the *Type-1*.

The second type of co-occurrence is that two segments belong to one category. Here we consider that one category contains more than one topic, and a category is the summary of several similar topics. In our experiment, we examine the categories in the bottom part of Wikipedia articles to acquire new categories. Example: the 4th segment of article *Barack Obama* and the 4th segment of article *Hillary Rodham Clinton* describe the two person’s work after graduating from law school. Even though the word “lawyer” is not in segments pair both segments belong to category lawyer. The two segments pair belongs to the *Type-2*.

If neither the first type, nor the second type, then the two segments are not related. In this case, their major topics are dissimilar, and they belong to no common category.

We utilize the following category hierarchy to test the above Type-1 and Type-2 co-occurrences. The ODP website [11] is the largest, most comprehensive human-edited directory of the Web. It is constructed and maintained by a passionate, global community of volunteer editors. Our standards of judgment which condition two segments belong to are as follows:

There are 16 roots in the category tree. If the category two segments describe is the leaf node in ODP, then we judge these two segments are Type-1. If the common category of two segments, if any, describe is the node having a height less than or equal to 3, we judge these two segments are Type-2. Otherwise, the category two segments describe is the node whose height is more than 3 or no common ancestor, and we judge these two segments have no co-occurring topics.

The potential part is closely related to the topics, while the obvious part just depends on term frequencies. So our prediction is that in the three types of co-occurrence, λ would make different influences.

4. Experiment and Evaluation

4.1 Corpus data

Our experimental corpus is from the latest revision of pairs of articles in Wikipedia. Each pair of articles (A_1, A_2) is used as central articles and expands the whole corpus by incorporating links within Wikipedia. Then divide all articles in set SP_k (A_1)∪ SP_k (A_2) into segments by paragraphs. Section titles or subsection titles are merged with their following paragraphs. Those segment sets will be used as our LDA training data. Here k=1.

The following article pairs are used in our experiment.

Article title	Word count	Segment count	Size(KB)
C Sharp (programming language)	5736	21	36.5
Java (programming language)	6771	31	43.7
Google	8081	21	50.3
Yahoo!	5075	31	31.2
Facebook	10959	46	67.3
Twitter	9835	41	60.5
Tencent QQ	2294	23	13.9
Windows Live Messenger	5278	20	31.8
Buddhism	16717	59	109.0
Christianity	12190	37	81.5
Apple Inc	12328	31	75.0
Samsung	6937	82	44.0
DirectX	4438	11	21.2
OpenGL	5794	26	37.8
League of Legends	4185	4	24.3
Defense of the Ancients	2196	14	13.4
Linux	6717	20	44.7
Microsoft Windows	5752	19	35.8
Barack Obama	11970	36	74.0
Hillary Rodham Clinton	14389	32	90.4
Shaquille O'Neal	12166	25	69.5
Kobe Bryant	13150	35	75.2
Winfield Scott	5293	25	32.1
Robert E. Lee	11921	23	71.3
Avril Lavigne	8473	24	49.6
Yui (singer)	3357	17	18.9
Lionel Messi	14054	46	83.1
Cristiano Ronaldo	16171	43	96.2
Max	16717	82	109
Min	2196	4	13.4
Average	8650	30.1	53.2

Table 1 Experiment data

4.2 Parameters setting and result

In our experiment, we need to determine the topic number K and smooth parameters λ and μ. In Wikipedia, articles are usually less than 100 paragraphs, and one long article usually has less than 300 links, so we set topic number K as 100, and$\lambda$is from 0.0 to 1.0. In Wikipedia articles, one paragraph usually has less than 300 words, so we set μ as 1000.We set α as 0.9.

We mark the co-occurring topics from human, and compare them with the co-occurring topics extracted by our method. One of the pairs of articles is “Barack Obama” and “Hillary Rahdom Clindon”. We track the three different types of pairs of segments which are separately marked as Type-1, Type-2, and no relation. We observe how the similarity changes with λ in each type of pair. The trend graphs are shown in Figures 4 to 6.

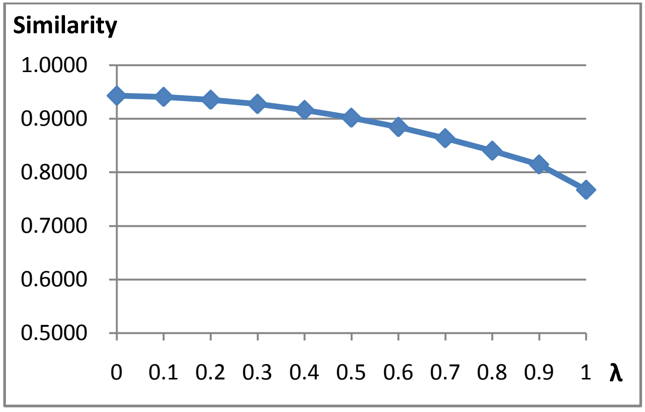

Fig. 4 Similarity between segments pair belongs toType-1.

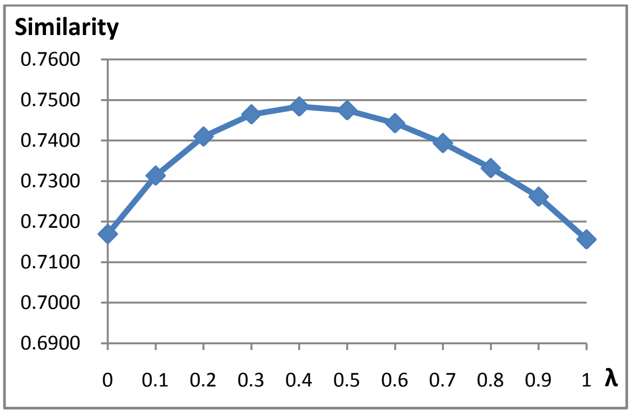

Fig. 5 Similarity between segments pair belongs toType-2.

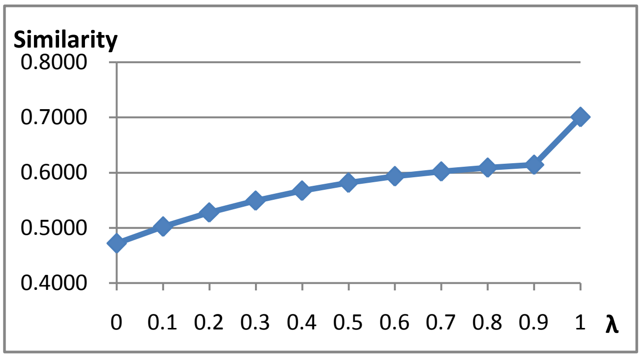

Fig. 6 Similarity between segments pair belongs to No-relation

According to the trend graphs, we can observe that in the case of two segments having co-occurring topics of Type-1, when λ becomes larger, the similarity between the two segments constantly decreases. The similarity is highest when λ is0. When λ is less than 0.5, the similarity between two segments is more than 0.9, and even at the lowest point, the similarity between segments is still around0.8.

If the two segments have co-occurring topics of Type-2, the similarity between two segments is not monotonically decreasing. There is a peak point. In this example, when λ is 0.6, for two segments which are marked as Type-2, the similarity between the two segments is highest. The peak value is not a

As Figure 5 shows, for segment pairs that are not related, the similarities of the two segments are low, usually less than 0.7. The similarity increases slowly with λ.

Figures 7-9 are the results when mutual information is used as the feature vectors. We can see the trends change little, but the magnitudes are different.

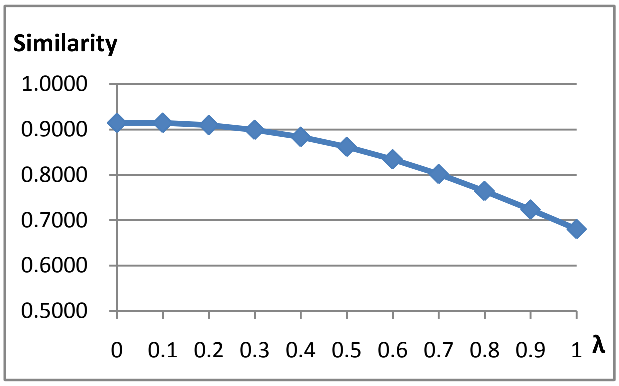

Fig. 7 Similarity between segments belongs to Type-1 (MI)

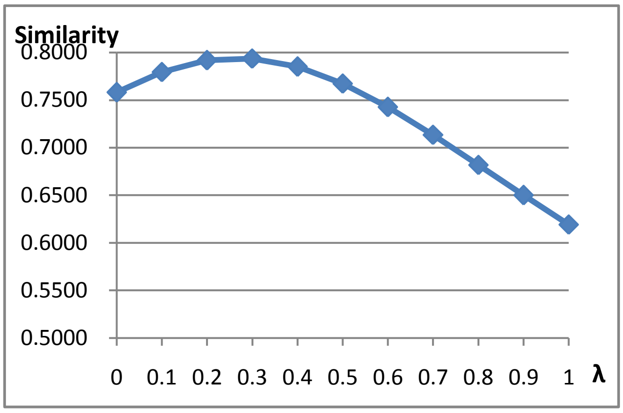

Fig. 8 Similarity between segments belongs toType-2 (MI)

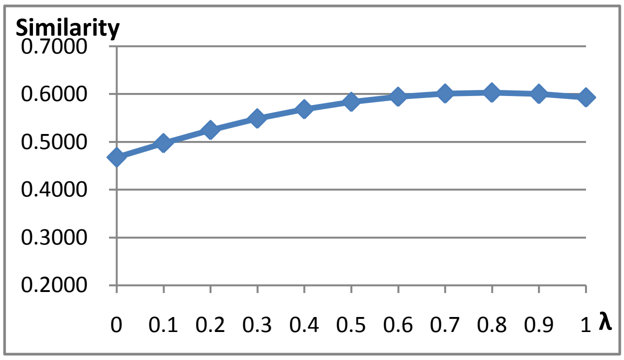

Fig. 9 Similarity between segments belongs to No-relation (MI)

Using mutual information as the feature vector more clearly distinguishes trends.

4.3 Analysis

The LDA model is a clustering model which documents are clustered into several topics. Meanwhile, the words are also projected onto these topics. As for semantically similar segment pairs, they should be clustered into the same topic. Therefore, in this case, the potential part should occupy a greater portion of the similarity. So for segment pairs that have co-occurring topics, the trend of similarity should decrease with increasing λ.

As for semantically dissimilar segment pairs, they should be clustered into different topics. Therefore, the potential part should be the major cause of dissimilarity. At the same time, the obvious part would play an important role in supporting similarity. So the trend of similarity should be increasing with increasing λ.

For Type-2, our assumption is that two segments belong to an identical category, but they are not about the same event. When a relatively large topic number for LDA is given, each resultant topic becomes more detailed. So in the LDA-based clustering, two segments are likely to be clustered into different topics. So the larger potential part tends to contribute to dissimilarity. Since they belong to different topics, there may be little identical terms in both segments, so only the obvious part will give dissimilarity. Even though they are in different topics, the similarity between topics is close because they are under the same category. In our observation on our experiment data, when we decrease the weight on the obvious part, the tendency of dissimilarity is weakened.

In addition to the trend of similarity, another requirement for judging pair belongs to which type is the value of similarity between two segments. The value should be more than a threshold if there are co-occurring topics in two segments. In our experiment, the value of the threshold on which pairs are judged as Type-1 is set to 0.9. For segments which are not related, the similarity value should be less than the threshold. In our experiment, the value of the threshold for which pairs are judged as No-relation is set to 0.6. For segments which belong to an identical category the similarity is neither very high nor very low, it is in the peak region. So it is necessary to give a threshold range to judge whether the segment pair belongs to Type-2.In our experiment we set the threshold range as [0.7,0.75]. We should compare the highest similarity of each segment pair with the threshold and predict which types the pair belongs to. In the case of mutual information used in the feature vector, the three similarity ranges for classifying segment pairsintoType-1, Type-2, No-relation are [0.9, 1], [0.75, 0.8], [0, 0.6], respectively. The range is the necessary condition of predicting which types the pair belongs to.

Finally, we measure the precision, recall and F1-score for each type to evaluate our method based on the data in Section 4.1. The reference relationship of the two segments is judged by human. The parameters are described above. We set μ as 1000, α as 0.9. λ is from 0.0 to 1.0.

$$\text{precision} = \frac{\text{number of pairs judged as related by both method and human}}{\text{number of pairs judged by the method}}$$

$$\text{recall} = \frac{\text{number of pairs judged as related by both method and human}}{\text{number of pairs judged by human}}$$

$$\text{F1} - \text{score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

	Type-1	Type-2
Precision	50%	35%
Recall	80%	60%
F1-score	0.615	0.442

Table 2 Precision and Recall

5. Conclusion and future work

In this paper, we propose an LDA-based algorithm to find co-occurring topics in different segments of Wikipedia's articles and evaluate the method. We make some improvements on collecting corpus by using neighboring articles with interlink of articles within Wikipedia. We are using the LDA model, combined with MLE and link information. Comparing the result of λ is 1 (no potential part) and other value (have potential part), we can see LDA is more suitable to extract topic than just term frequency clearly in segments, similarity is high. As Figure 5 shows, combine LDA with MLE improve the similarity of segment which are marked as Type-2. In our experiment, we give a weight to calculate how much impact a link on segment vector. The smoothing parameters can be different for different field. And also the threshold can be chosen differently. The mutual information between word and segment will give better distinction for classifying segment pairs. The trend of similarity between two segments with changing λ indicates how to determine this smoothing parameter.

In future work, for explaining the co-occurring topic meaning, top-N words are not appropriate, since they are hard to comprehend. Our method of finding co-occurring topics between Wikipedia articles should help illuminate overlapping topics between long articles, so that users can discover multiple articles dealing with the same topic, and compare viewpoints of these articles.

Reference

1) Blei, D. M., Ng, A. Y., and Jordan, M. J. Latent Dirichlet allocation. In Journal of Machine Learning Research, 3, 2003, pp. 993-1022.

2) D. M. Blei and P. J. Moreno. Topic segmentation with an aspect hidden markov model. In Proceedings of SIGIR,2001.

3) Lavrenko, V. and Croft, W.B. (2001). Relevance-based language models. In SIGIR 2001, pp.120-127.

4) Liu, X. and Croft, W. B. Cluster-based retrieval using language models. In Proc. 27th International ACM SIGIRConf. Research and Development Information Retrieval 186-193, 2004.

5) Xing Wei and W. Bruce Croft. LDA-Based Document Models for Ad-hoc Retrieval. Proc. 29thACM SIGIR Conf., pp. 178 – 185, 2006.

6) Zhai, C. and Lafferty, J. A study of smoothing methods for language models applied to ad hoc information retrieval. In Proc. 24th ACM SIGIR2001, pp. 334-34

7) Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. In J.Info.Processing and Management: an International Journal, Vol. 24 Issue 5, pp. 513-523,1988.

8) B. Sun, P. Mitra, H. Zha, C. L. Giles, and J. Yen. Topic segmentation with shared topic detection and alignment of multiple documents. ACM SIGIR, pp.199–206, 2007.

9) Evgeniy Gabrilovich, Shaul Markovitch.Computing semantic relatedness using Wikipedia-based explicit semantic analysis. Proc.IJCAI'07 Proceedings of the 20th international joint conference on Artifical intelligence, pp. 1606-1611, San Francisco, 2007.

10) David Milne, Ian H. Witten. An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links.Proc. AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, Chicago, pp. 25-30, 2008.

11) http://www.dmoz.org