Ranking Image Annotation Using Vector Space Model

Shan-Bin Chan † Duy-Dinh Le ‡ Shin'ichi Satoh ‡ Hayato Yamana † ‡

† School of Fundamental Science and Engineering, Waseda University 3–4–1 Okubo, Shinjuku-ku, Tokyo, 169–8555, Japan

‡ National Institute of Informatics 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan

E-mail: † {chrisjan, yamana}@yama.info.waseda.ac.jp, ‡ {ledduy, satoh}@nii.ac.jp

Abstract Many researches perform Support Vector Machine (SVM) to annotate images. Researchers collecte sample images and train concept classifiers. After classifier training, they annotate an image by employing many classifiers. Different classifiers require different sample image sets. Therefore, the image annotation predicted from different classifiers form the independent probability. Nether less, literature shows this kind of annotation results is not capable of being ranked. This research will use the Vector Space Model (VSM) to rank the images, which resemble which other and managed to annotate a relevant concept for a query image. This research uses the WordNet API to build a hierarchical structure and uses this hierarchical structure to generate negative dataset for SVM training. 100 concepts are chosen for experiment. We use the SVM predicted results to build a matrix for VSM calculation. At the final stage, the VSM calculation results are carefully employed to annotate a best concept for a test image. This research uses the fusing SVM probabilitic results as the baseline. In the VSM calculation, one experimental test performs the term weight calculation by using TF-IDF, and the other one performs SVM probabilistic result predicted from each classifier as the term weight. The accuracy of baseline is 23%, the best accuracy of VSM calculation with TF-IDF experimental test is 42%, and the accuracy of VSM calculation with SVM probabilistic result is 47%.

Keyword Image Recognition, SVM, Machine Learning, Vector Space Model, WordNet, ImageNet

1. Introduction

There are many methods of collecting concepts from an image. The main methods of concepts collection are Text-based Image Retrieval (TBIR) and Content-based Image Retrieval (CBIR). TBIR method collects the articles around the image, extracts the main keywords from the article, and then indexes with these keywords for image search engines.[7] CBIR method uses the low-level features of an image, trains a concepts database by using pre-collected sample images and pre-defined concepts, and then annotates an image by trained concepts database.

Many researches implement Support Vector Machine (SVM) method to training concepts classifier.[19] The SVM is fundamentally a two-class classifier. In other words, for each concept requires different classifier for image annotation. And different classifiers require different sample image sets. Therefore, the image annotation predicted from different classifiers form the independent probability. This kind of annotation results is not capable of being ranked. Although, some researches fuse multi low-level features and annotate an image by using SVM. But the accuracy of the annotation results is around 20-30%.[5] Our research will fuse the CBIR and TBIR method for annotating an image. We extract low-level features from sample images and train classifiers by using SVM. Then we reuse the pre-collected sample images and match with trained classifiers. We collect the SVM probabilistic results for each sample images and convert these results to high-level features. Then we regard these high-level features as keywords and combine these keywords to a document for each sample image.

Vector Space Model (VSM) is an algebraic model that can rank the relevance between a query and documents. We regard test image as a query and all the pre-collected sample images as documents. We employ VSM to calculate the relevance between test image and sample images, and try to find the images that are similar with the test image from sample images. Because the pre-collected sample images from ImageNet[10] are gathered by specialists, each sample image has the precise concept. We choose the relevant sample images and use the concepts on these relevant sample images to find the best annotation for a test image. This kind of method fuse with SVM and VSM is not proposed by other researchers. The result of our fused method also improves the accuracy of annotation results to 32-47%.

2. Methods

Our approach will focus on the image annotation process. The flowchart of image annotation is shown as Fig 1. Our research flow is as the following steps:

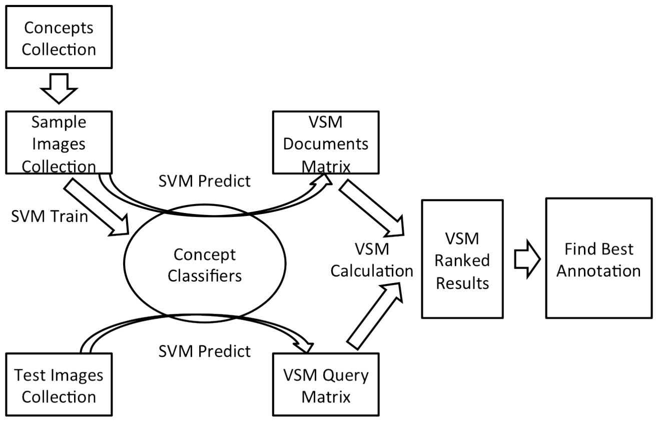


Fig 1. Research Procedure

- Choose 100 concepts randomly from image-tag pool where it is constructed by gathering image-tags, i.e., concepts, from recently uploaded photos on Flickr. We call 100 concepts as "loosely defined concepts."
- Gather 2,000 sample-images for each concept at maximum by employing ImageNet.
- For each concept, prepare one SVM classifier [5][6] and train it by using image features of the 2,000 sample-images; color moments, edge of histogram, and local binary pattern of the gathered sample-images. Before training, we use a concept hierarchal tree built by using WordNet[21], and generate positive and negative datasets for each concept.
- For each SVM, in total for 100 SVMs, perform SVM prediction with test-images. Here, one test-images for each concept, in total 100 test-images, are prepared.
- Perform SVM prediction for each sample-image and test-image.
- Convert SVM predicted probabilistic results to VSM term-weight matrix. Then, regard sample-images as "documents" and test-image as "query".
- Calculate VSM cosine similarity between "query" and "documents".
- Query with test image, and rank entire positive sample-images by VSM cosine similarity.
- Discover the best annotation for test-image from top 100 results.

The following subsections will introduce our approach in detail.

2.1 Concepts Collection

We want to know what kinds of concepts are popular or what kinds of keyword are often used for tagging an image. So, the first step, we collect concepts from the photos on Flickr by using Flickr API. We do not crawl all the images on the Flickr, because the amounts of images are too huge. We employ the "flickr.photos.getRecent" API to retrieve the recently uploaded photos randomly. We only retrieve the tags of recently uploaded photos and we use the POS (part-of-speech) application[23] to tag the retrieved tags. And we store the image tags, counts, and POS tags to the database. We have already collected 176,303 concepts from July 2009 to July 2010. We will only use the noun as the concepts for sample images retrieval. Finally, the amounts of noun concepts are 21,309.

2.2 Sample-Images Collection

We use these noun concepts collected from Flickr as the queries and search the WordNet ID by using WordNet API. And then we use the WordNet ID to gather the sample image list from ImageNet. Then we retrieve the sample images from the sample image list. For each concept, we retrieve 2,000 images at maximum as sample-images. And we delete the sample image whose image size is under 10Kbytes. We also skip the noun concepts if the number of their sample images is less than 50. Finally, the total number of image-tags is 4,751. In this research we will only pick 100 concepts from image-tag pool for baseline and experimental tests.

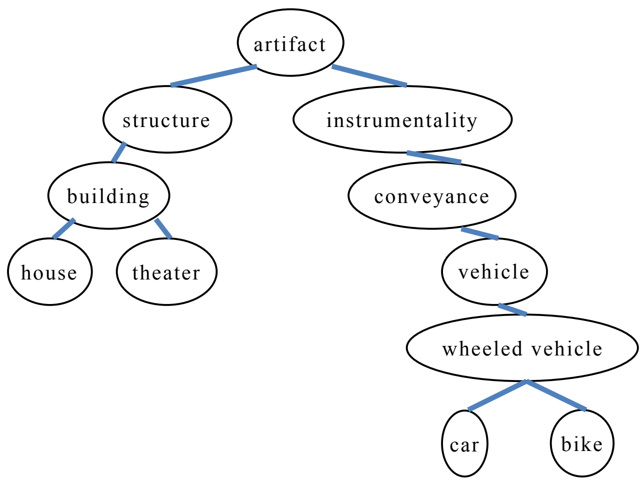


Fig 2. Apart of concept hierarchical structure

Before training a SVM classifier, we must prepare negative sample images. We build a hierarchical structure of collected concepts by using WordNet ID API. Fig 2 is a part of concept hierarchical structure. We randomly pick 20 concepts for maximum from a sub-tree that is the nearest sub-tree of positive concept. For example, if the positive concept is "car", we pick the nearest sub-tree "bike" and randomly choose 20 concepts for maximum as negative concepts. We also randomly pick 20 concepts for maximum from a sub-tree that is far away from positive concept. For example, if the positive concept is “car”, we pick the far away sub-tree “building” and randomly choose 20 concepts for maximum as negative concepts. The amounts of negative sample images will equal to the amounts of positive sample images. In other words, we choose 40 negative concepts for maximum and totally 2,000 negative sample images for maximum.

2.3 SVM Training and Predict

We extract low-level features from the entire sample images, and perform the SVM classifier training for each concept. After the training of concept classifiers, we perform the image annotation for each sample images by using SVM. We employ the annotation results from each sample-images to build a VSM matrix. Each matrix from each sample image is known as VSM “document”. The procedure of SVM training and predict stage is shown as Fig 3.

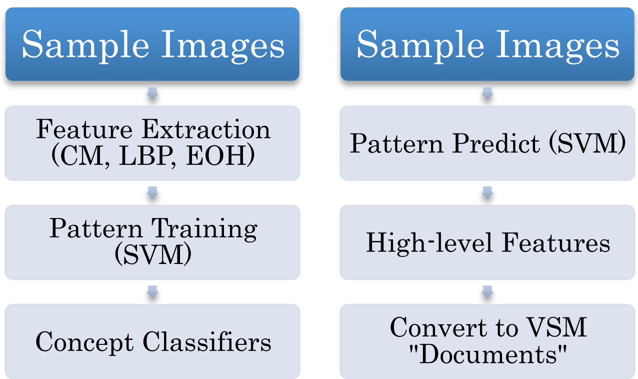


Fig 3. SVM Training and Predict Procedure

Our approach preform three feature extraction methods, Color Moments (CM), Local Binary Pattern (LBP), and Edge of Histogram (EOH) on each sample images. On CM feature extraction method, we use 6x6 grids and YCbCr as the parameters and the total extracted dimensions are 324. On LBP feature extraction method, we use 4x4 grids and 59 quantization as the parameters and the total extracted dimensions are 944. On EOH feature extraction method, we use 6x6 grids and 72 quantization as the parameters and the total extracted dimensions are 1,168.[5][6] After finishing the feature extraction of sample images, we implement SVM training for each pre-collected concept. We use C-SVC as the SVM type and use Chi-Square as the kernel type for SVM training. Because we choose 100 concepts for experiment and for each concept we use 3 feature extraction methods. Finally, we have trained 300 classifiers.

After training the SVM classifiers, we collect 100 test-images from Google image search. For each test-image whose main concepts are the same as 100 pre-chosen concepts. For each test-image we perform SVM predict with the pre-trained 300 classifiers. In the same time, we also perform SVM predict with the pre-trained 300 classifiers for the entire sample-images in 100 pre-chosen concepts. So, for each test-image and sample image, we have 300 probabilistic results from 300 classifiers. Then we convert these 300 results to high-level features. We regard these high-level features as “keywords” in VSM and combine these keywords to a VSM “document” for each test image and sample image. Then we perform VSM to calculate the relevance between test-images with entire sample-images.

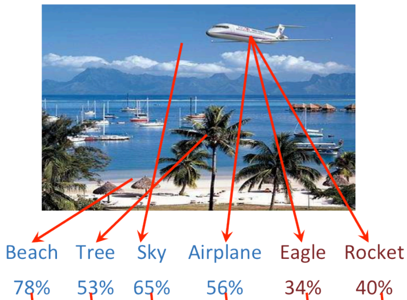
Our approach will try to convert the low-level features (CM, LBP, EOH) to high-level features (airplane, sky, eagle, bird, etc…) by using pre-trained concept classifiers. For each image (including test-images and entire sample-images), we have 300 probabilistic results from 300 SVM classifiers. We will convert these 300 probabilistic results to a 1x300 matrix for each image. We consider the annotation results from each classifier (concept with extraction method) as VSM “keywords” (airplane, sky, eagle, bird, etc…) in each image. We also regard each sample-image as VSM “document” and each test-image as VSM “query”. We will implement two experimental tests for comparing with baseline.

2.4 Image Annotation

In this paper, the baseline approach is a fusion result of SVM probabilistic results. We just simply sum the SVM predict value of three different feature extraction methods. For example, a test image matching with the airplane CM, LBP, and EOH these 3 classifiers, and the SVM probabilistic results is 0.5(CM), 0.8(LBP), 0.4(EOH). The fusion score of SVM predict probabilistic results (0.17) will equal to the summation of 0.5(CM), 0.8(LBP), and 0.4(EOH). And we also calculate the other 99 pre-selected concepts with the same method. Then we use these score for descending sorting and choose the top N (N will be 1 to 3) as the best annotation for the test image.

The two different experimental tests will perform VSM to calculate the relevance between test images with all the sample images. The first experimental test will perform TF-IDF approach as the term weight in VSM calculation. If the probabilistic results of a concept classifier is larger than certain probability P (P will be 0.5, 0.6, 0.7, 0.8, 0.9

in this experimental test), we regard the term frequency is 1 from this concept classifier. If the probabilistic results of a classifier is lower than P, we regard the term frequency is 0 from this concept classifier. For example, in the SVM prediction results of an airplane test-image, the probabilistic result of classifier "Beach" is 0.78, and P is set to be 0.5. Then we count 1 as the term frequency in the term "Beach". If the probabilistic result of classifier "Eagle" is 0.34, then we count 0 as the term frequency in the term "Eagle". Fig 4 shows the process of this kind approach.



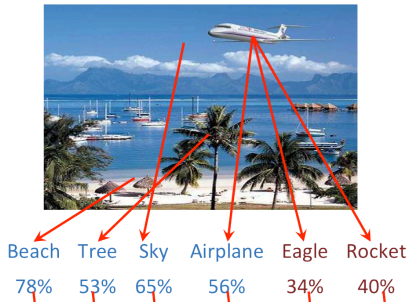
	Beach	Tree	Sky	Airplane	Eagle	Rocket	...
Test Image	1	1	1	1	0	0	...
Sample, Airplane	1	0	0	1	0	0	...
Sample, Eagle	0	1	1	0	1	0	...
All Sample Images	...	...	...	...	...	...	

Fig 4. VSM Term-Frequency Matrix (TF-IDF)

	Beach	Tree	Sky	Airplane	Eagle	Rocket	...
Test Image	0.47	0.31	0.25	0.53	0	0	...
Sample, Airplane	0.47	0	0	0.53	0	0	...
Sample, Eagle	0	0.31	0.25	0	0.77	0	...
All Sample Images	...	...	...	...	...	...	

Fig 5. VSM Term-Weight Matrix (TF-IDF)

We perform this approach to build a 300 row matrix for each sample-image (document). We also build this kind of matrix for each test-image (query). Then we calculate the TF-IDF value by using these term-frequency matrixes. And we use the TF-IDF value as term-weight in VSM, shown as Fig 5.



	Beach	Tree	Sky	Airplane	Eagle	Rocket	...
Test Image	0.78	0.53	0.65	0.56	0.34	0.40	...
Sample, Airplane	0.66	0.12	0.36	0.73	0.45	0.23	...
Sample, Eagle	0.06	0.51	0.55	0.46	0.87	0.18	...
All Sample Images	...	...	...	...	...	...	

Fig 6. VSM Term-Weight Matrix (NON TF-IDF)

The second experimental test will directly perform SVM probabilistic results from a concept classifier as the term-weight in VSM calculation. For example, The probabilistic result of classifier "Beach" is 0.78, then we directly count 0.78 as the term-weight in the term "Beach". If the probabilistic result of classifier "Eagle" is 0.34, then we directly count 0.34 as the term-weight in the term "Beach". Fig 6 shows the term-weight matrix of test-image (query) and sample-images (documents).

After calculating of the term-weight of test-image (query) and entire sample-images (documents), we implement the calculation of VSM (cosine θ) between "query"(Q) and each "document"(D) by using this equation.

$$\cos\theta = \frac{Q \cdot D}{|Q| \times |D|}$$

After calculating the cosine θ of entire sample-images, we can rank the results by using these cosine θ values. After ranking the results, we choose top 100 VSM results as the candidates for discovering the best annotation of a test image.

In this research, we consider that the pre-collected sample images from ImageNet are gathered by specialists, and each sample image has the precise concept. So, we perform VSM approach to search the images that are similar to the test image from pre-collected sample images. Because the search results are gathered by specialists and each result has the precise concept, we will try to use these search results and concepts to annotate a test image. We use this equation for discovering the best annotation.

$$\sum_{concept} cosine\theta \times \frac{1}{rank}$$

We consider that the best rank of VSM results will be very similar to the test-image. So we give the best rank of search result with larger weight. Then we multiply (1/rank) and (cosine θ) for each result. And we also consider that the best occurrence frequency of similar images with the same concepts will be most relevant to the test-image. Base on this propose, we sum the values of cosine θ multiple 1/rank for each concept in the results. Then we pick the best value for the best annotation of the test image. For example, the top 5 results of airplane test-image are shown as Fig 7. There are two concepts (airplane and eagle) in the similar search results of test-image airplane. The annotation score of concept airplane equals to (0.94372/1+0.92358/2+0.92058/3+0.92004/5), and The annotation score of concept eagle is 0.92028/5. Afterward, the best annotation score is airplane. Then we will annotate the test image as "airplane".

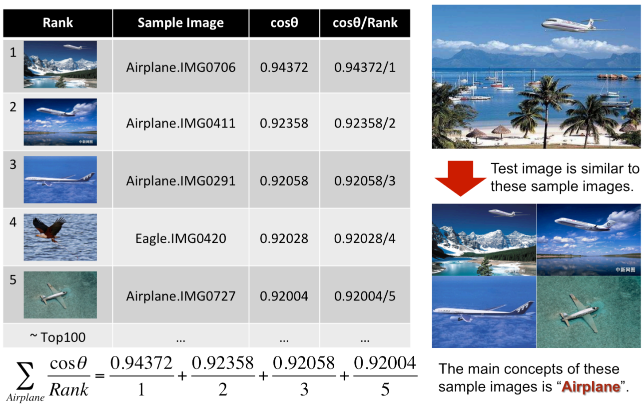


Fig 7. Annotation score calculation procedure

3. Experiments

The collection of concepts of this research is 4,751. When implementing the VSM and SVM calculation, the calculation is really very heavy. It indeed requires long time and huge resource. So we just pick 100 concepts for experiments. The concepts we picked in this research are as Table 1. We also collect a test-image for each pre-selected concepts by using Google image search manually.

Table 1. 100 pre-selected concepts

airplane	ant	ape	apple	ball	beach	bike	bird
butterfly	car	cat	cow	crane	dog	dolphin	door
ducks	eagle	elephant	enve-lopes	eques-trian	fabric	fish	foot-ball
fruit	garden	glow-worm	golf	gorilla	hall	heron	hook
horse	ice cream	ice tea	iceberg	insect	jacket	jasmine	jeans
jellyfish	key	keyboard	kitchen	knife	lamps	leopard	lion
lobby	mango	maple	motor-cycle	mush-room	nails	nar-cissus	narthex
news-paper	onion	orange	orchid	owls	palace	pencil	penguin
people	pine-apple	quads	quahog	quesa-dillas	rabbit	rack	res-taurant
rocket	sea	shark	sheep	sky	swallow	temple	tiger
tree	turtle	umbrella	unicycle	utensil	vessel	video-tapes	violin
volleyball	whale	window	wolf	worms	yam	yard	yellow jacket
zebra	ziggurat	zinnia	zither				

Table 2. Negative concepts list of positive concepts "airplane"

Near Concepts	glider, helicopter, warplane
Far-away Concepts	vessel, boat, tugboat, small, skiff, canoe, kayak, pirogue, birch_bark, rowboats, ferryboat, car_ferry, motorboat, speedboat, hydroplane, cruiser, barge, house_boat, dredger, sea, lifeboat, steamboat, pilot_boat, narrow_boat, fire_boat, ship, warship, submersible, submarine, attack_submarine, battleship, dreadnaughts, destroyer, frigate, aircraft_carrier, surface_ship, steamship, paddle_steamer, cargo, tanker, freighter, containership, liberty_ship, derelicts, pirate, passenger, liner, cruise_liner, hulk, tender, hospital_ship, shipwreck, fishing_boats, trawler, sailing, sailboat, catamaran, sloop, schooner, windjammer, barque, brig, felucca, galleon, dhows, yacht, bareboat, iceboat, shrimpers, spacecraft, spaceship, space_shuttle, lander, lem, hovercraft
Chosen Negative Concepts	glider, helicopter, warplane, small, skiff, rowboats, ferryboat, motorboat, cruiser, dredger, pilot_boat, submarine, frigate, steamship, cargo, containership, hospital_ship, trawler, sailboat, windjammer, bareboat, shrimpers, lem

This research employs the hierarchical concept structure from WordNet to gather the negative sample images. We take positive concept "airplane" for example. Table 2 shows the 3 concepts which are near to positive concept "airplane" and the 75 concepts which are far-away from positive concepts "airplane". We choose the 3 near concepts and randomly choose 20 far-away concepts from 75 far-away concepts. Totally we choose 23 concepts as negative concepts for positive concept “airplane”. Then we implement the SVM classifier training with these negative and positive sample images

In this research, we use the fusion SVM predict probabilistic results as the baseline. Fusion SVM predict probabilistic method is that sum the SVM predict value of three different feature extraction methods. For example, a test image matching with the airplane CM, LBP, and EOH these 3 classifiers, and the results is 0.5(CM), 0.8(LBP), 0.4(EOH). The fusion score of SVM predict probabilistic results (0.17) will equal to the summation of 0.5(CM), 0.8(LBP), and 0.4(EOH). We calculate the SVM predict probabilistic score for 100 pre-selected concepts separately, and sort by descending. If the top N annotated concepts contain the main concept of test image, we regard it is correct and calculate the accuracy of the fusing SVM predict probabilistic method. We calculate the accuracy for different N (1, 2, and 3). Fig 8 is the top 20 annotation results of test image “butterfly”.

Fig 8. Top 20 annotations for test image “butterfly

In our research, we select two different experimental tests. One is using TF-IDF as term weight to calculate the VSM cosine θ. And we use different SVM predict probability (0.5, 0.6, 0.7, 0.8, 0.9) to build VSM matrix for cosine θ calculation. The other one test is using SVM predict probabilistic results directly as term weight for the VSM cosine θ calculation. The accuracy estimation is the same as the fusing SVM predict probabilistic method. If the top N of the annotated concepts contains the main concept of test image, we regard it as correct.

The comparisons of the baseline and experimental tests are shown as Fig 9. In the case of the top 1 annotation concept contains the main concept of test image. The accuracy of fusing SVM probabilistic approach is 23%. The VSM approach with using SVM probabilistic results directly as term weight performs the best accuracy 47%. And the accuracy of using TF-IDF as term weight when SVM predict probabilistic score is larger than 0.7 is 42%. In the case of the top 2 annotation concepts contains the main concept of test image. The accuracy of fusing SVM probabilistic approach is 32%. The VSM approach with using SVM probabilistic results directly as term weight performs the best accuracy 56%. And the accuracy of using TF-IDF as term weight when SVM predict probabilistic score is larger than 0.5 is 55%. In the case of the top 3 annotation concepts contains the main concept of test image. The accuracy of fusing SVM probabilistic approach is 42%. The VSM approach with using SVM probabilistic results directly as term weight performs the accuracy 63%. And the accuracy of using TF-IDF as term weight when SVM predict probabilistic score is larger than 0.5 is 65%.

Overall, The VSM approach with using SVM probabilistic results directly as term weight performs the better accuracy. In the case of the top 1 annotation concept contains the main concept of test image, this fusion method of SVM with VSM outperforms traditional SVM method by about 23% in accuracy. It shows that the proposed approach is effectiveness.

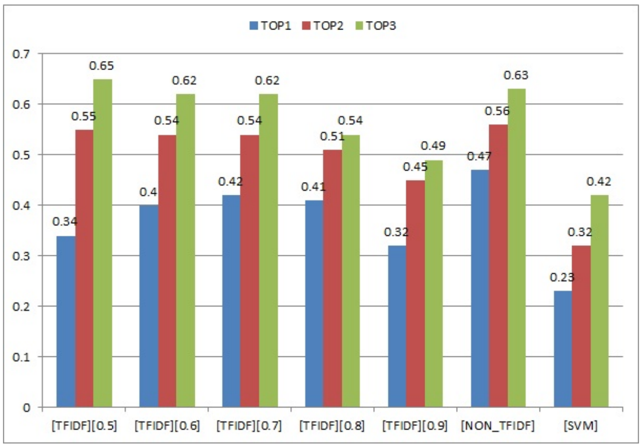


Fig 9. Accuracy of different experimental tests

4. Conclusions

Our approach focuses on de-noise and discovering the suitable annotation for an image. We employ the SVM for training concept classifiers and implement VSM for discovering candidate annotations. Then we choose the best annotation from these candidate annotations from VSM ranking. Our VSM approach actually can improve the accuracy of annotating an image. And the VSM approach can also be a kind of similar image search. In the experimental process, we found that the ranked results of top 1 to 10 images (base on sample images) in VSM approach are very similar to the test image

We randomly pick only 100 concepts for testing in this research. We will try more concepts in the future work. And our approach of choosing negative sample images by using hierarchical structure generated from WordNet is also an entirely new method of negative chosen. In this research, we use 20 near concepts and 20 far-away concepts form positive concept. We will try to implement different choosing approaches by using this hierarchical structure. Such as, randomly picks negative concepts from whole hierarchical tree. And then we will compare the accuracy when implementing different negative sample images choosing methods.

One image contents not only one concept; sometimes one image contents many concepts. In this research, we only annotate one concept for an image. Actually, it is better to annotate a set of keywords for an image. In the future works, we will annotate a set of keywords for a test-image, and discuss the recall and precision of our proposed method.

In this paper, we propose vector space model for improving the accuracy of image annotation. We will also implement other methods, such as LSA or pLSA. And discuss the recall and precision of different variety methods.

Acknowledgements

This research was supported by grant from JST "Multimedia Web Analysis Framework towards Development of Social Analysis Software," Grant-in-Aid for Scientific Research (No. 18049068), and Ambient SoC Global COE Program of Waseda University" of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] 秋間雄太，川久保秀敏，柳井啟司，"Folksonomyによる階層構造画像データベースの構築", MIRU 2010.

[2] Bangpeng Yao, Aditya Khosla, and Li Fei-Fei, "Combining Randomization and Discrimination for Fine-Grained Image Categorization", CVPR 2011, pp. 1577-1584

[3] Christopher Moretti, Karsten Steinhaeuser, Douglas Thain, and Nitesh V. Chawla, "Scaling Up Classifiers to Cloud Computers", ICDM 2008, pp. 472-481

[4] David G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", Computer Vision 2004, pp. 91-110

[5] Duy-Dinh Le, Shin'ichi Satoh, and Tomoko Matsui, "High Level Feature Extraction", TRECVID 2007

[6] Duy-Dinh Le and Shin'ichi Satoh, "A Comprehensive Study of Feature Representations for Semantic Concept Detection", TRECVID 2011

[7] Grigory Begelman, Philipp Keller, and Frank Smadja, "Automated Tag Clustering: Improving search and exploration in the tag space", WWW 2006

[8] Hangjun Ye and Guangyou Xu, "Fast Search in Large-Scale Image Database Using Vector Quatization", CIVR 2003, pp. 477-487

[9] Hao Zhang, Alexander C. Berg, Michael Maire, and Jitendra Malik, "SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition", CVPR 2006, pp. 2126-2136

[10] ImageNet, http://www.image-net.org/

[11] James Z. Wang, Jia Li, and Gio Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries", IEEE Trans. on Pattern Analysis and Machine Intelligence 2001, Vol.23 No.9, pp. 947-963

[12] Jerry Ye, Jyh-herng Chow, Jiang Chen, and Zhaohui Zheng, "Stochastic Gradient Boosted Distributed Decision Trees", CIKM 2009

[13] Jia Li and James Z. Wang, "Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach", IEEE Transactions on Pattern Analysis and Machine Intelligence 2003, Vol.25 No.9, pp. 1075-1088

[14] Marcus Rohrbach, Michael Stark, Gyorgy Szarvas, Iryna Gurevych, and Bernt Schiele, "What Helps Where-And Why? Semantic Relatedness for Knowledge Transfer", CVPR 2010, pp. 910-917

[15] Sergey Brin, "Near Neighbor Search in Large Metric Spaces", VLDB 1995

[16] Sung Ju Hwang, Fei Sha, and Kristen Grauman, "Sharing Features Between Objects and Their Attributes", CVPR 2011, pp. 1761-1768

[17] Tin Kam Ho, "The Random Subspace Method for Constructing Decision Forests", IEEE Trans. on Pattern Analysis and Machine Intelligence 1998, Vol.20 No.8, pp. 832-844

[18] Trevor Hastie and Robert Tibshirani, "Discriminant Adaptive Nearest Neighbor Classification", IEEE Trans. on Pattern Analysis and Machine Intelligence 1996, Vol.18 No.6, pp. 607-616

[19] Vincent. S. Tseng, Ja-Hwung Su, Bo-Wen Wang, and Yu-Ming Lin, "Web Image Annotation by Fusing Visual Features and Textual Information", SAC 2007

[20] Wei Li, and Maosong Sun, "Automatic Image Annotation Based on WordNet and Hierarchical Ensembles", CICLing 2006, pp. 417-428

[21] WordNet, http://wordnet.princeton.edu/

[22] Xiance Si and Maosong Sun, "Tag-LDA for Scalable Real-time Tag Recommendation", J. of Information & Computational Science 2009, pp. 23-31

[23] Yoshimasa Tsuruoka and Jun'ichi Tsujii, "Bidirectional Inference with the Easiest-First Strategy for Tagging Sequence Data", Proceedings of HLT/EMNLP 2005, pp. 467-474