

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN - ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



# BÁO CÁO ĐỒ ÁN THỊ GIÁC MÁY TÍNH

*TP. Hồ Chí Minh – 4/5/2021*

## MỤC LỤC

<b>I. Giới thiệu:</b>	<b>3</b>
A. Thông tin nhóm:	3
B. Phát biểu bài toán:	3
<b>II. Công trình liên quan:</b>	<b>4</b>
<b>III. Phương pháp:</b>	<b>5</b>
A. Mô hình phân loại:	5
B. Tối ưu hoá và mở rộng:	6
<b>IV. Kết quả thực nghiệm:</b>	<b>8</b>
A. Thực nghiệm:	8
B. Phân tích:	10
<b>V. Kết luận:</b>	<b>11</b>
<b>VI. Tài liệu tham khảo:</b>	<b>12</b>

# I. Giới thiệu:

## A. Thông tin nhóm:

STT	MSSV	Họ Tên	Email	Số điện thoại
1	18120468	Lê Hoài Nam	kimnam.cpc@gmail.com	0358783238
2	18120389	Trịnh Phú Hồng	phuhong2000@gmail.com	0908126935

## B. Phát biểu bài toán:

- Gắn thẻ cho hình ảnh (image tagging) là một thành phần quan trọng trong cơ sở dữ liệu hình ảnh có thể tìm kiếm như Flickr, Picassa hoặc Facebook. Tuy nhiên, một phần lớn (hơn 50% trong Flickr) hình ảnh không có thẻ nào cả, do đó chúng không thể tìm thấy bằng truy vấn văn bản. Tự động gắn thẻ cho hình ảnh là một công cụ được thiết kế để khắc phục điểm này.
- Gắn thẻ cho hình ảnh là công cụ đề xuất thẻ cho người dùng và do đó tăng số lượng hình ảnh được gắn thẻ, hoặc tạo các thẻ có liên quan để truy xuất hình ảnh.
- Gắn thẻ cho hình ảnh tự động (automatic image tagging) là một vấn đề khó trong học máy (machine learning). Các loại đối tượng khác nhau yêu cầu mô tả hình ảnh khác nhau.
- Trong các ứng dụng thế giới thực, số lượng hình ảnh có thể rất lớn. Hàng triệu hình ảnh được bổ sung mỗi ngày (ví dụ: 300 triệu hình ảnh được tải lên Facebook mỗi ngày, trong tổng số 100 tỷ hình ảnh), Những phương pháp giới thiệu ở trên không đáp ứng được yêu cầu này..
- Báo cáo này trình bày một thuật toán học tập mới để gắn thẻ hình ảnh đạt được độ chính xác tương đương với Guillaumin et al. (2009), nhưng có thể được đào tạo trong thời gian  $O(n)$  và ổn định quá trình thử nghiệm với các kích thước tập huấn luyện khác nhau.
- Thuật toán FastTag, có thể kết hợp một cách tự nhiên nhiều bộ mô tả hình ảnh và giải quyết những khó khăn về độ thừa thớt của nhãn bằng một cách tiếp cận mới. Nó diễn giải dữ liệu đào tạo của nó thành tập dữ liệu dưới và tạo lại bộ nhãn hoàn chỉnh (từ một vài thẻ có sẵn trong quá trình đào tạo; và các nhãn khác trong quá trình học dữ liệu. Thuật toán được chứng minh trên tập dữ liệu thế giới thực, với sự chính xác cao và khả năng thu hồi, nhanh trong quá trình học, gần như tức thời trong quá trình thử nghiệm.

## II. Công trình liên quan:

Trong phần này, chúng tôi xem xét một số phương pháp phổ biến để gắn thẻ hình ảnh tự động:

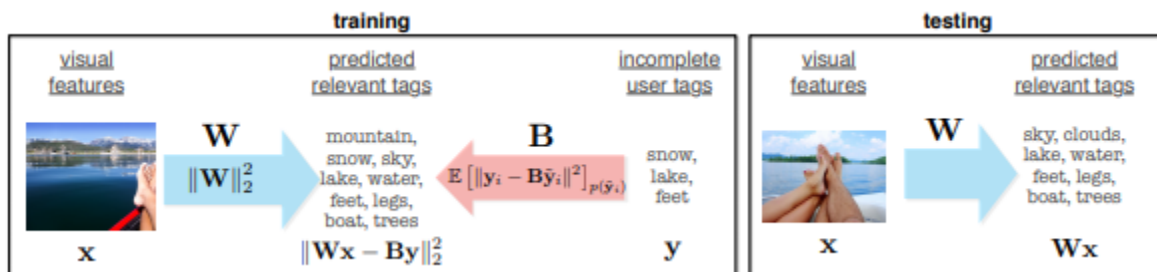
- Nhóm phương pháp đầu tiên dựa trên mô hình chủ đề tham số. Monay & Gatica-Perez (2004) mở rộng mô hình phân tích ngữ nghĩa tiềm ẩn theo xác suất, và Barnard et al. (2003) mở rộng mô hình phân bố tiềm ẩn sang dữ liệu đa phương thức. Mỗi hình ảnh được chú thích được tạo mô hình như một hỗn hợp các chủ đề trên các tính năng trực quan và văn bản. Tỷ lệ hỗn hợp được chia sẻ giữa các chế độ tính năng, nhưng sự phân bố chủ đề là khác biệt.
- Nhóm phương pháp thứ hai (Jeon và cộng sự, 2003; Lavrenko và cộng sự, 2003; Feng và cộng sự, 2004) lập mô hình phân phối chung của các đặc điểm hình ảnh và các thẻ với các mô hình hỗn hợp.
- Nhóm phương pháp thứ ba đào tạo các mô hình phân biệt, chẳng hạn như SVM (Cusano và cộng sự, 2003), xếp hạng SVM (Grangier & Bengio, 2008) và tăng cường (Hertz và cộng sự, 2004), để dự đoán các thẻ từ các đặc điểm hình ảnh. Trong khi các phương pháp này đạt được kết quả đầy hứa hẹn, các quy trình đào tạo phức tạp của chúng hạn chế số lượng bộ mô tả có thể được kết hợp. Các mô hình được đề xuất gần đây như mô hình Đóng góp bình đẳng chung của (Makadia và cộng sự, 2008) và mô hình TagProp của (Guillaumin và cộng sự, 2009) dựa trên các vùng lân cận gần nhất của địa phương và hoạt động hiệu quả một cách đáng ngạc nhiên mặc dù chúng đơn giản. TagProp là phương pháp hiện đại nhất dành cho chú thích hình ảnh. Thành công của nó có thể là được quy cho ba yếu tố: 1. nó kết hợp một số lượng lớn các bộ mô tả trực quan khác nhau; 2. nó có thể được đào tạo một cách hiệu quả trên những hình ảnh có bộ thẻ chưa hoàn chỉnh; 3. nó xử lý các thẻ hiếm đặc biệt. Mặc dù Tagprop đạt được hiệu suất vượt trội trên một số bộ dữ liệu chuẩn, độ phức tạp của quá trình huấn luyện  $O(n^2)$  và  $O(n)$  cản trở khả năng áp dụng của nó cho các bộ dữ liệu quy mô lớn (trong đó  $n$  là số lượng ví dụ trong bộ huấn luyện).

### III. Phương pháp:

- Giả định các trường hợp của các ứng dụng trong thế giới thực, trong quá trình training, tập dữ liệu gồm các hình ảnh được gắn với các thẻ có liên quan. Mục tiêu là tìm ra danh sách tất cả các thẻ liên quan đến một hình ảnh trong quá trình thử nghiệm. Thuật toán được đề xuất này rất nhanh trong quá trình đào tạo và dự đoán gần như tức thì trong quá trình thử nghiệm (chỉ cần một phép biến đổi tuyến tính). Do đó, thuật toán này được gọi là FastTag.
- Ký hiệu:  
Gọi  $T = \{\omega_1, \dots, \omega_T\}$  biểu thị các thẻ của T thẻ chú thích có thể có.
- $D = \{(x_1, y_1), \dots, (x_n, y_n)\} \subset \mathbb{R}^d \times \{0, 1\}^T$ ,  $D$  là tập dữ liệu huấn luyện  
Trong đó mỗi vector  $x_i \in \mathbb{R}^d$  đại diện cho các đặc trưng được trích xuất từ hình ảnh thứ  $i$  và mỗi  $y_i$  là một tập hợp các thẻ thích hợp cho hình ảnh thứ  $i$ .
- Mục tiêu của chúng tôi là tìm hiểu một hàm tuyến tính  $W: \mathbb{R}^d \rightarrow T$ , hàm này ánh xạ một hình ảnh thử nghiệm  $x_i$  vào tập thẻ hoàn chỉnh của nó.

#### A. Mô hình phân loại:

- Trong phần này, chúng tôi giới thiệu một mô hình mới để tự động gắn thẻ cho hình ảnh từ các thẻ chưa hoàn chỉnh. Được phân loại thành hai nguồn, bao gồm hình ảnh và văn bản, để thống nhất danh sách các thẻ được dự đoán cho mỗi hình ảnh.



#### - Co-regularized learning:

Vì tập dữ liệu chỉ có các bộ thẻ chưa hoàn chỉnh, cho nên cần xử lý thêm các vấn đề phụ như sau:

- 1) Training phân loại hình ảnh  $x_i \rightarrow Wx_i$  dự đoán các thẻ hoàn chỉnh từ tập dữ liệu các hình ảnh
- 2) Training một ánh xạ  $y_i \rightarrow By_i$  để tăng thêm sự hiện diện của vector  $y_i$  trong tập dữ liệu bằng cách ước tính những thẻ nào có khả năng xảy ra giống như những thẻ đã có trong  $y_i$ . Đào tạo cả 2 bộ dữ liệu và ép đầu ra của nó tuân thủ giảm thiểu:

$$\frac{1}{n} \sum_{i=1}^n \|B y_i - W x_i\|^2. \quad (1)$$

Trong đó:

$B y_i$  là các thẻ được thêm vào cho hình ảnh thứ  $i$  và mỗi hàng của  $W$  chứa các trọng số của bộ phân loại tuyến tính cố gắng dự đoán thẻ dựa trên các đặc trưng của hình ảnh

Hàm mất mát

$B = 0 = W$ , là hàm mất mát (The loss function) cho thấy công thức hiện tại vẫn còn khả năng mở rộng.

- **Marginalized blank-out regularization:**

Xây dựng ánh xạ làm giàu thẻ  $B: \{0, 1\}^T \rightarrow \mathbb{R}^T$ .

Mục đích là làm phong phú thêm các thẻ bằng các từ khoá.

Xây dựng tập dữ liệu các thẻ chưa được gắn thẻ và training ngược từ tập dữ liệu và nếu cơ chế này phù hợp thì áp dụng cho tập dữ liệu với ảnh ban đầu.

- **Joint loss function:**

Sự mất mát theo công thức:

$$\begin{aligned} \ell(B, W; x, y) &= \underbrace{\frac{1}{n} \sum_{i=1}^n \|B y_i - W x_i\|^2}_{\text{Co-regularization}} + \underbrace{\lambda \|W\|_2^2}_{\gamma^r(B)}. \quad (5) \\ &\quad \text{Marginalized blank-out} \end{aligned}$$

Thuật toán thực thi và gắn các thẻ vào các hình ảnh khi có sự trùng khớp giữa các thẻ dự đoán và nội dung hình ảnh. Điều tiết trên tập dữ liệu để giảm độ phức tạp và đảm bảo sự tin cậy, đặc biệt kiểm tra sự phong phú của tập dữ liệu các thẻ khi xoá thẻ.

## B. Tối ưu hoá và mở rộng:

Sự mất mát trong thuật toán có thể được tối ưu hóa hiệu quả bằng cách sử dụng gốc tọa độ khối. Khi  $B$  cố định,  $W$  giảm xuống tiêu chuẩn và có thể được tính bằng:

$$W = B Y X_T (X X_T + n \lambda I)^{-1}$$

trong đó  $X$  và  $Y$  tương ứng chứa hình ảnh training các đặc trưng và thẻ.

Tương tự, khi  $W$  cố định, phương trình mất mát có thể được thể hiện dưới dạng bình phương nhỏ nhất:

$$B = (\gamma P + WXY)(\gamma Q + YY)^{-1}$$

trong đó  $P$  và  $Q$  có thể được tính toán theo biểu thức:

$$\begin{aligned} P &= (1-p)YY^T \\ Q &= (1-p)^2YY^T + p(1-p)\delta(YY^T). \end{aligned}$$

Như vậy, chúng ta có thể tìm ra phương pháp tối ưu ánh xạ  $B$ , do các thẻ có giá trị như nhau, dẫn tới mất mát bị chi phối. Chúng ta sửa lỗi bằng cách sử dụng các trọng số cho các thẻ, để các thẻ xuất hiện thường xuyên có trọng số cao hơn, hy sinh độ chính xác của các thẻ xuất hiện hiếm.

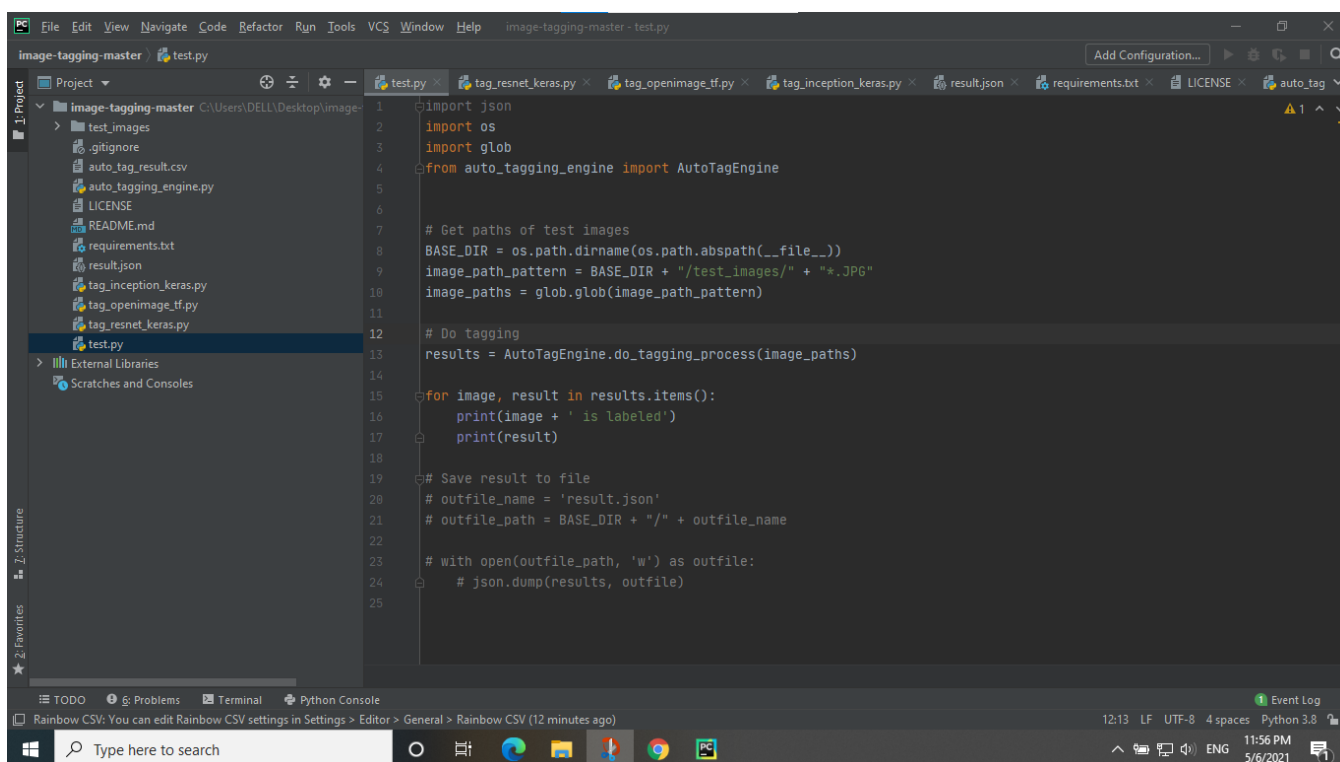
Tối ưu hoá trên từng giai đoạn, trong đó ở mỗi giai đoạn, xác định một tập hợp con các thẻ dưới ngưỡng quy định (mức thu hồi trung bình), ưu hóa lại  $B$  và  $W$  tương ứng.

## IV. Kết quả thực nghiệm:

- Chúng tôi thực hiện FastTag trên bộ dữ liệu chuẩn. Tất cả các tập dữ liệu và mã nguồn được lấy từ <https://github.com/harrywang/image-tagging>

### A. Thực nghiệm:

- Chuẩn bị source code và các dữ liệu trong folder Source
- Cài đặt môi trường cũng như các package yêu cầu (trong file requirements.txt)
- Chạy thử file test.py với các hình ảnh thử nghiệm trong folder test\_images (Hướng dẫn chi tiết hơn đọc file readme.md)



```
1 import json
2 import os
3 import glob
4 from auto_tagging_engine import AutoTagEngine
5
6
7 # Get paths of test images
8 BASE_DIR = os.path.dirname(os.path.abspath(__file__))
9 image_path_pattern = BASE_DIR + "/test_images/" + "*.JPG"
10 image_paths = glob.glob(image_path_pattern)
11
12 # Do tagging
13 results = AutoTagEngine.do_tagging_process(image_paths)
14
15 for image, result in results.items():
16     print(image + ' is labeled')
17     print(result)
18
19 # Save result to file
20 # outfile_name = 'result.json'
21 # outfile_path = BASE_DIR + "/" + outfile_name
22
23 # with open(outfile_path, 'w') as outfile:
24 #     json.dump(results, outfile)
25
```



- Xem kết quả khi chạy test.py từng ảnh trong result.json
- Xem kết quả tất cả ảnh trong file auto\_tag\_result.csv với các ảnh airshow.jpg, bike.jpg, img\_1.jpg, img\_2.jpg. Kết quả như sau:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	image_na	Isle of ma	Amuseme	Roller coa	Granny sm	Air show	Reflex car	Membran	String inst	Wind insti	Gourd	Lasagne	Tortoise	Team	Vespa	Acrylic pa	Advertisir	Meringue	Stained gl	Plantatio	Stone
2	img_1.JPG	0.000255	0.004161	0.001671	0.000272	0.000249	0.000232	0.000275	0.000434	0.000339	0.000247	0.000239	0.00025	0.000418	0.000248	6.47E-05	0.002785	0.000269	0.000651	0.000267	0.000
3	img_2.JPG	0.000223	0.013207	0.000481	0.000271	0.000211	0.000215	0.001017	0.000252	0.0003	0.000662	0.000225	0.000242	0.000112	0.000301	0.000867	0.001474	0.000252	0.000588	0.000304	0.000
4	airshow.JPG	0.000338	0.001462	0.000859	0.000248	0.612608	0.000164	0.000119	0.000157	0.000264	0.000232	0.000215	7.96E-05	0.000385	0.000314	0.000228	0.000235	0.000243	0.000217	0.000126	0.000
5	bike.jpg	0.000353	0.000226	0.000443	0.000216	0.000181	0.000377	0.000569	0.0003	0.000272	0.000194	0.000188	0.00028	0.000584	0.000617	0.000149	0.000366	0.000205	0.000155	0.000162	0.000

## B. Phân tích:

- Hình mô tả sự so sánh của FastTag và TagProp (thuật toán ở công trình liên quan) ở các mức độ thưa thớt khác nhau của tập thẻ trong tập dữ liệu đào tạo.

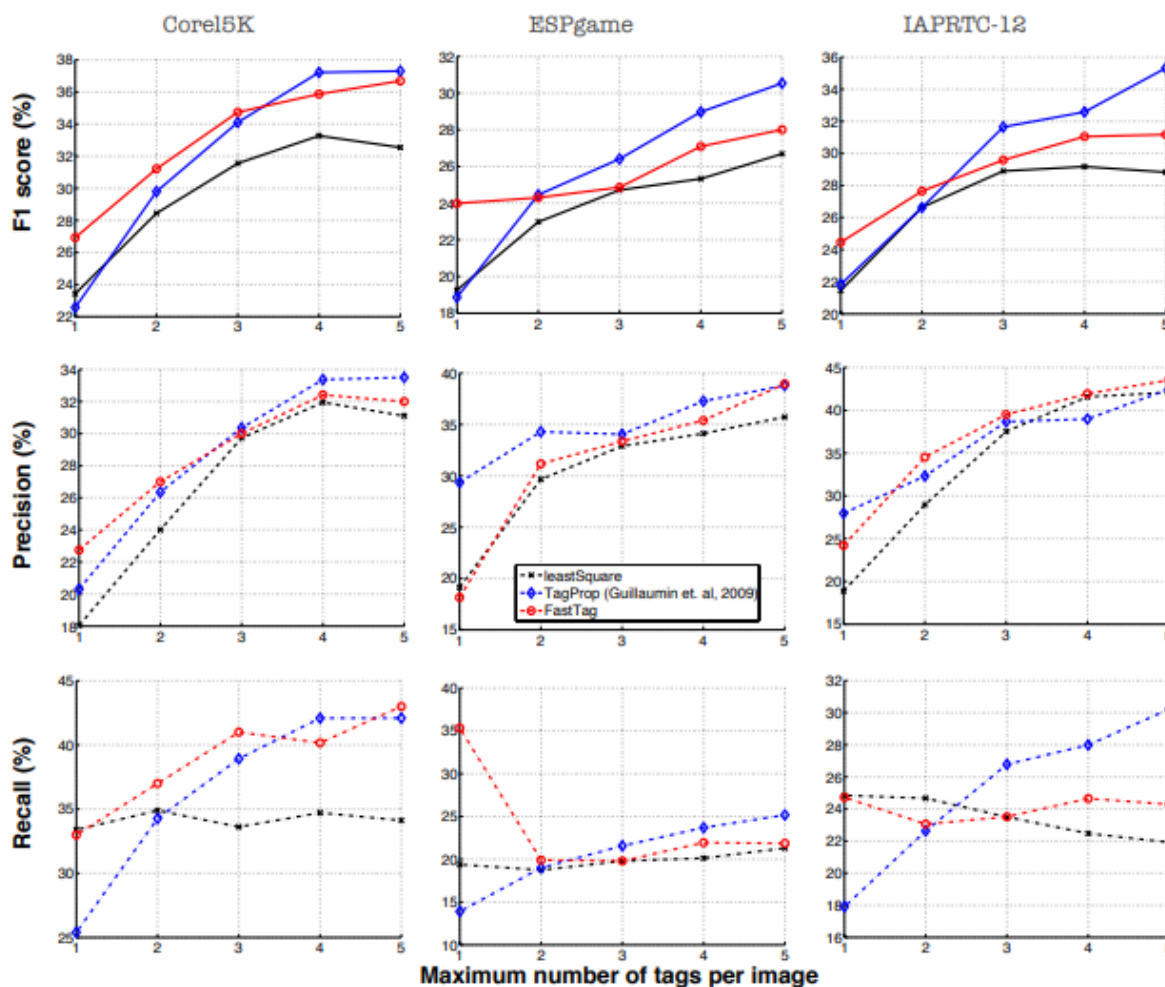


Figure 4. Performance in terms of Precision (P), Recall (R) and F1 score (F1) as a function of the maximum number of tags provided for each training image on the three benchmark datasets. The graphs compare the results of FastTag with the TagProp algorithm at different levels of tag sparsity.

- Chúng ta có thể thấy rằng FastTag hoạt động tốt hơn TagProp khi số lượng thẻ được cung cấp là nhỏ. Ở các mức độ khác, FastTag hoạt động tương đương với Tagprop.
- Nói cách khác, tính năng ánh xạ làm phong phú thẻ của FastTag giúp giảm bớt vấn đề về sự thưa thớt của tập thẻ mẫu.

## V. Kết luận:

- Phương pháp gắn thẻ cho hình ảnh, FastTag, được thực hiện với hiệu năng bằng với các thuật toán hiện đại, nhưng với chi phí tính toán nhỏ hơn. Giải quyết vấn đề phân loại nhiều nhãn, dưới dạng học nhiều từ chế độ xem không gắn nhãn.
- Xác định hai bộ phân loại, một bộ phân loại cho mỗi chế độ xem dữ liệu và buộc chúng phải thỏa thuận thông qua đồng chính quy hóa trong một chức năng. Đánh đổi sự phức tạp trong bộ phân loại bằng ánh xạ phi tuyến tính của các đối tượng và chứng minh rằng sự lựa chọn như vậy có lợi. FastTag hiệu quả về mặt tính toán trong quá trình training và thử nghiệm nhưng vẫn duy trì độ chính xác của việc gắn thẻ một cho mỗi lần xem dữ liệu,
- Điều này giúp xử lý hiệu quả các dữ liệu lớn được gắn thẻ thừa thớt trong quá trình training.

## VI. Tài liệu tham khảo:

- Source code: <https://github.com/harrywang/image-tagging>
- Fast Image Tagging, Minmin Chen, Alice Zheng, Kilian Q. Weinberger