

Image-to-text Model for Social Media Network Analysis

Team 29

Chih-Han Yeh, Ching Han Chang, Vanessa Garcia de Aquino

PROJECT STATEMENT

In today's digital age, social media plays a central role in how we connect, communicate, and express ourselves. However, the abundance of data generated on these platforms can be overwhelming, making it challenging to navigate and extract meaningful insights. Our project seeks to address this issue by expanding and enhancing an existing application, NodeXL, transforming it into a more comprehensive social media analysis tool.

Our primary goal is to improve the user experience by introducing new features that facilitate a more dynamic and comprehensive exploration of social media networks. By doing so, we aim to empower users to interact with the data more effectively, enabling them to uncover valuable insights and make informed decisions. More specifically, our project focuses on integrating image analysis to enrich the social media analytics framework

METHODOLOGY

We propose a methodology that leverages models and cloud computing resources. Our approach involved the integration of **Amazon Rekognition Image**, **Microsoft Kosmos-2**, and **Salesforce BLIP-2** for image analysis. We chose **Rekognition** due to its cloud-based image analysis service, while Kosmos-2 and BLIP-2 feature as some of [the most popular image-to-text models on Hugging Face](#). We utilized AWS EC2 instances to harness the computing power required for running the models. In order to illustrate our methodology and the outputs from the above method, we will use the **Image 1** below:



Image 1.⁴

1. Amazon Rekognition:

Amazon Rekognition is a cloud-based image and video analysis service provided by Amazon Web Services (AWS). It provides labels (such as 'Outdoors', 'Nature', 'Flame', 'Bonfire') along with confidence scores indicating the likelihood of each label being

present in the image. Additionally, it shows parent labels for some entries, indicating hierarchical relationships (e.g., 'Bonfire' is a type of 'Fire'). For the sample image the outputs are:

```
Labels: [{ 'Name': 'Outdoors', 'Confidence': 98.39 }  
{ 'Name': 'Nature', 'Confidence': 97.77 }  
{ 'Name': 'Flame', 'Confidence': 96.74 }  
{ 'Name': 'Bonfire', 'Confidence': 90.43, 'Parents': [{ 'Name':  
'Fire' }] }  
{ 'Name': 'Snowman', 'Confidence': 81.52 } }
```

Recognition does not provide captions, only **labels**.

2. Kosmos-2:

Microsoft Kosmos does not directly generate labels for images. Instead, it is a large language model (LLM) designed primarily for natural language understanding and generation tasks. We use the [microsoft/kosmos-2-patch14-224 from Hugging Face](#), in which the pre-trained auto class model ([AutoModelForVision2Seq](#)) uses Microsoft Kosmos for generating descriptive text based on the image. Ultimately, that generates labels or tags based on the visual features of the image. For the sample image the outputs are:

```
Caption: 'a snowman sitting by a campfire with a pot of hot water'
```

```
Entities: `[('a snowman', (12, 21), [(0.390625, 0.046875, 0.984375,  
0.828125)]), ('a fire', (41, 47), [(0.171875, 0.015625, 0.484375,  
0.890625)])]`
```

Kosmos-2 can provide both **entities** and **captions**.

It is worth mentioning the difference between labels and entities. Labels in image analysis are descriptive tags assigned to images to describe their content, such as objects or scenes. Entities, on the other hand, represent specific objects or regions detected within an image, providing detailed information about individual instances of objects present in it.

3. Salesforce BLIP-2:

BLIP-2 consists of 3 models: a CLIP-like image encoder, a Querying Transformer (Q-Former) and a large language model, as described in [Salesforce/blip2-opt-2.7b · Hugging Face](#). The goal for the model is simply to predict the next text token, giving the query embeddings and the previous text, which allows the model to be used for tasks like image captioning. For the sample image the outputs are:

```
Caption: 'a snowman sitting by a campfire with a pot of hot water'
```

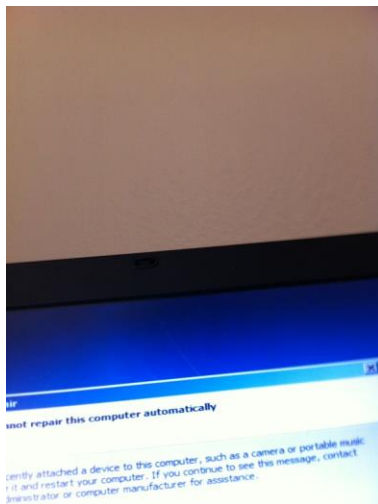
BLIP-2 provides **captions**. Although lacking entity extraction capabilities, post-processing with spaCy can be deployed to extract entities from the captions, as shown below.

```
Extracted entities using spaCy: [snowman, campfire, pot, water]
```

By testing three different image-to-text models, we aim at a comprehensive evaluation of performance, robustness, and applicability across diverse scenarios. Amazon Rekognition, a widely used service, provides established benchmarks in image analysis. Kosmos-2, with its unique large language model capabilities, offers insights into nuanced text generation from images. BLIP-2, another large language model, presents an alternative approach, enriching the analysis. By assessing these models together, we gain a holistic understanding of their strengths and limitations, informing the selection of the most suitable model.

EVALUATION STRATEGY

In order to evaluate the outcomes from the models, we collected the [VizWiz-Captions dataset](#) as the groundtruth captions for the models. The dataset consists of 39,181 images each paired with 5 captions originating from people who are blind. Sample image-captions from VizWiz validation dataset is as **Image 2**. Our proposed challenge addresses the task of predicting a suitable caption given an image.



Caption 1: "A computer screen shows a repair prompt on the screen."

Caption 2: "a computer screen with a repair automatically pop up"

Caption 3: "partial computer screen showing the need of repairs"

Caption 4: "Part of a computer monitor showing a computer repair message."

Caption 5: "The top of a laptop with a blue background and dark blue text."

Image 2. Sample image from VizWiz validation dataset (image_id: 23431) with respective captions.

First, we evaluated **Kosmos-2** and **BLIP-2 captions** by **BLEU scoring**, which compares the matches between the Vizwiz image captions as ground truth. BLEU has been used for evaluating translation⁵. The caveat of using it on image-to-text captions is that captions can have higher variations as they transcribe images and not sentences. However, we have five

reference sentences for each image from Vizwiz, which we can compare with each other to get an average baseline to compare our models.

Our first attempt focused on comparing the five baseline captions against the LLM outputs. One of our initial findings was that both models will try to infer information despite the quality issues from the image (e.g. blurry, too bright). Thus, we removed the images whose quality problems were too severe to infer the content. The resulting dataset size was 5199 rows (-13.42%) after filtering. A second consideration was to call the SmoothingFunction() method in order to address issues where certain n-grams (sequences of words) in a generated text may not overlap with those in a reference text, leading to zero counts and consequently a BLEU score of zero, despite potential partial semantic similarity. BLIP-2 performed better over the majority of the image IDs by an average difference of 0.0435, however, there was a weak linear correlation between the output using the initial strategy ($\text{corr} = 0.0763$). Here is a sample for the first 25 ID pictures considering the best score among the five captions.

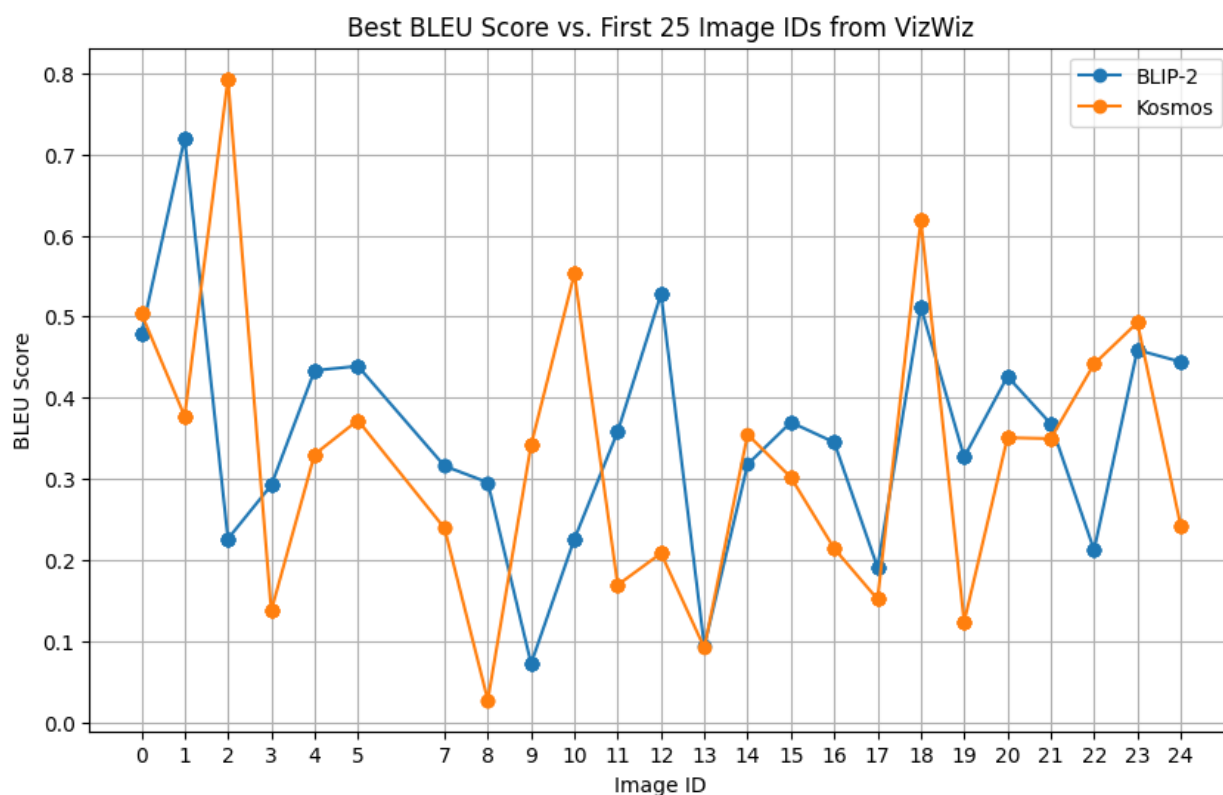


Image 3 - Sample score that illustrates the differences in the models behavior. Image_ID 6 was removed due to quality issues.

During the error analysis, we sampled some of the low-performing caption results. Higher BLEU scores indicate better similarity between generated and reference captions, but we must be cautious of its limitations, such as inability to capture semantic nuances. In the examples below we could notice some of those limitations.

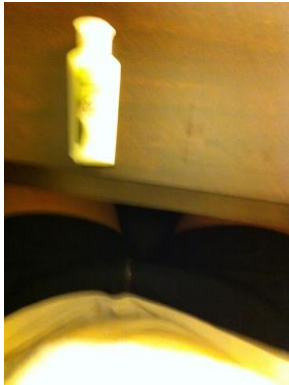


Image ID	8	3	18
			
VizWiz caption (Reference)	"A bottle of lotion on the table in front of someone sitting down wearing shorts."	"A small rectangular red and white box next to a small rectangular blue box on a wooden surface."	<p>"part of the keyboard and screen on a laptop" (Kosmos)</p> <p>—</p> <p>"A black laptop with the computer's specs on the screen." (BLIP-2)</p>
Kosmos-2 caption	"a woman peeing into a cup"	"two boxes of condoms"	"a laptop with a broken screen and keyboard"
BLIP-2 caption	"a bottle of shampoo on a table"	"a toothbrush and a tube of toothpaste on a table"	"a laptop computer with a black screen"

Table 1. Examples of model outputs vs. ground truth caption (best score). Green captions indicate a high BLEU score, whereas red captions indicate a mislabelling.

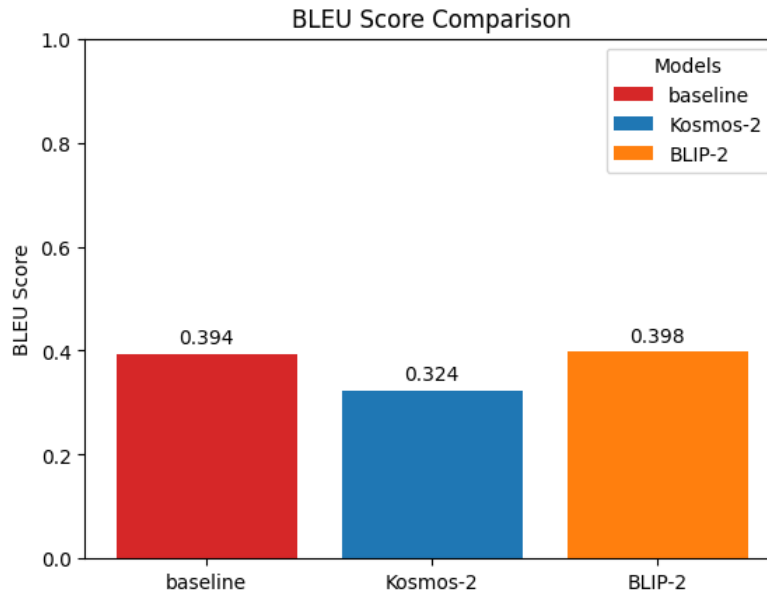


Image 4. Overall performance of Kosmos-2 and BLIP-2 models in caption generation. BLIP-2 is on par with human references in describing the images.

Amazon Rekognition only generates labels instead of captions, so we have developed a different method of comparing model output with ground truth. First, we perform text preprocessing on the captions from VizWiz and labels from AWS Rekognition. Our text preprocessing process includes lowercasing, tokenization, stopwords removal, and lemmatization. Text preprocessing is crucial because it can help the models learn better representations. After text preprocessing, we convert the collection of texts with two different methods and different similarity estimation measures.

- Method 1. convert into a matrix of TF-IDF features. Then, we use cosine similarity to assess how similar two documents are to each other. Cosine similarity can measure similarity by measuring the cosine of the angle between two vectors. When the cosine similarity value is closer to one, it means the documents are more similar.
- Method 2. convert into a matrix of token counts. Then, we use Jaccard similarity to assess the similarity between the two documents. A Jaccard score is the number of the intersection divided by the number of the union of two sets. When the Jaccard similarity value is closer to one, it means the documents are more similar.

The TF-IDF representation of VizWiz captions and Rekognition labels have an average cosine similarity score of 0.122. On the other hand, the BoW (Bag of Words) representation of VizWiz captions and Rekognition labels has an average Jaccard similarity score of 0.236.

Here is a sample for the first 25 ID pictures considering the cosine similarity scores and Jaccard similarity scores.

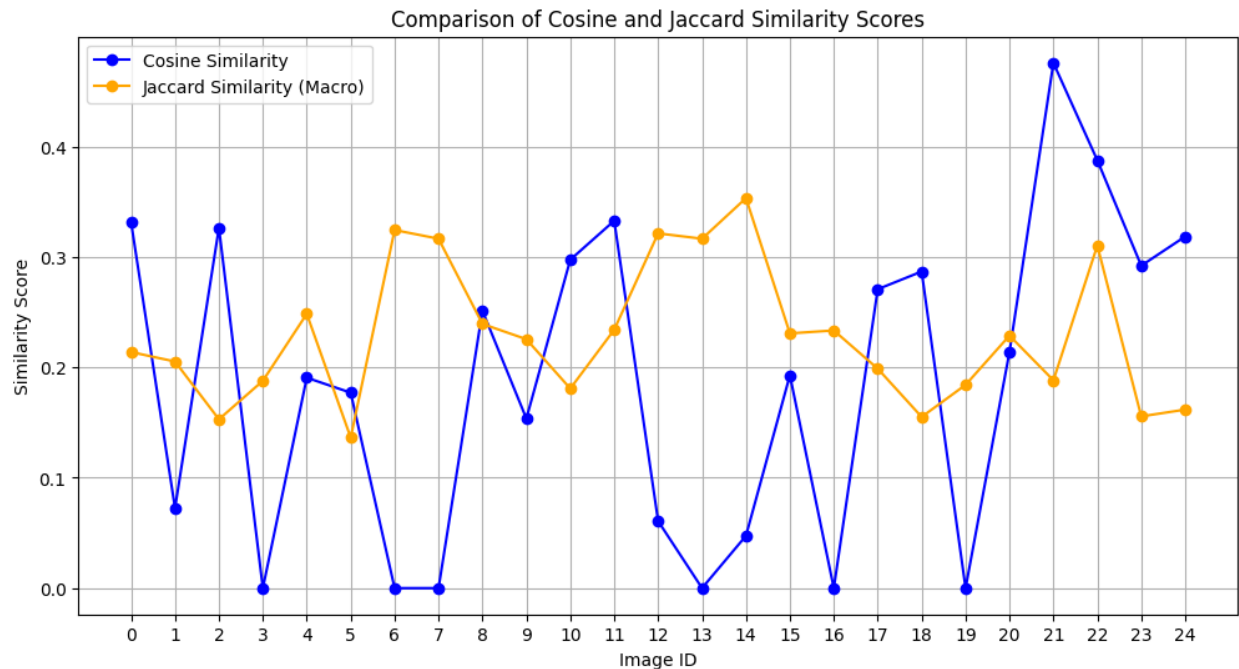

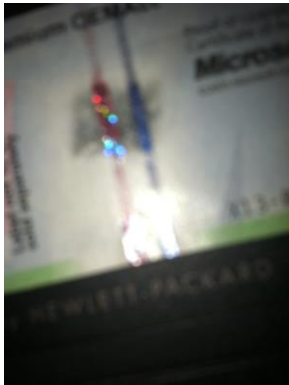
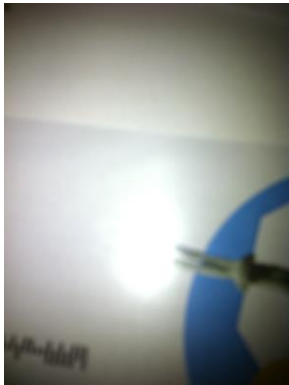


Image 5. Sample score that illustrates the differences in the cosine and Jaccard scores behavior within AWS Rekognition model.

In **Image 4**. ID 6, 7, and 13, the similarity scores are quite different. We can identify some limitations as shown in **Table 2**. within the AWS Rekognition image-to-text model.

Image ID	6	7	13
			
VizWiz caption (Reference)	"Quality issues are too severe to recognize visual content."	"A close-up of a package of a computer-related item."	"A white piece of paper with a blue emblem off to the side."
AWS Rekognition Labels	['Flare', 'Light', 'Lighting', 'Sunlight']	[]	['Lighting', 'Flare', 'Light', 'Nature',

			'Outdoors', 'Sky', 'Sun']
--	--	--	---------------------------

Table 2. Examples of model outputs vs. ground truth caption. The red captions indicate a mislabelling.

ETHICAL CONSIDERATIONS

Ethical considerations for image-to-text models encompass various facets, including bias and fairness, misinterpretation, misrepresentation, and cultural sensitivity. Although the project's target was not to specifically train an algorithm, we relied on existing models trained to pursue our analyses and evaluations. Thus, there is a risk perpetuating societal biases, leading to unfair outcomes based on biased datasets. Additionally, there's a fear of misinterpreting images, generating text that misrepresents their context and potentially disseminating misinformation. Lastly, cultural nuances may be overlooked, resulting in insensitive descriptions. For example, BLIP2, fine-tuned on internet-collected datasets like LAION, may inadvertently replicate biases or generate inappropriate content. Its untested nature underscores the importance of thorough safety and fairness assessments before deployment in any real-world applications.

APPLICATION

Based on the [dataset of social media \(X, formerly Twitter\) posts that mention #MWC24](#) (2024 Mobile World Congress) and contain images provided by our client, the Social Media Research Foundation, we utilize AWS Rekognition to conduct image analysis for a total of 3,178 unique images and generate image-labels data. The data contains a series of labels for each image. Based on the data, we perform network analysis and identify the representative images in the entire image network.

For the image network analysis, we create a graph with its nodes are the images and edges being built based on the presence of any shared label between two images. The weight is the number of shared labels between image pairs.

We then use different centrality measures to identify the most central image in the entire network. The centrality measures are eigenvector centrality, degree centrality, closeness centrality, and betweenness centrality. The corresponding most representative images being identified are shown in **Table 3**.

Most central image based on eigenvector centrality	Most central image based on degree centrality	Most central image based on closeness centrality	Most central image based on betweenness centrality
--	---	--	--

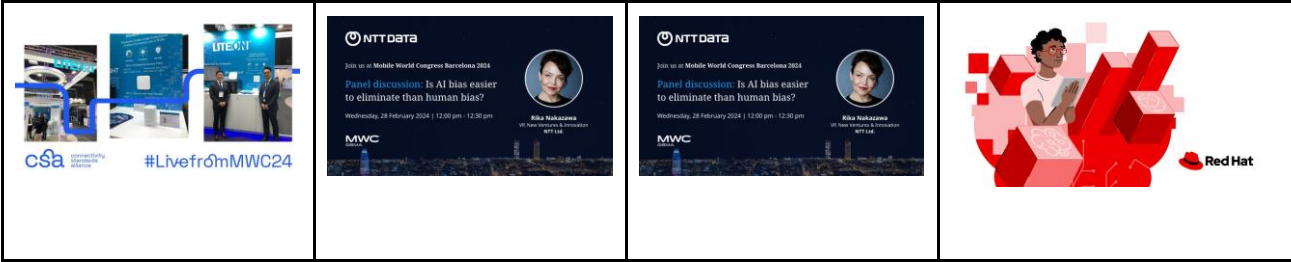


Table 3. The most representative images identified by 4 different centrality measures within the entire network.

The labels of the most representative image being identified by eigenvector centrality are: ['Advertisement', 'Poster', 'Person', 'Computer Hardware', 'Electronics', 'Hardware', 'Architecture', 'Building']. The labels of the one being identified by both degree centrality and closeness centrality are: ['Advertisement', 'Adult', 'Female', 'Person', 'Woman', 'Face', 'Head', 'Nature', 'Night', 'Outdoors', 'Text', 'Architecture', 'Building', 'Tower']. The most central image labels based on betweenness centrality are: ['Person', 'Logo', 'Art', 'Graphics', 'Face', 'Head']. We find that the 'Person' is the key shared label, and the most central image identified by eigenvector centrality, degree centrality, and closeness centrality is closely related to 'Advertisement', which aligns with the context of the analysis of tweets that mention #MWC24 and include images.

For the network building and centrality computation, it takes over 49 minutes to process a network of a total of 3,178 images with CPU, which may be a potential challenge when implementing the application to NodeXL product because it may be a bit long for the product user. Nevertheless, the client has a plan to utilize cloud computing services to decrease the needed computational time.

Besides, with the data of image pairs, the number of shared labels between image pairs, and the actual list of the shared labels generated based on the image-to-text output, the client generates visualizations of the complicated network with their network analysis product, NodeXL. Based on the network information, the client creates the network visualization by first filtering out the weight (the number of shared labels) less than 10 to reduce the size of the dataset and de-complicate the output visualization. As shown in **Figure 4**, the generated network visualization contains clusters of images and we can clearly see the differences in different clusters of images and text describing the images.

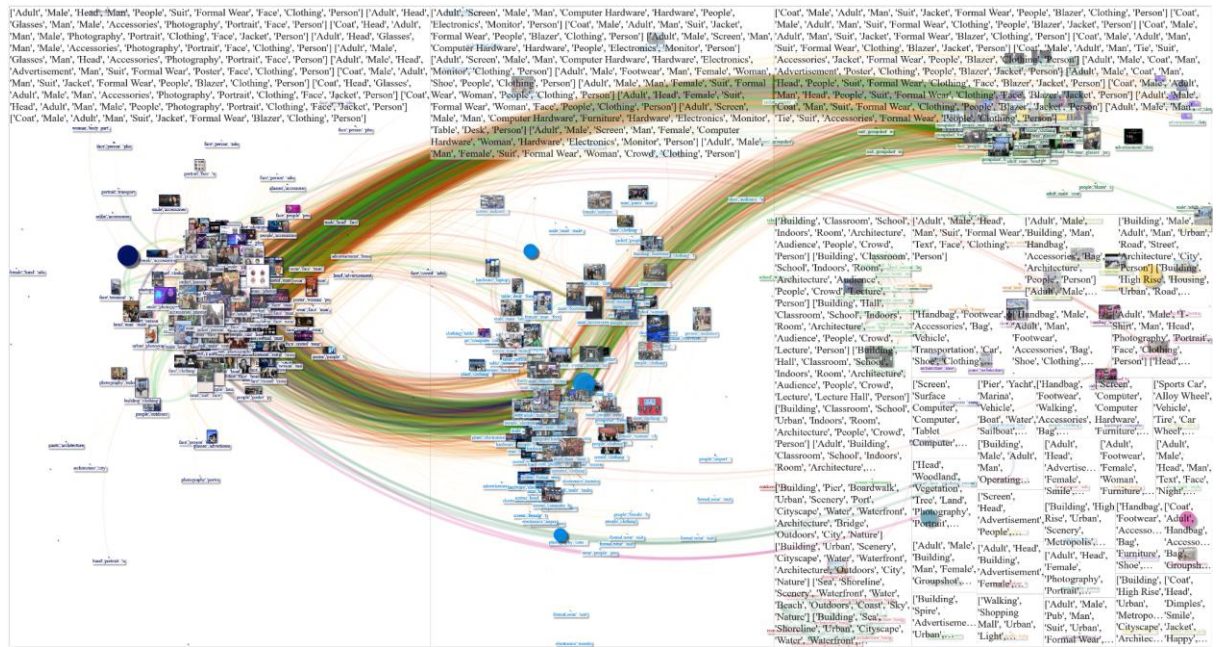


Figure 4. Image network visualization (after filter out weight less than 10).

The visualization as shown in **Figure 5** demonstrates a grid layout of the network. It is a functionality inherent in the NodeXL that users can manipulate and decide the preferred network layout. In the grid layout, the images are ranked based on their betweenness centrality, with the bottom four images having the highest betweenness centrality values. This indicates that these images have a significant influence within the network due to their crucial role in controlling the flow of information between other nodes. The representative image for each cluster can be clearly recognized in the grid layout visualization when the number of images in the network is not too large.

In addition to the network visualization, NodeXL can bring deeper analysis into the network. It generates an Excel file regarding the network on a granular level. For example, it presents the information of network top labels identified by NodeXL in **Table 2**. Kind reminder that the network analyzed and visualized by NodeXL is after filtering out the weight (the number of shared labels) less than 10 for demonstration purposes.

Top shared_labels in Entire Graph	Top shared_labels in G1	Top shared_labels in G2	Top shared_labels in G3	Top shared_labels in G4
['Adult', 'Male', 'Head', 'Man', 'People', 'Suit', 'Formal Wear', 'Face', 'Clothing', 'Person']	['Adult', 'Male', 'Head', 'Man', 'People', 'Suit', 'Formal Wear', 'Face', 'Clothing', 'Person']	['Adult', 'Screen', 'Male', 'Man', 'Computer Hardware', 'Hardware', 'People', 'Electronics']	['Coat', 'Male', 'Adult', 'Man', 'Suit', 'Jacket', 'Formal Wear', 'People', 'Blazer', 'Clothing', 'Person']	['Building', 'Classroom', 'School', 'Indoors', 'Room', 'Architecture', 'Audience']

data demands significant computational resources, potentially leading to longer processing times, increased hardware costs, and scalability issues, particularly for resource-constrained environments. For this project, we utilize AWS EC2 to perform image-to-text modeling and evaluation.

Future work

With more time and resources, the directions for extending the project can be:

1. Broader analysis by including text and user: Bring the text and author data back into the image data. Conduct broader analysis into not only the images but also the author of the image and the text associated with the image. With this, a deeper understanding of the influential social media users and topics can be achieved.
2. Sentiment Analysis Network: Use labels extracted from images and textual content related to the images, such as hashtags or text descriptions, to perform sentiment analysis. This allows for understanding of popular topics and community sentiment, which is valuable for marketing and campaign strategies.

Practical implications

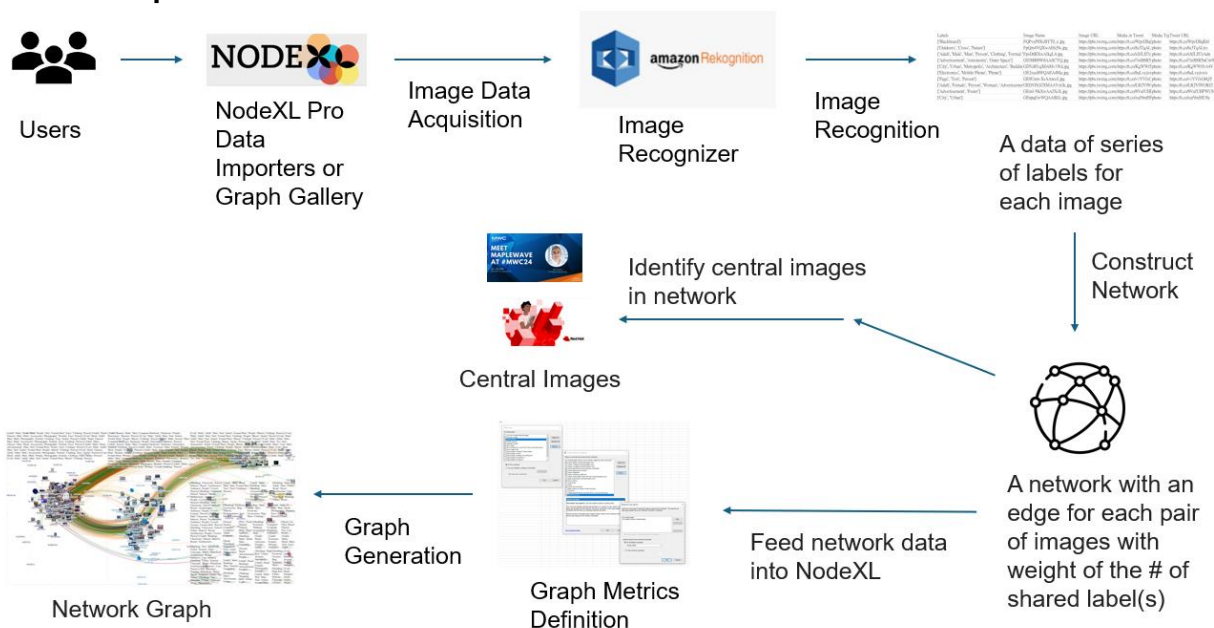


Figure 3. Flow diagram of the identification of central images and the creation of the network graph.⁶

Figure 3 illustrates the process implemented in our project, showcasing the identification of the central images within an image network and the creation of network graphs. Based on the image analysis and network analysis, we not only identify key images from social media data containing images, but also build the image network visualization within the NodeXL product. It showcases that the application of the project enables the identification of key images and influential themes within the image network. Utilizing image-based network analysis, users can derive insights from images, which are as crucial as text information for understanding social media dynamic.

STATEMENT OF WORK

- Chih-Han Yeh: network analysis and application at customer, model evaluation with AWS Rekognition, communication with client, report writing, poster creation, standup presentations
- Ching Han Chang: image-to-text analysis with Kosmos-2, BLIP-2, and Amazon Rekognition, model evaluation with Kosmos-2, BLIP-2, and Amazon Rekognition, communication with client, report writing, standup presentations
- Vanessa Garcia de Aquino: model evaluation with Kosmos-2 and BLIP-2 captions, communication with client and stakeholders, report writing, github preparation, standup presentations

REFERENCES

1. Smith, M.A., et al. (2009). *Analyzing (social media) networks with NodeXL*. Proceedings of the fourth international conference on Communities and technologies, 255-264. <https://dl.acm.org/doi/abs/10.1145/1556460.1556497>
2. Onielfa, C. et al. (2022). *Influence in Social Networks Through Visual Analysis of Image Memes*. Artificial Intelligence Research and Development. https://www.researchgate.net/publication/364397680_Influence_in_Social_Networks_Through_Visual_Analysis_of_Image_Memes
3. Jose, N. C. (2016). *Social Media Network Data Mining and Optimization*. Master's thesis, Louisiana State University and Agricultural and Mechanical College. https://repository.lsu.edu/cgi/viewcontent.cgi?article=4023&context=gradschool_theses
4. Image 1. Source: <https://huggingface.co/microsoft/kosmos-2-patch14-224/resolve/main/snowman.png>
5. Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, 311–318. IBM T. J. Watson Research Center, Yorktown Heights, NY 10598, USA. [PDF](#)
6. Image Sources: <https://nodexl.com/>. <https://docs.aws.amazon.com/rekognition/>.

APPENDIX

Regarding the application, in addition to the network analysis, our team also explores the potential uses of the labels. We try to find the most representative images with labels directly as it may take less amount of time to process. First, we assume that images with more commonly shared labels are more central in a network, and we use the frequency of labels to infer the most representative images. However, the challenge is that the common labels overly dominate our dataset, therefore, the result is skewed towards the common labels. It may potentially obscure more uniquely representative images, so we then adjust the scores for each image by considering the inverse frequency of each label, but the result is overly skewed to the minority. We also explore the strategy of normalizing the label occurrences with Min-Max scaling and excluding the labels above a certain occurrence threshold but both of the results skew to overly

common labels too. In conclusion, the explorations of labels may not be ready to be used for our skewed datasets. We decide to focus on network analysis to identify the representative images. The code and efforts can be found in 'Exploration of potential uses of labels' section in `network_analysis_on_nodexl_data.ipynb`.