Arizona State University

Investigating Influencer Endorsements: A Data-Driven Exploration

Jinhang Jiang, Bhavana Patil, Dhruv Tyagi, Jinghuan Li

Guided by: Dr. Victor Benjamin

Agenda

- Business Motivation
- Model Design & Implementation
  - Text Analysis
  - Social Network Analysis
- Case study
- Exploring Instagram
- Summary

Business Motivation

- Influencer marketing is increasingly important
- Adidas pursues this strategy with great success (e.g., Kanye West)
- But, so far the effort is largely ad-hoc or relies on much manual effort
- Current approaches fails to consider unstructured data in social media (text, networks, image, video)
- Thus, a decision support system would be a great of asset

Where did we start ?

A social news aggregation, web content rating, and discussion website

www.reddit.com

Why Reddit?

Reddit has over 430 million monthly active users. An increase of 30.3% from a year earlier.

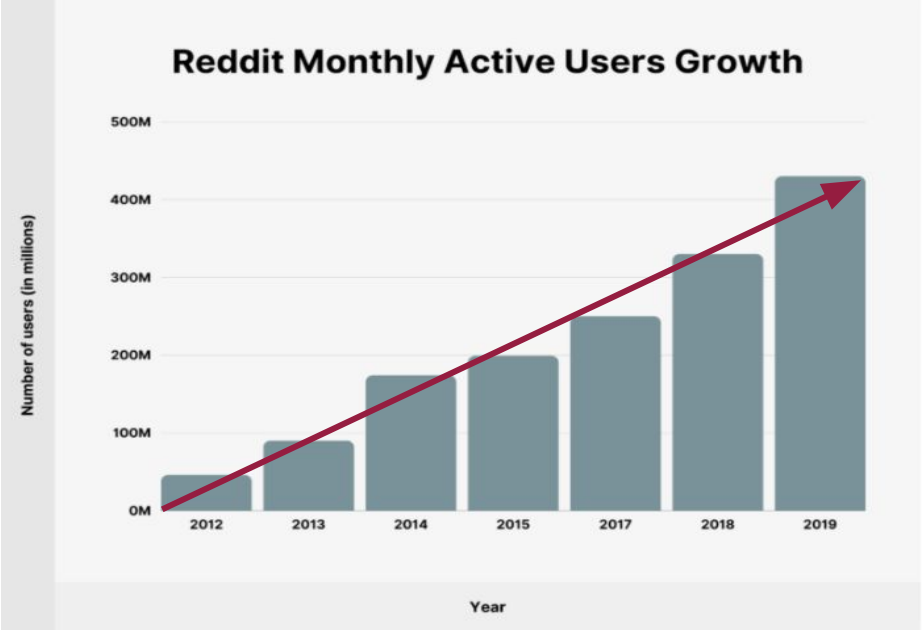
- **52 million daily active users** access Reddit.[6]
- **25% of US adults** use Reddit.
- In 2020, Reddit was ranked the **7th most popular social media app in the US**.
- **303.4 million posts** were shared on Reddit in 2020, a 52.4% year-over-year growth.
- **2 billion comments** were shared on Reddit in 2020.

It also allows for analysis of long-form social media instead of just short messages (twitter)

Source:https://backlinko.com/reddit-users

Reddit - a great source for data collection

- From 2015 to 2019, the number of Reddit monthly active users **has grown by 258.33%**. And it continues to grow in 2020 and 2021 ...[6]



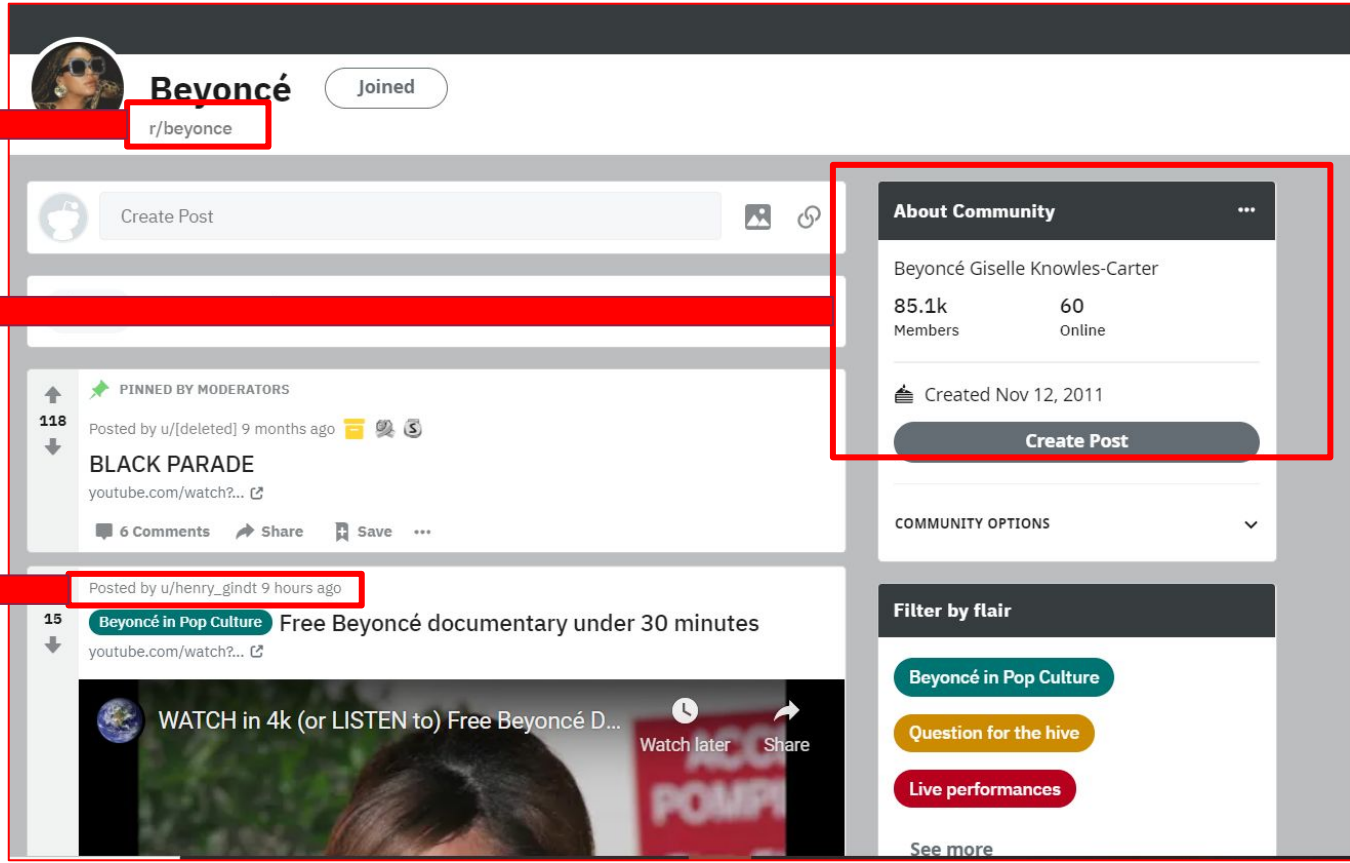
Source:https://backlinko.com/reddit-users

What is a subreddit?

Subreddit name

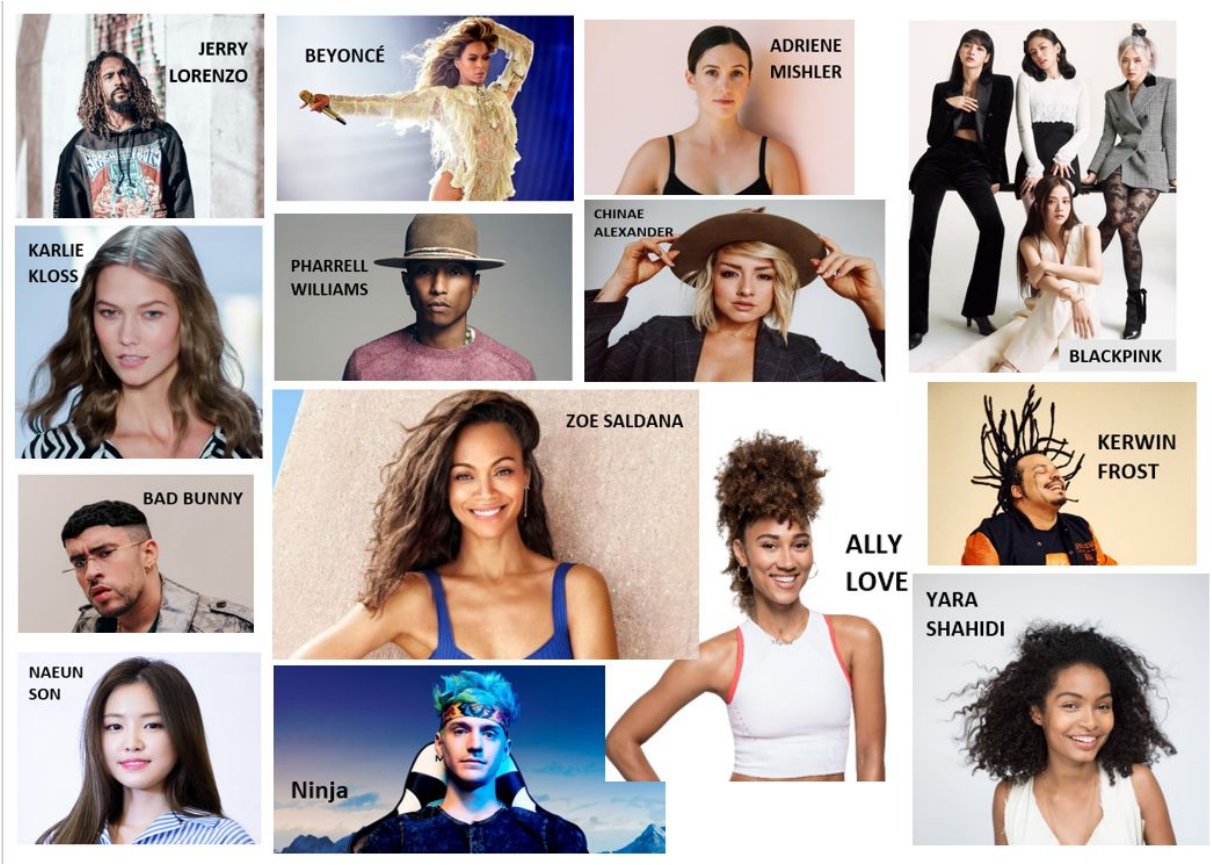
Information on the Subreddit

Information on the post



The influencers we started with:

Mr. Brooks Clemens gave us **a list of influencers who are sponsored by Adidas**.[5] We picked 14 of them for our research.



About the Data

- Types of data collected : ***user information***, ***text data***, ***image data***[5]
- Total posts collected :39K+ comments, which are summed to **22M words**
- The number of users : **62K+ (For network analysis)**
- Data was collected from their respective subreddit and based on **n-gram search**.

Influencer	MsgID	Timestamp	Author	ThreadID	ThreadTitle	MsgBody	ReplyTo
NinjasHyper	*'gnqnhbb'*	*'2021-02-17'*	*'KidCaker'*	*'co3y25'*	*'Stunning!...'*	*'Epic! He…'*	*'gnkvpte'*
...	...	...	...	...	...	...	...

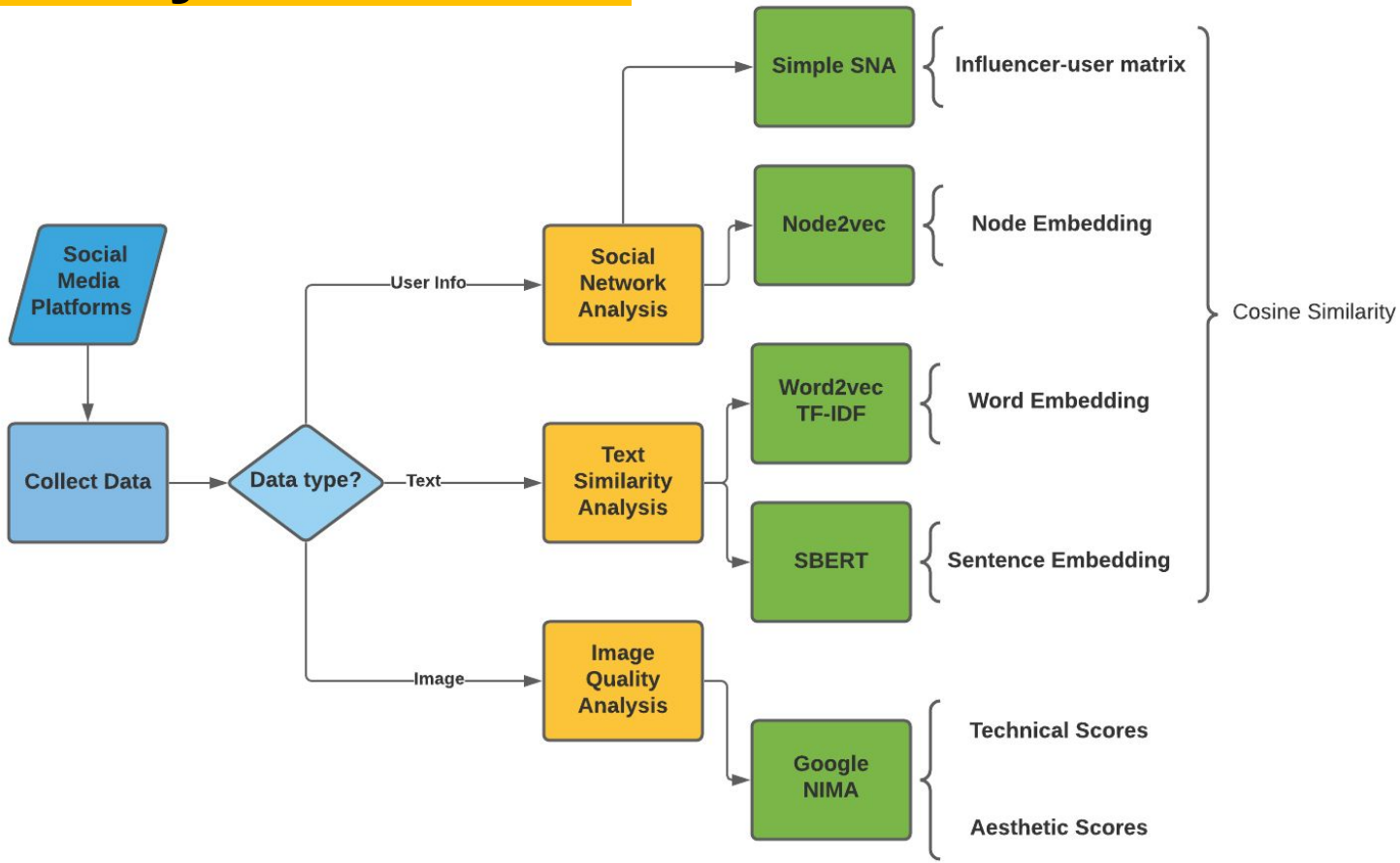
Collected Data

- Data is grouped into two versions for implementation testing and case study:
  - Version 1: 14 Celebrities is sponsored by Adidas
  - Version 2: Two adidas sponsored Korean celebrities + another eight picked Korean celebrities (based on stage style and popularity)

The design goal of the system is to:

- **Build a computational, data-driven, and generalizable system**
- **Support the business decisions with little human intervention**

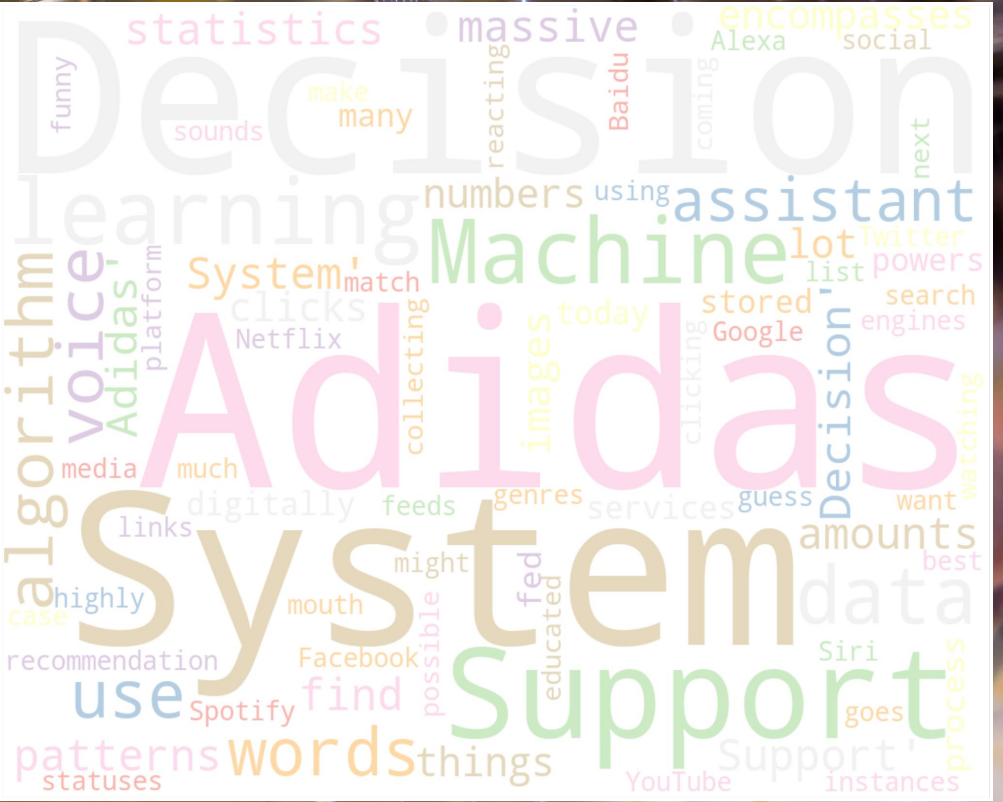
How the system works:



Why Text Analytics?

- **Text analytics** is the automated process of **translating large volumes of unstructured text into quantitative data to uncover insights**, trends, and patterns. [18]

- Combined with data visualization tools or other statistical models, this technique enables companies to understand the story behind the numbers (the transformed text) and make better decisions.[18]



How Text Analytics benefits business analysis?

We wanted the models to look at the conversations between celebrities' fans to capture different cultural patterns buried in the unstructured text data.

It helps our business to have a better understanding of:

- how similar the way fans/users talk about the influencers
- the type of customers each influencers can reach
- similarity and dissimilarity of language habits between the customer segments
- how the fans view the influencers

The Models in Our Decision Support System

We conducted research and experiments on a list of techniques that are popular in the academic literature and business applications, and we selected the following two models to our decision support system based on performance and computational speed.

01	Word2vec + TF-IDF[9]	• Word embeddings • Vectors of numeric representations of words • Self-trained model
02	Sentence_Transformers	• Sentence embeddings • BERT based, state of the art model • 10+ pre-trained models from billions of words

“Embedding” : A Task-specific Dictionary

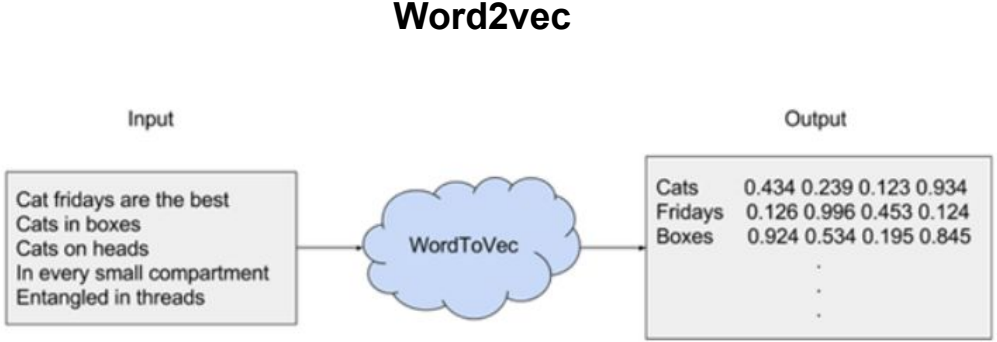
- In the context of machine learning, an **embedding is a low-dimensional, learned continuous vector representation of discrete variables** into which you can translate high-dimensional vectors.[8]

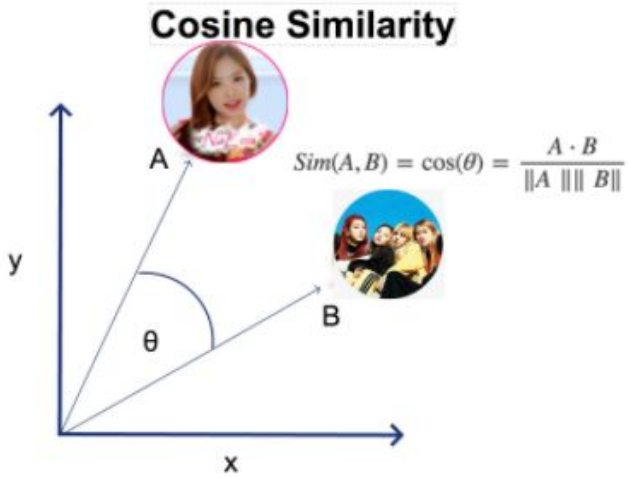
- Generally, embeddings make ML models more efficient and easier to work with, and can be used with other models as well.

- For example, when applied to text, we can learn the semantic meaning of a word by observing what other words it frequently appears next to. And then we can generate a list of embeddings, which can be seen as a task-specific dictionary.

Word2vec and Cosine Similarity

- Word2Vec is an algorithm that accepts text corpus as an input and outputs numerical representations for each word
- Cosine similarity is a metric used to measure how similar the documents are irrespective of their size.

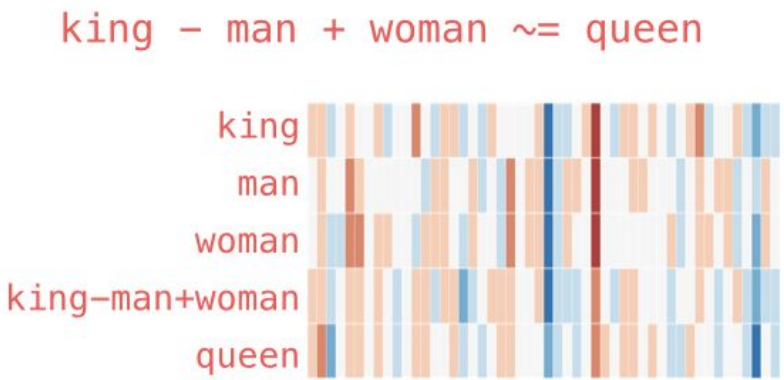




Source: https://jalammar.github.io/illustrated-word2vec/

The Dictionary Captures Semantic Meaning

A famous example of the concept of analogy:



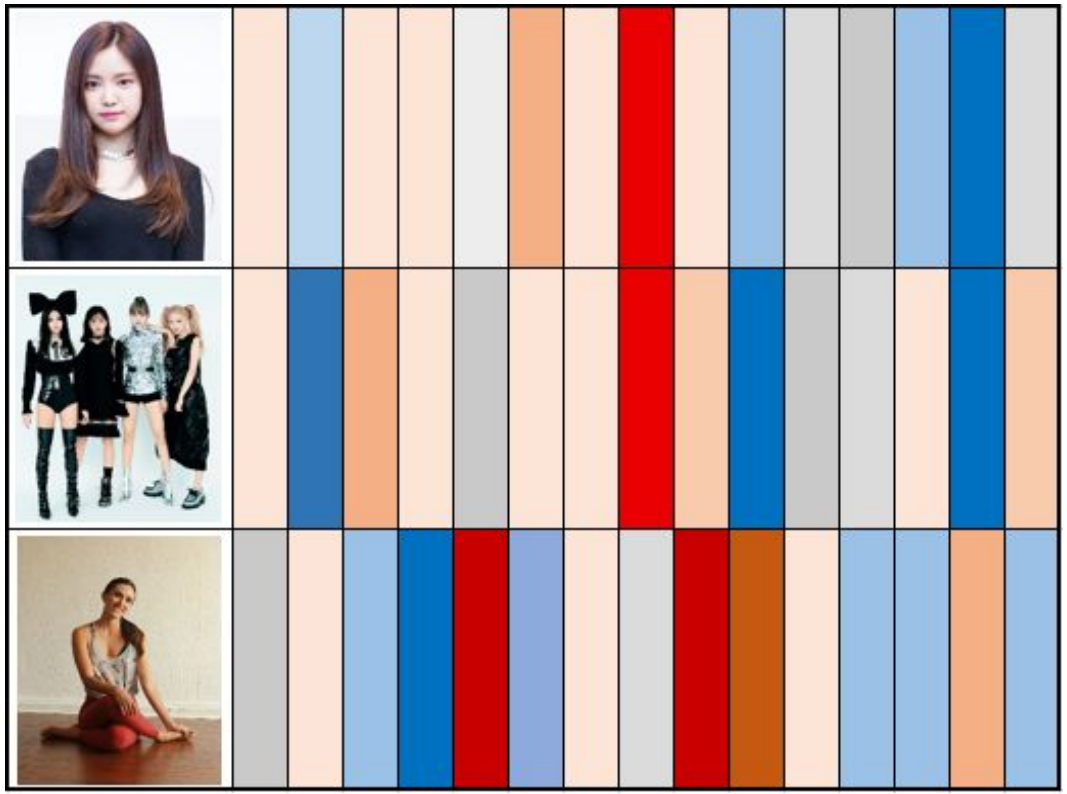
The resulting vector from "king-man+woman" doesn't exactly equal "queen", but "queen" is the closest word to it from the 400,000 word embeddings in this collection.[1]

Analogy for Our Case

We can clearly perceive that the users talked about Naeun Son and BlackPink in a more similar way.

We can use this information to find :

- overlaps among the users' culture
- specific groups of customers



Word2vec - How to Validate the Dictionary

- This is an unsupervised method, so we do not have a ground truth to evaluate if the model has captured the semantic meaning of the words
- We used the most similar words of a specific word *[“kpop”]* as the evaluation metrics

Top 3 of the most similar words to " *kpop* "	
Words	Similarity Score
"makestar"	0.783166170
"broadcasting"	0.725594878
"blackpink"	0.672631085

Word2vec - How to Validate the Dictionary

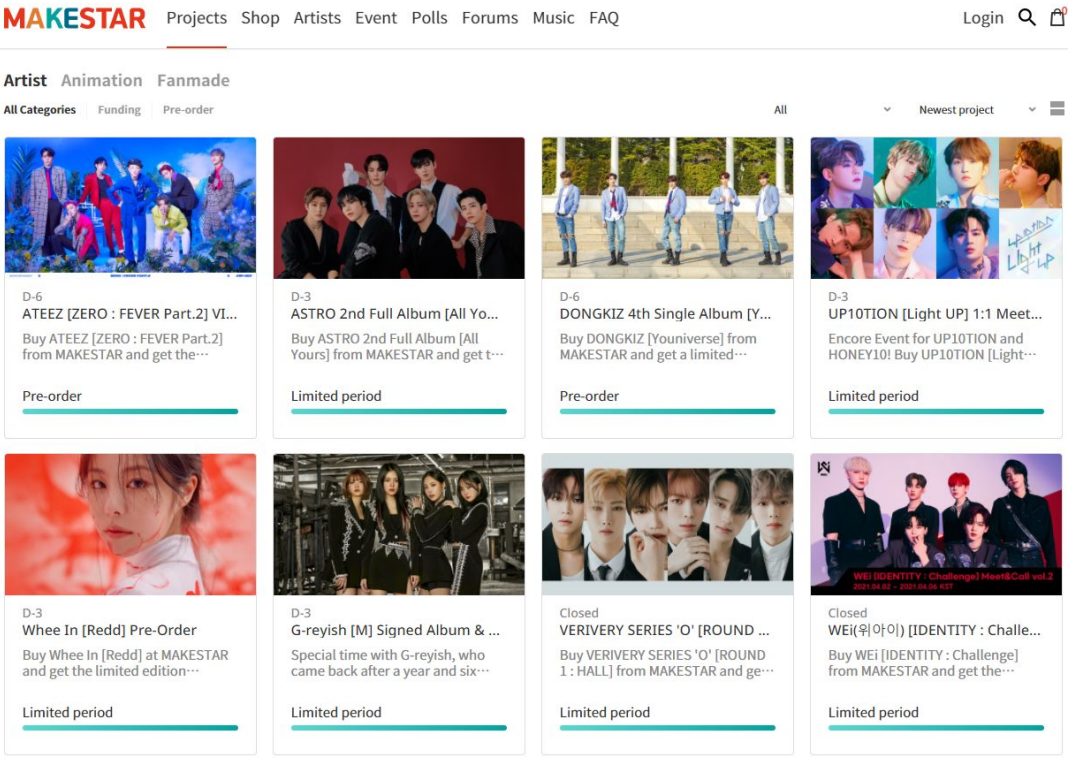
“Makestar”:

Makestar, which will present the projects that stars and fans make together, is a two-way communication window that helps their communication.

“Broadcasting”:

Music programs of South Korea are broadcast weekly, with different artists performing on the shows to promote their music.

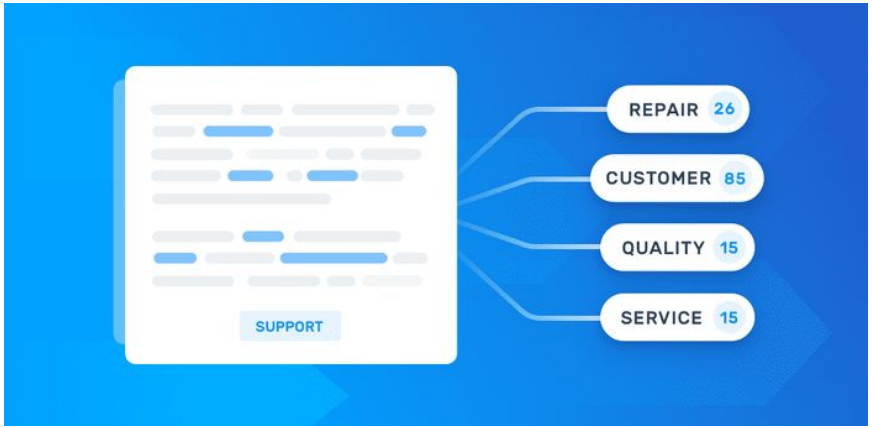
more examples...



How to Reflect the Importance and Relevance of a Word?

TF-IDF, is a numerical statistic that is intended ***to reflect how important a word is to a document in a collection.***

- Documents with similar, relevant words will have similar vectors
- It also helps discover the uniqueness of the language habits of the certain customer groups



Similarity - Wording Relevance and Semantic Meaning

Top 3 of the most similar influencers to " *Beyonce* "	
Influencers	Similarity Score
"Zoe Saldana"	0.869546415
"Pharrell Williams"	0.865572384
"BlackPink"	0.863266794

Top 3 of the most similar influencers to " *Pharrell Williams* "	
Influencers	Similarity Score
"Beyonce"	0.865572384
"Jerry Lorenzo"	0.856141943
"Bad Bunny"	0.834991535

SBERT with Sentence_Transformers

- BERT, machine learning technique for natural language processing pre-training developed by Google, published in 2018

- Sentence BERT (or SBERT) was developed by UKP Lab from Darmstadt University of Technology in Germany

- It provides an increasing number of state-of-the-art **pretrained models for more than 100 languages**, fine-tuned for various use-cases.

The List of Popular Text Mining Techniques

01	TF-IDF	• Term frequency–inverse document frequency • Reflects importance of a word to a document
02	Word2vec + TF-IDF[a]	• Word embeddings • Vectors of numeric representations of words • Self-trained model
03	GloVe + TF-IDF	• Word embeddings • Pre-trained model • 1M+ words
04	Sentence_Transformers	• Sentence embeddings • BERT based, state of the art model • 10+ pre-trained models from billions of words
05	Sent2vec	• Sentence embeddings • Good on small dataset (pre-trained models) • Crashed with large dataset due to RAM limits
06	Doc2vec	• Document embeddings • Vectors of numeric representations of docs • Self-trained model

Social Network Analysis (SNA)

Social network analysis is the process of **investigating social structures** through the use of networks and graph theory.

It characterizes networked structures in terms of nodes and the ties, edges, or links that connect them.

How SNA Benefits Business Analysis

In the private sector, businesses use SNA to support activities such as:

- customer interaction and analysis
- information system development analysis
- marketing and business intelligence needs

In our case, we performed SNA to reveal information about the influencer's fanbase activities, overlaps, and interactions

SNA with Influencer-User Matrix

- We created an influencer-user (I-U) matrix to compute cosine similarity scores
  - Find the number of occurrences of each users for each influencer
  - Compute the cosine similarity between the vectors of occurrences
  - Sorting the results and find the closest ones

	User #1	User #2	User #3	User #4	User #5
Jay	3	0	7	2	1
Influencer1	2	1	2	0	0
Influencer2	0	6	2	1	0

cosine_similarity(Jay [3 0 7 2 1] , Influencer1 [2 1 2 0 0]) = 0.84 ✓

cosine_similarity(Jay [3 0 7 2 1] , Influencer2 [0 6 2 1 0]) = 0.31

How I-U Matrix Tackles Business Questions

Beyonce

```
model.best_sub("Beyonce",n=2)
['Bad Bunny', 'Pharrell Williams']
---------------------------------
model.best_complete("Beyonce",n=2)
['Adriene Mishler', 'Jerry Lorenzo']
```

⇌

+

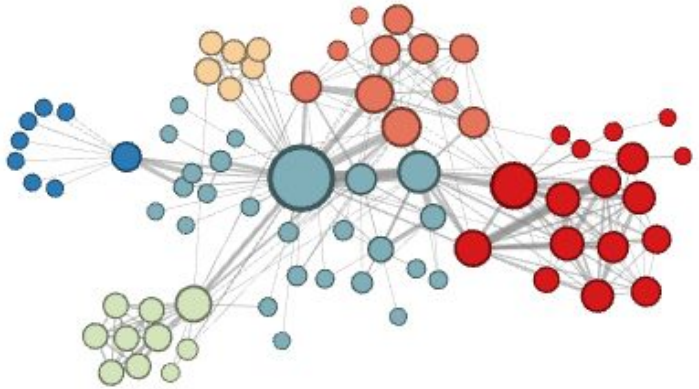
⇌

+

Naeun Son

```
model.best_sub("Naeun Son",n=2)
['BlackPink', 'Bad Bunny' ]
---------------------------------
model.best_complete("Naeun Son",n=2)
['Kerwin Frost', 'Zoe Saldana' ]
```

SNA with node2vec

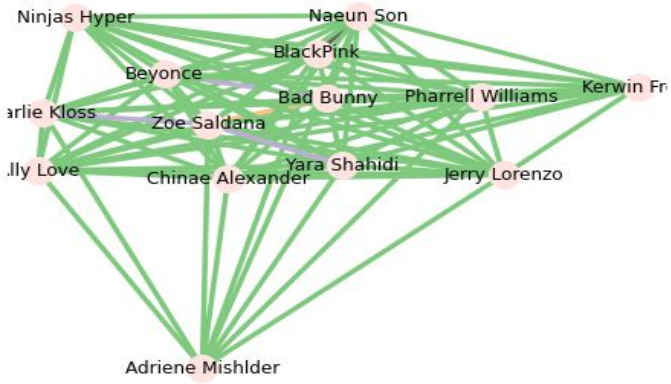


- Developed by Stanford
- Built upon the idea of word2vec
- Convert high dimensional data/information into a low-dimensional representations[2]
- By studying the relationship between the nodes, business can generate the insights of the people to whom the influencers reached
- A node embeddings (or influencer's fanbase dictionary) will be generated:
    - Can be used for various analysis, like K-means clustering

Node2vec - Generate an Node Dictionary

Data for SNA	
Celebrity (Nodes)	Usernames (Edges)
Beyonce	User01
….	….

Graph



Embedding

Nodes	Embeddings
Beyonce	-0.1315, 0.7745, ...
….	….
Naeun Son	0.8516, 0.1442, ...

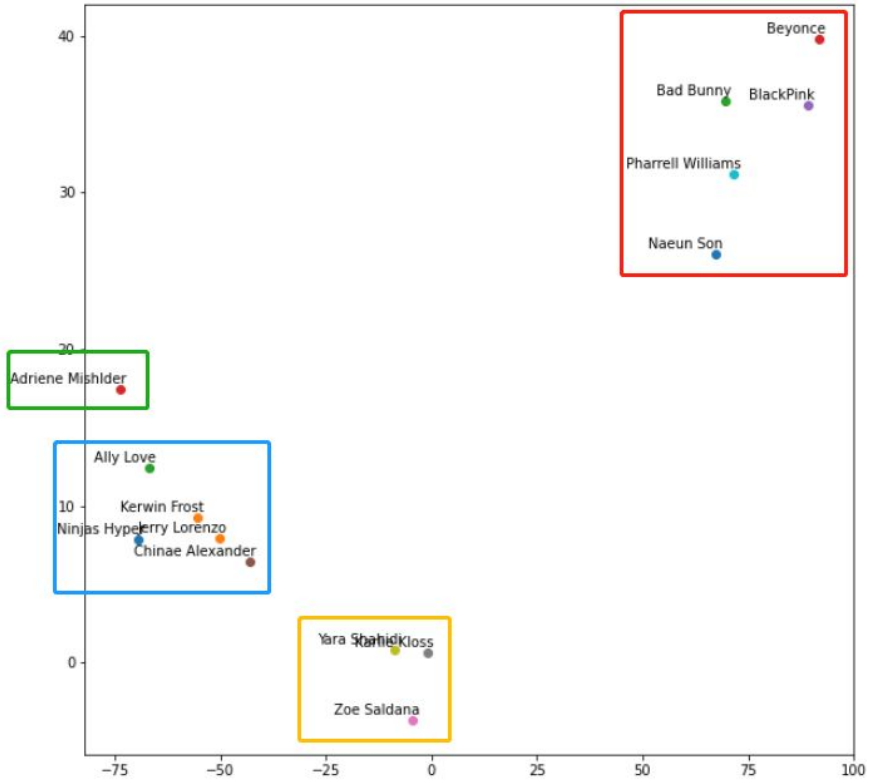
Usages

T-SNE Plot
Node2vec Built-in Feature
K-means Clustering

Influencers' Fanbase Segments

K-means Clustering with k = 4

Groups	Influencers
Group 1	Ninjas Hyper
	Jerry Lorenzo
	Kerwin Frost
	Ally Love
Group 2	Bad Bunny
	Beyonce
	BlackPink
	Pharrell Williams
	Naeun Son
Group 3	Adriene Mishler
Group 4	Chinae Alexander
	Zoe Saldana
	Karlie Kloss
	Yara Shahidi



CASE STUDY

Business Question

Due to a controversy, if we had to substitute Naeun Son and BlackPink with someone else, who would be the ideal candidate?

=

?

=

?

Case study - Background

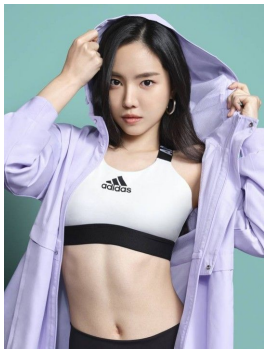
To better understand the international market, as well as to demonstrate the potential usage of the decision support system we built, we specifically picked 8 more korean groups/individuals (based on popularity, stage style, etc.) along with “Black Pink” and “Naeun Son” to conduct a case study.[11]

Groups	Ranking	Reddit Subscribers
Black Pink	#2	122k
IZ*ONE	#5	27.4k
BTS (boy group)	#1	214k
NCT (boy group)	#6	29.4k
GFriend	#9	23.8k

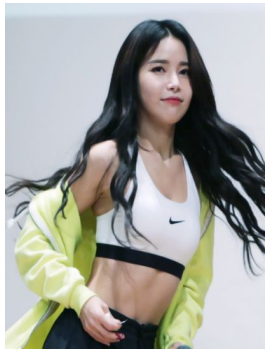
Individuals	Groups	Ranking	Reddit Subscribers
Naeun Son	Apink	#14	7.5k
Seolhyun	AOA	(Nike)	4k
Sana	TWICE	#2	102K
Solar	MAMAMOO	#7	18.5k
Miyeon	GI-DLE	#4	15k

Case study - Hypothesis

- Either Solar or Seolhyun could be the most similar individual to Naeun Son. Naeun and Seolhyun look very similar to each other and Naeun and Solar are both prominent members of a girl group.
- IZ*ONE could be the most similar girl group to BlackPink based on ranking (#5 & #2) and origin (Seoul, South Korea)
- Sana and Miyeon could be very unique in this datasets since their stage positionings and artistic style are very different from the other groups or individuals.[11]


Naeun

≈


Solar

≈


Seolhyun

Case study - Data

Data for Text Mining[5]:

- ★ # of comments: 23381
- ★ # of words: 15M+

Data for SNA[5]:

- ★ # of Influencers : 10
- ★ # of observations: 49274
- ★ # of users: 35147

Case study - Text Mining (word2vec+Tf-idf)

```
most_similar("BlackPink",
             Pairwise_similarities,
             'Cosine Similarity',
             topn=2)
```

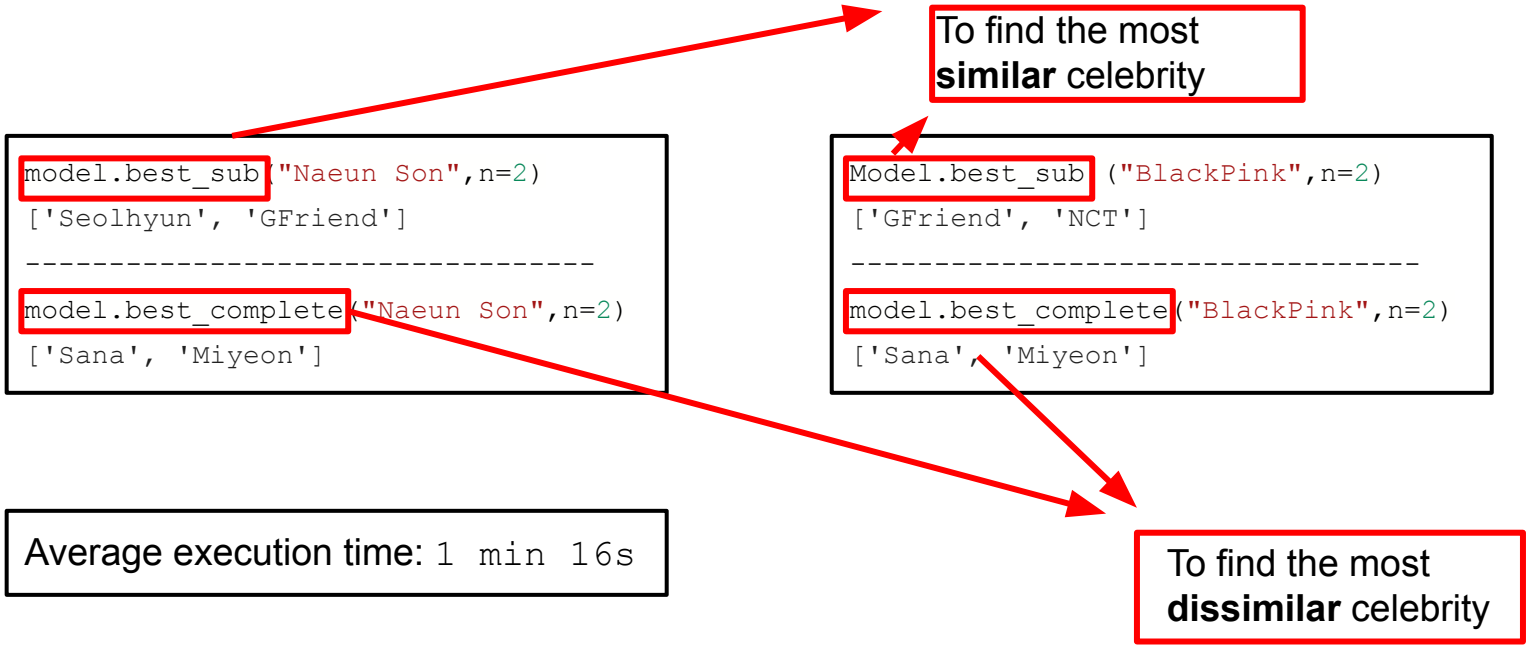
```
Name: BlackPink
Similar Influencers:
Celebrity: iZone : 0.7479952518415461
Celebrity: GFriend : 0.7131607796176859
```

```
most_similar("Naeun Son",
             Pairwise_similarities,
             'Cosine Similarity',
             topn=2)
```

```
Name: Naeun Son
Similar Influencers:
Celebrity: Seolhyun : 0.6698078801647593
Celebrity: Solar : 0.6178915677265608
```

"Followers of BlackPink talk 74.79% similar to followers of IZ*ONE on reddit "

Case study - Social Network Analysis (basic decision modeling)



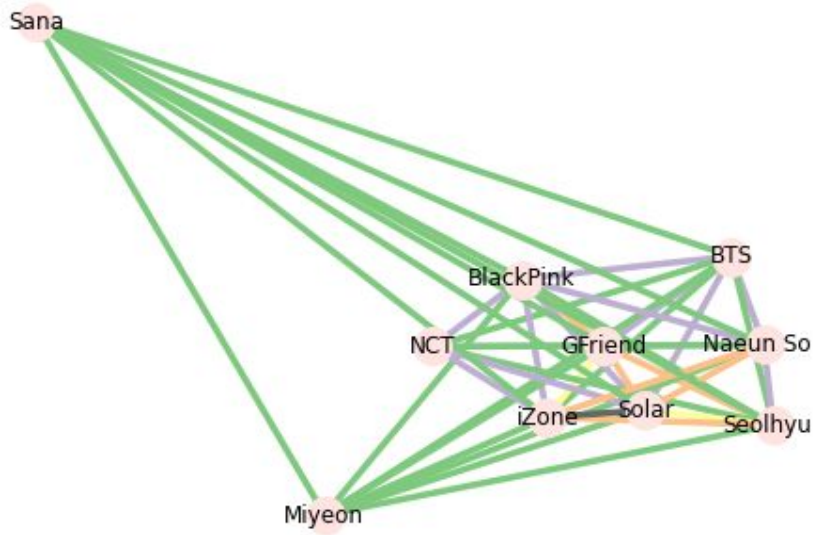
Case study - Social Network Analysis (node2vec)

```
Name: Case Study for Korean Influencers
Type: Graph
Number of nodes: 10
Number of edges: 43
Average degree:   8.6000
```

Average execution time: `188 ms`

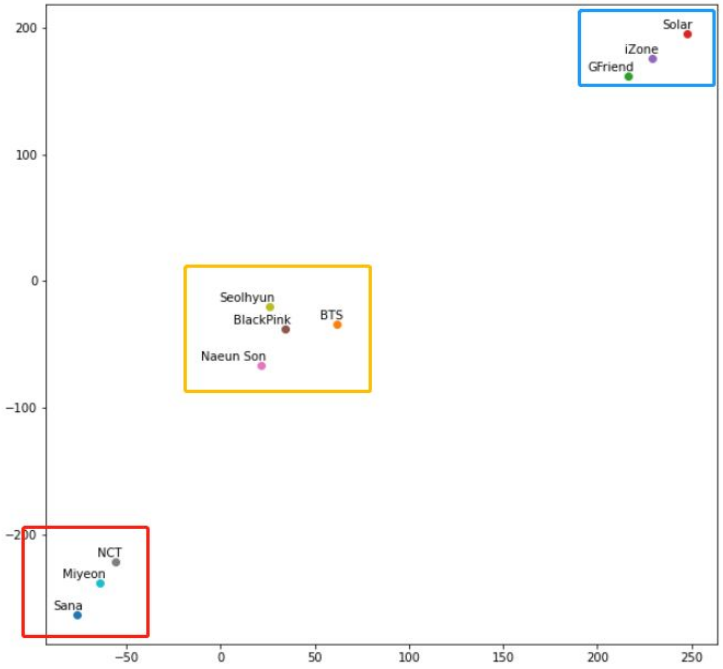
---

node2vec saved ***97.52%*** of execution time



Case study - node2vec

T-SNE Plot



node2vec built-in feature

```
model.wv.most_similar("Naeun Son",topn=4)
```

(the groups were filtered out)

```
[('Seolhyun', 0.9898267984390259),
 ('Solar', 0.9860391616821289),
 ('Miyeon', 0.9639727878570557),
 ('Sana', 0.9561481475830078)]
```

```
model.wv.most_similar("BlackPink",topn=4)
```

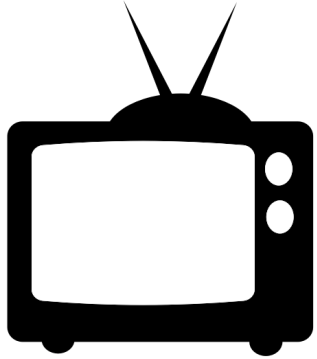
(the individuals were filtered out)

```
[('iZone', 0.996224582195282),
 ('GFriend', 0.9955752491950989),
 ('BTS', 0.9739093589782715),
 ('NCT', 0.9657029318809509),]
```

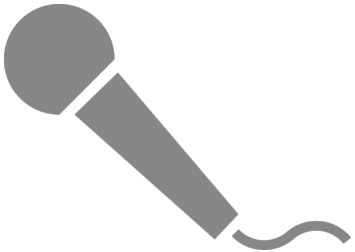
Conclusion

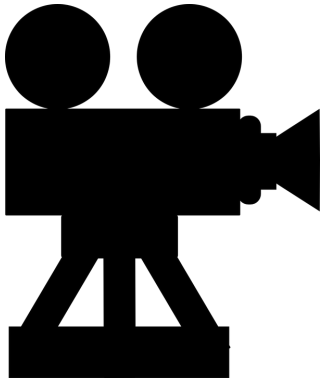
LIMITATION & FUTURE WORK

Data



1. Different backgrounds.
2. Not all of them on Reddit .
3. 22% of users aged from 18 to 29 (as of February 2019)[6]
4. We observed that the functionality Reddit API is not constant when we try to extract large amounts of data at once.





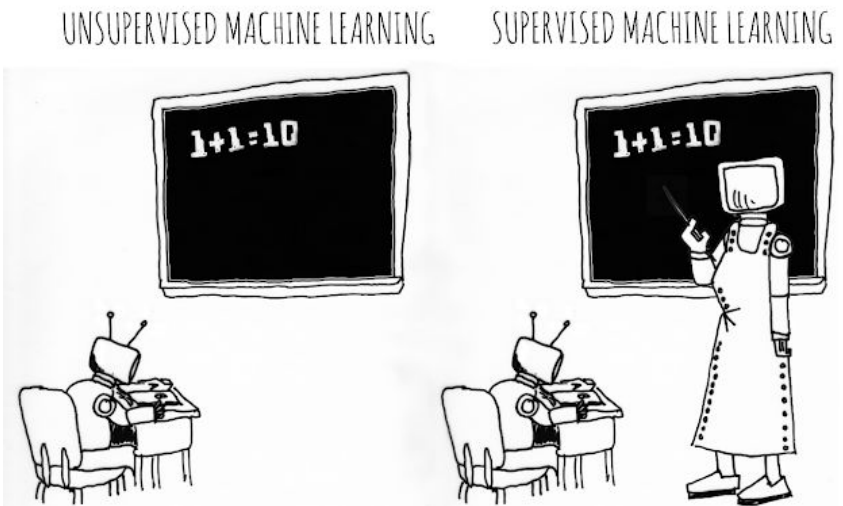
Text Mining

Limitations:

- No ground truth that can help measure the performance of the models
- Better performing models need labeled data

Potential Future Work:

- Additional preprocessing of data & vetting of the processed data
- Move on to supervised models



Limitations on the SNA

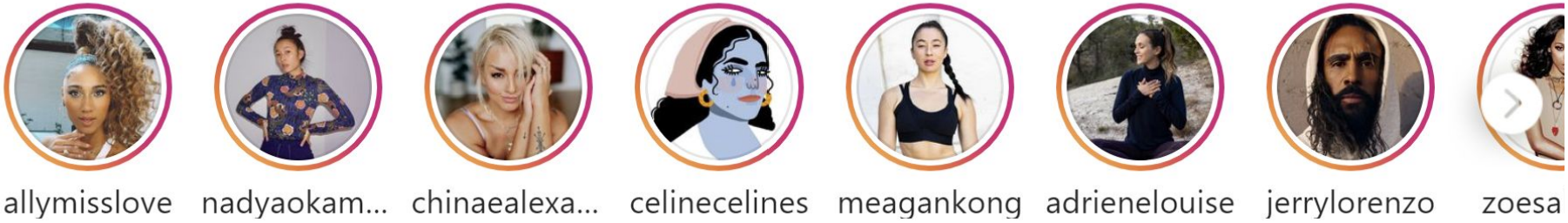
- A small number of nodes will limit the performance (e.g. we only had 14 influencers in the dataset)
- The data is unbalanced, so the model still could be biased.

Potential Future Work of SNA

- Expand the influencer list to conduct a larger scope of case studies
- Collect data periodically (e.g. every 3 month) to study seasonal patterns and capture the fandom changes over time
- Node2vec can be applied for online product recommendations

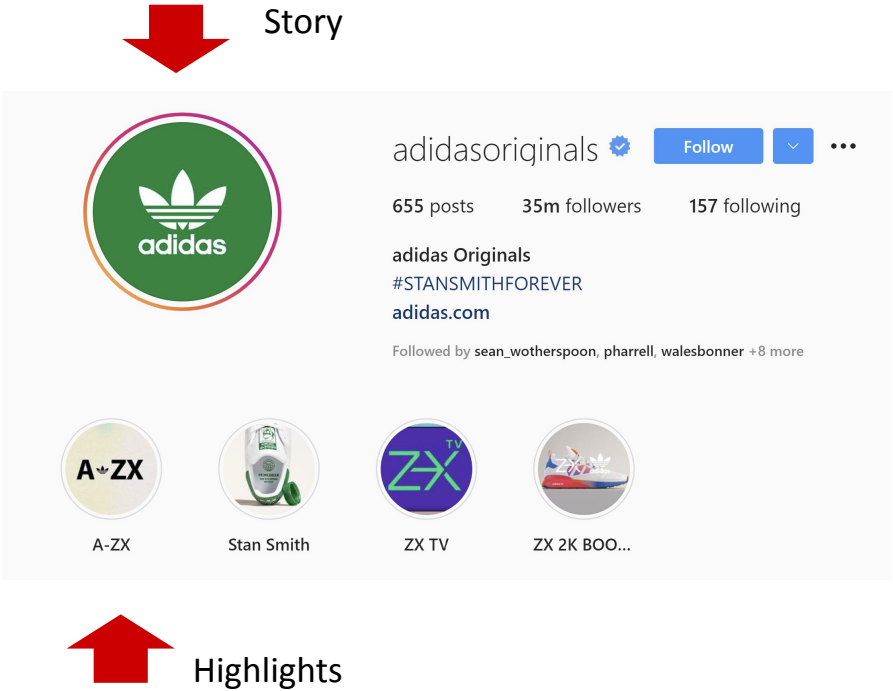
Arizona State University

Possibilities on Analyzing Instagram Stories



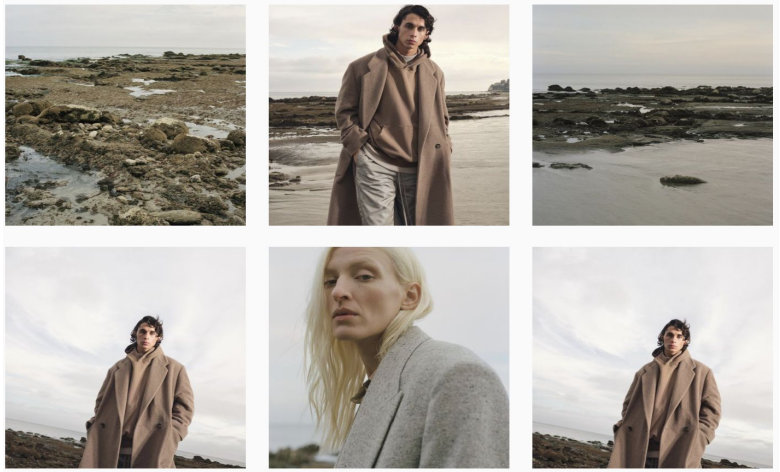
What are instagram Stories?

- "Share all the moments of your day, not just the ones you want to keep on your profile"
- Ephemeral: expires 24 hours after posting
- Can be permanent if user add it as a highlight

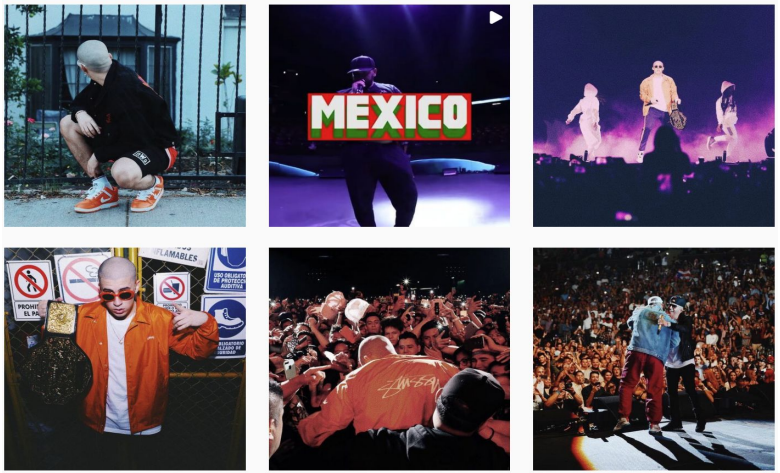


Business Value of Understanding Stories

- Understand celebrities' different aesthetics (visual feeling)
- Avoid style overlap when marketing one product
- Tailor product endorsement matchup with celebrities' show style



Jerry Lorenzo



Bad Bunny

Problems to Solve

1. **Computationally assessing image quality**

How effectively they convey the visual idea?

2. **Classifying images by style**

How do they convey the visual idea uniquely?

Getting Instagram Stories

Open source API - Instaloader:

- Python Module, Ready to use
- Full capability of downloading media along with its metadata

Local output:

- Media (jpg for image, mp4 and jpg thumbnail for videos)
- Metadata (zipped json file)

https://instaloader.github.io/index.html
https://instaloader.github.io/as-module.html#module-instaloader

Limitations on Getting Instagram Stories

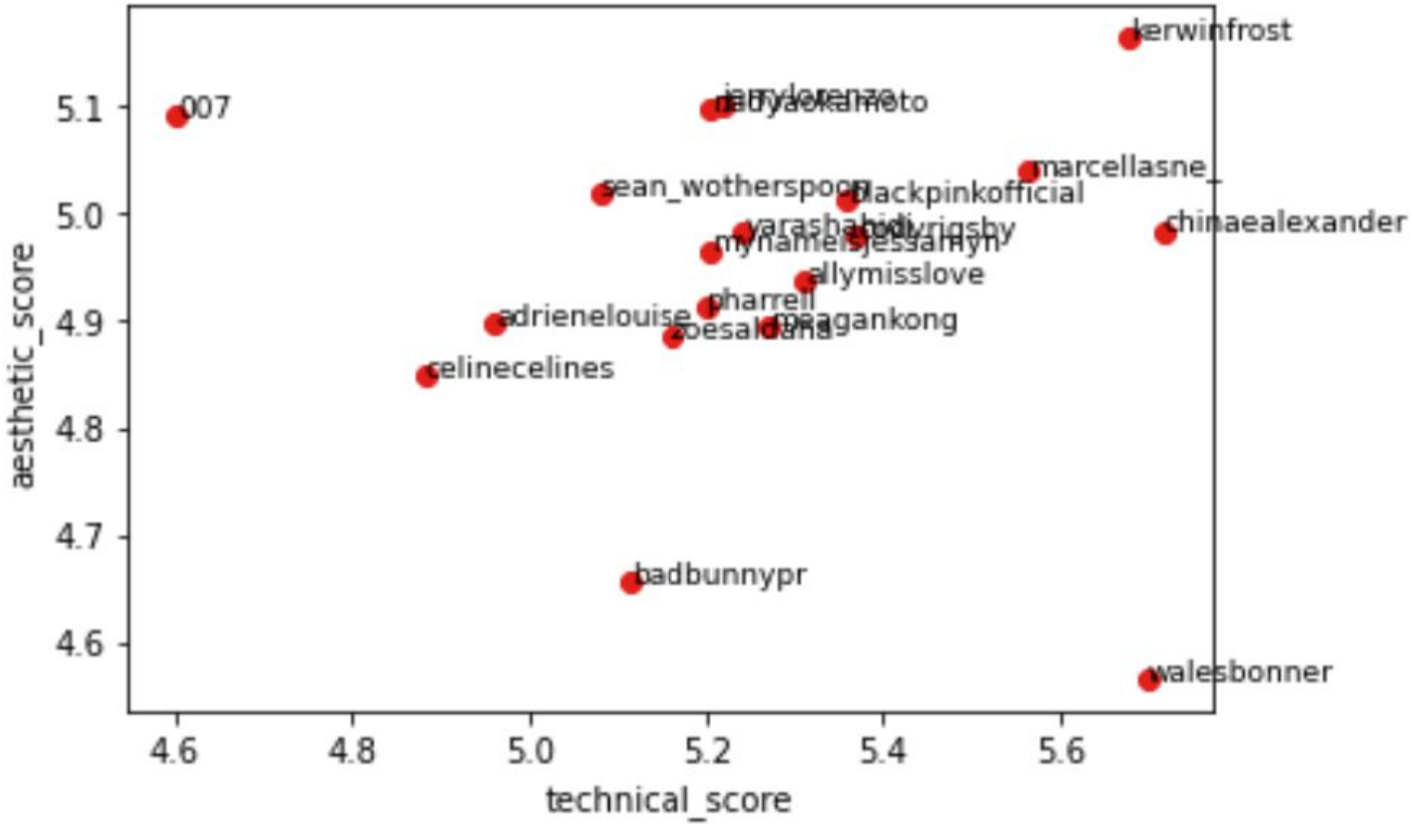
- Need to create an instagram account and follow all target users
- No way to catch expired stories
  - Need to run the script on 24-hour intervals to get all stories
- 429 Too Many Requests
  - Sometimes throw this error at around 200 fetches
  - Need multiple accounts to scrape the data

Google NIMA

- A Neural Image Assessment model
- Quantification of image quality and aesthetics
  - Technical Score: Looking good? Noise, blur, compression artifacts...
  - Aesthetic Score: Attractive? Beautifulness, human perception of good feeling.
- Trained with histograms of human ratings for every image
- Produces a distribution of ratings for any given image — on a scale of 1 to 10, assigning likelihoods to each of the possible scores
- Calculates a mean score correlates closely with human perception.

https://ai.googleblog.com/2017/12/introducing-nima-neural-image-assessment.html
https://idealo.github.io/image-quality-assessment/

Assessing Instagram Stories Quality



Analyzing Images with Google NIMA

Technical: 5.3163

Aesthetic: 5.4846

Wales Bonner

Karlie Kloss

Technical: 4.3772

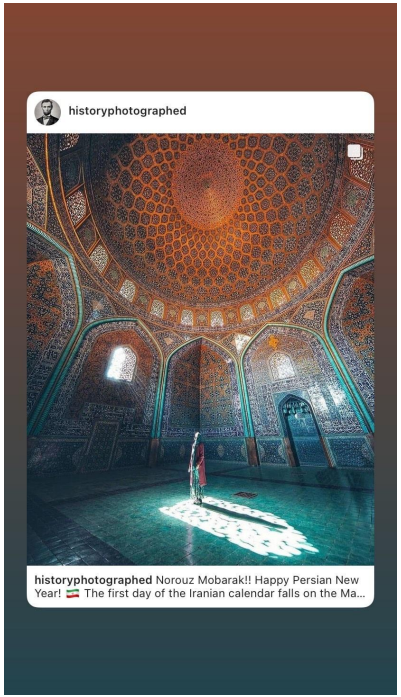
Aesthetic: 4.3204

- Blurry
- Text on top
- Color Fades out
- Color composition not ideal

Analyzing Images with Google NIMA

Technical: 4.2074

Aesthetic: 6.0637



Technical: 5.5548

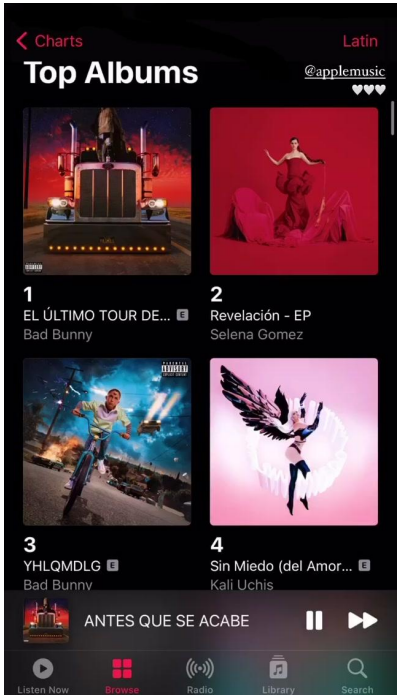
Aesthetic: 4.0486

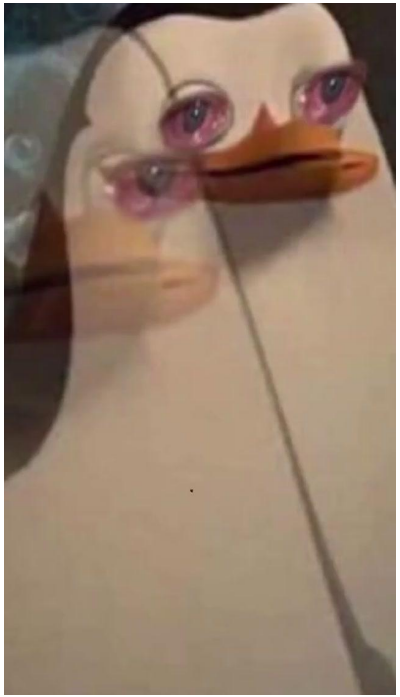
Adriene Mishler

Analyzing Images with Google NIMA

Technical: 4.5234

Aesthetic: 5.9242





Technical: 4.2735

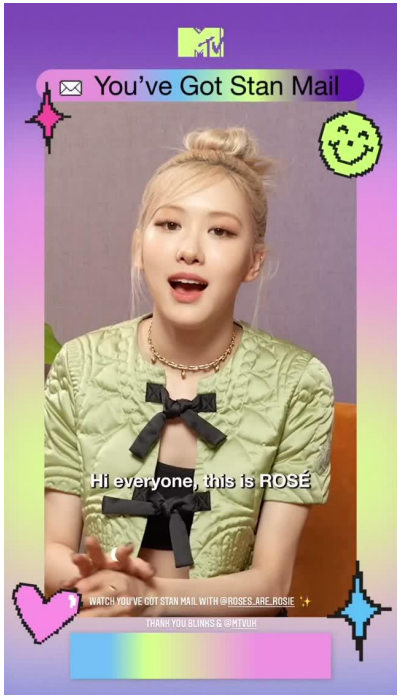
Aesthetic: 3.2516

Bad Bunny

Analyzing Images with Google NIMA

Technical: 6.2090

Aesthetic: 5.6127



Technical: 4.6019

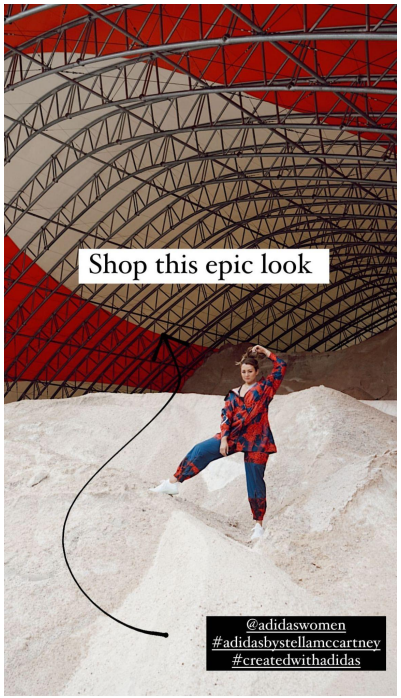
Aesthetic: 4.3183

Black Pink

Analyzing Images with Google NIMA

Technical: 4.9158

Aesthetic: 5.9319



Technical: 5.1978

Aesthetic: 3.6918

Chinae Alexander

Classifying Images by Style

- Vislab by Sergey Karayev - a set of Python modules for visual recognition
  - Features: based on color dependency, composition, visual attention…
  - Architecture: Fine-tuned Caffe Deep convolutional Framework
  - Dataset: 80K flickr images with 20 style labels
  - Style labels based on Optical techniques, Atmosphere, Mood, Composition styles, Color and Genre.
  - Scalability:
    - Crossing-platform: train from flicker and test on pinterest, AVA
    - Add own training set with labels

https://sergeykarayev.com/files/1311.3715v3.pdf
https://github.com/sergeyk/vislab
http://vislab.berkeleyvision.org/

20 Style Labels

Macro

Bokeh

Depth-of-Field

Long Exposure

HDR

Hazy

Sunny

Serene

Melancholy

Ethereal

20 Style Labels

Minimal

Geometric

Detailed

Texture

Romantic

Pastel

Bright

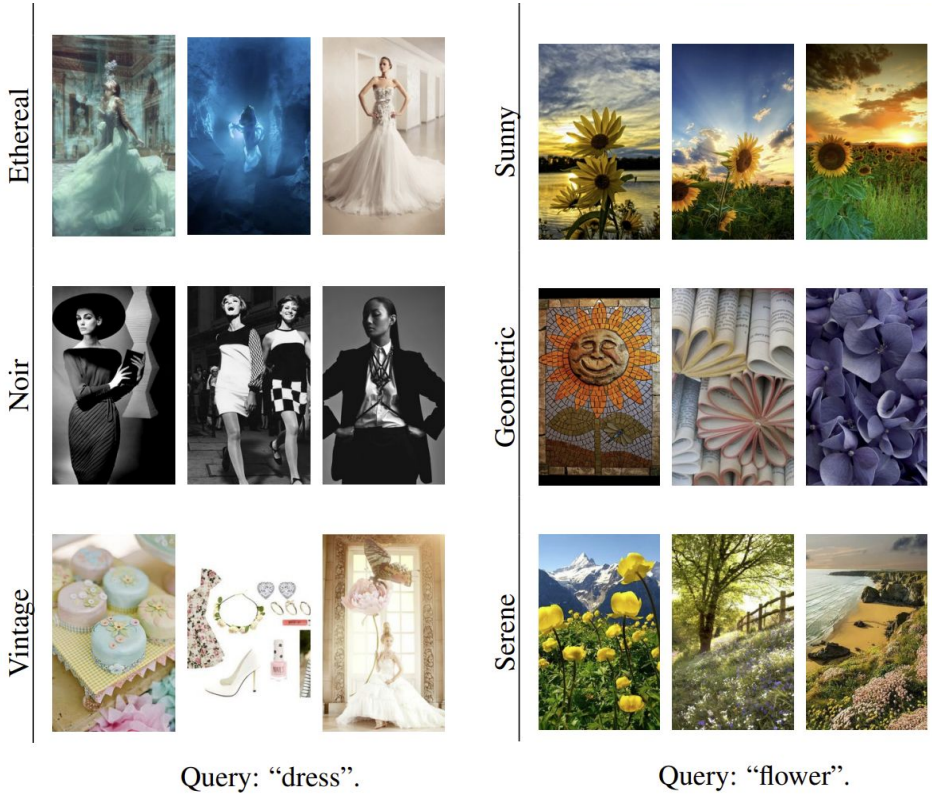
Noir

Vintage

Horror

Classifying Stories by Style

- We can get all images of same style.
- Combine with query of celebrities to explore each one's dominant visual style.
- Helps making marketing strategies
  - Minimal overlap
  - Target specific groups



Limitations of Image Style Classification

- Pre-trained model with fixed labels
- For better customized performance, need to create own dataset and manually label data, then retrain the classifier.

Thank you!

REFERENCE

Reference

1. Alammar, J. (2019, March 27). *The Illustrated Word2vec.* Jay Alammar's Blog. https://jalammar.github.io/illustrated-word2vec/.
2. Cohen, E. (2018, April 23). *node2vec: Embeddings for Graph Data.* Medium. https://towardsdatascience.com/node2vec-embeddings-for-graph-data-32a866340fef.
3. Graf, A., & Koch-Kramer, A. (n.d.). *Instaloader*. https://instaloader.github.io/.
4. Grover, A., & Leskovec, J. (2016, July 3). *node2vec: Scalable Feature Learning for Networks.* arXiv.org. https://arxiv.org/abs/1607.00653.
5. Jiang, J. (2021, March 23). *Use R to Calculate Boilerplate for Accounting Analysis.* Medium. https://towardsdatascience.com/use-r-to-calculate-boilerplate-for-accounting-analysis-f4a5b64e9b0d.
6. Jiang, J. (n.d.). *ASU_Adidas_Applied_Project_2021.* GitHub. https://github.com/jinhangjiang/ASU_Adidas_CapstoneProject.
7. Karayev, S.(n.d.) *vislab.* Github. https://github.com/sergeyk/vislab
8. Karayev, S., Hertzmann, A., Trentacoste, M., Han, H., Winnemoeller, H., Agarwala, A., & Darrell, T. (2014). *Recognizing Image Style*. Proceedings of the British Machine Vision Conference 2014. https://doi.org/10.5244/c.28.122
9. Lennan, C. (2018, July 9). *Using Deep Learning to automatically rank millions of hotel images.* Medium. https://medium.com/idealo-tech-blog/using-deep-learning-to-automatically-rank-millions-of-hotel-images-c7e2d2e5cae2
10. Lennan, C., Tran, D., et al. (n.d.) *image-quality-assessment.* Github. https://github.com/idealo/image-quality-assessment/
11. *Reddit Usage and Growth Statistics: How Many People Use Reddit in 2021?* Backlinko. (2021, February 25). https://backlinko.com/reddit-users.
12. Reimers, N., & Gurevych, I. (2019, August 27). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.* arXiv.org. https://arxiv.org/abs/1908.10084/.
13. sdaylorsdaylor. (2019, March 1). *What does embedding mean in machine learning?* Data Science Stack Exchange. https://datascience.stackexchange.com/questions/53995/what-does-embedding-mean-in-machine-learning.
14. Sieg, A. (2019, November 13). *Text Similarities : Estimate the degree of similarity between two texts.* https://medium.com/@adriensieg/text-similarities-da019229c894.
15. Talebi, H., & Milanfar, P. (2018, April 26). *NIMA: Neural Image Assessment*. IEEE Transactions on Image Processing, 27(8), 3998–4011. https://doi.org/10.1109/tip.2018.2831899
16. TensorFlow Team, G. (2017, November 20). *Introducing TensorFlow Feature Columns.* Google Developers Blog. https://developers.googleblog.com/2017/11/introducing-tensorflow-feature-columns.html.
17. [UPDATE] *January 2021 Top 50 KPOP Popularity Ranking.* KPOP OFFICIAL. (2021, February 1). https://kpopofficial.com/top-50-kpop-popularity-reputation-ranking-january-2021/#3_Kpop_Idol_Group_Popularity_Brand_Reputation_Ranking_All_Kpop_Groups.
18. What Is Text Analytics? MonkeyLearn Blog. (2019, November 20). https://monkeylearn.com/blog/what-is-text-analytics/.

APPENDIX

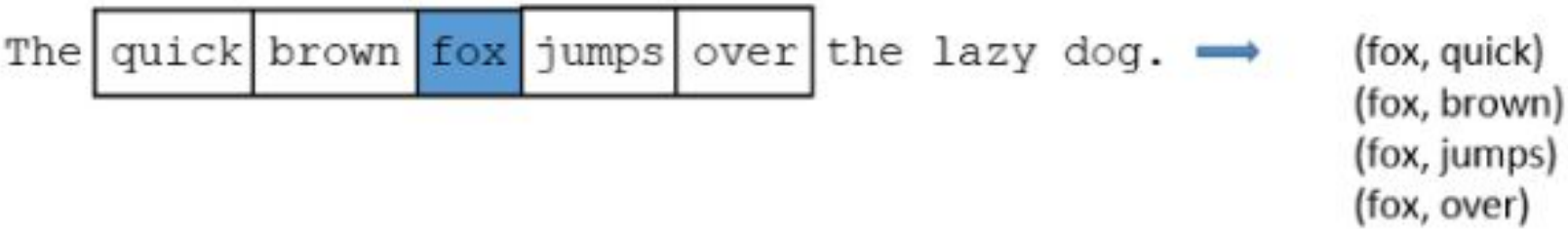
Word2vec - Skipgram

The pink boxes are in different shades because this sliding window actually creates four separate samples in our training dataset:

Jay was hit by a red bus in...

by	a	red	bus	in

input	output
red	by
red	a
red	bus
red	in



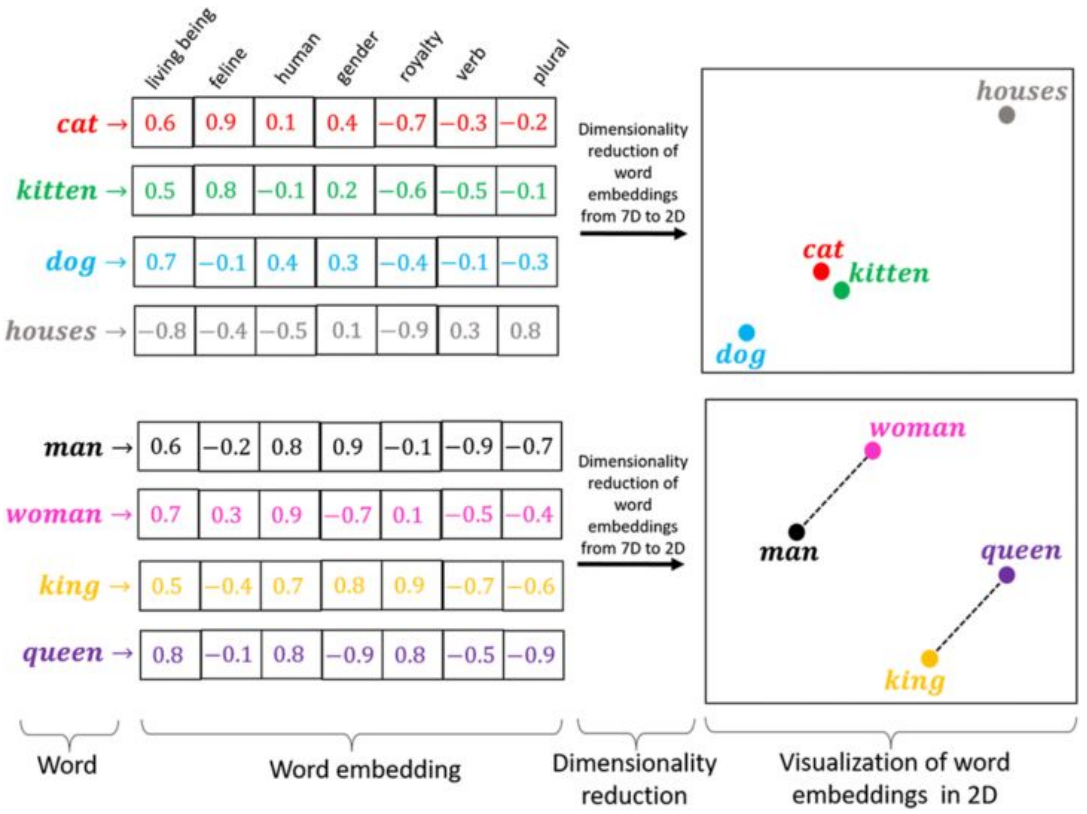
Source: https://jalammar.github.io/illustrated-word2vec/

Word2vec

It did twice dimension reduction during the process:

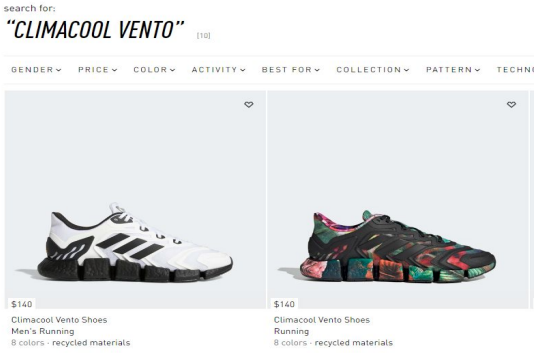
a corpus of words → a list of unique words with vector representations

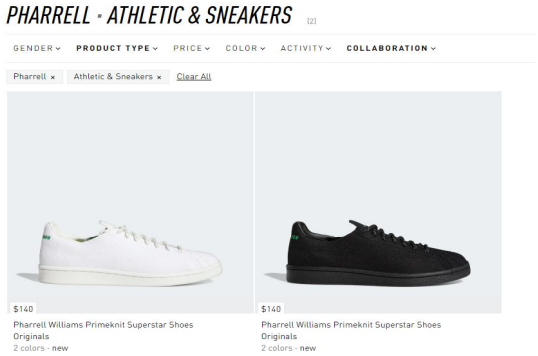
vectors → 2D embeddings



Word2vec - Evaluation Metrics

Top 3 of the most similar words to " *adidas* "	
Words	**Similarity Score**
"yzy"	0.754484594
"pharrellwilliams"	0.694894612
"climacool"	0.676122606





How the TF-IDF Calculated

We have the following four documents:

- D1 "The sky is blue"
- D2 "The sun is bright today"
- D3 "The sun in the sky is bright"
- D4 "We can see the shining sun and the bright sun"

After removing stop-words, the bag of words table is as follows:

	blue	bright	can	see	shining	sky	sun	today
D1	1	0	0	0	0	1	0	0
D2	0	1	0	0	0	0	1	1
D3	0	1	0	0	0	1	1	0
D4	0	1	1	1	1	0	2	0

$$\mathrm{IDF}(t) = 1 + \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing } t}\right)$$

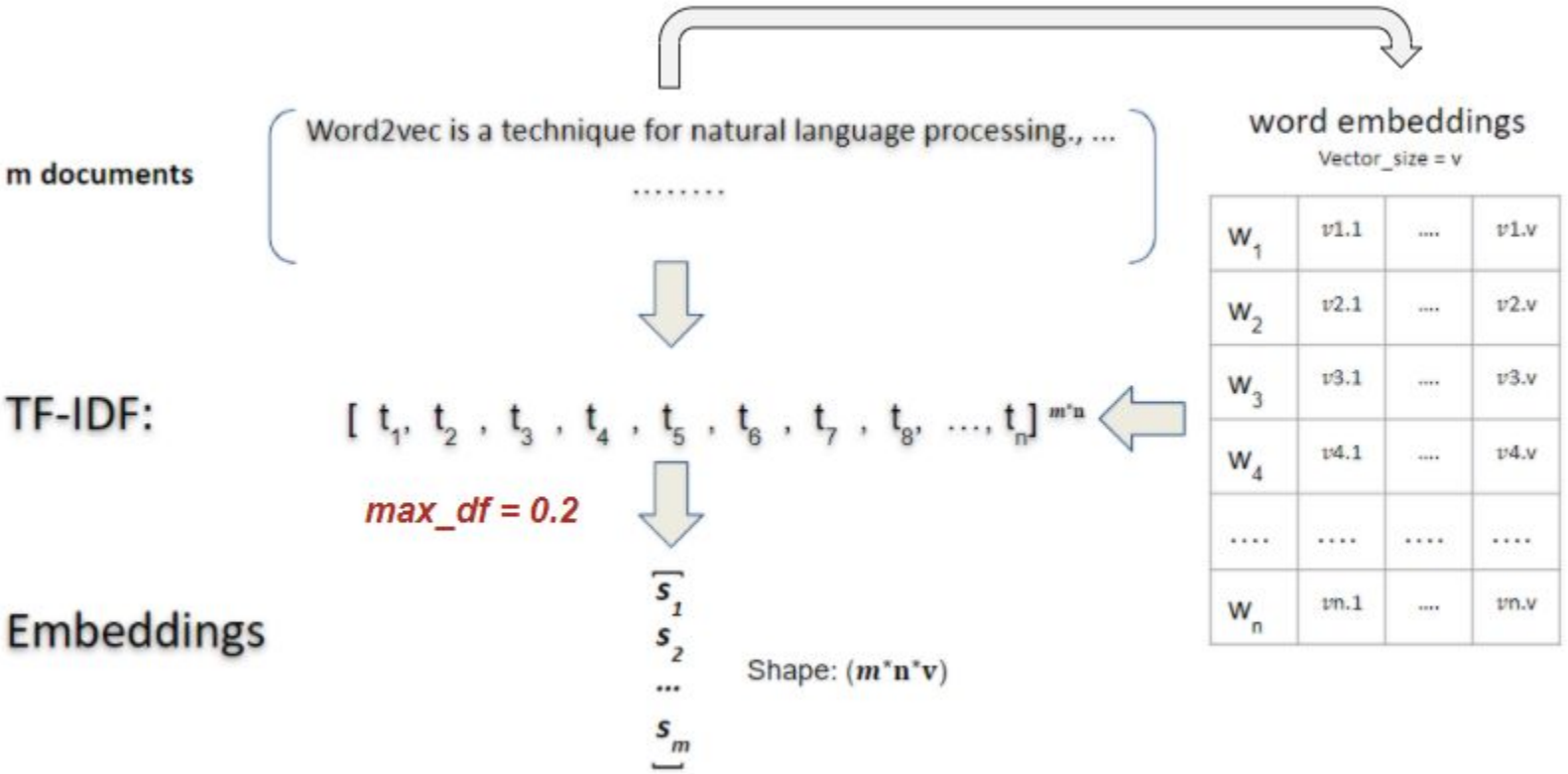
$$\mathrm{TF}(t, d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in the document } d}$$

$$\mathrm{TFIDF}(t, d) = \mathrm{TF}(t, d) \times \mathrm{IDF}(t)$$

How the model works

1. Treat 14 celebrities as 14 documents
   - *m = 14*
2. Top 30% of unique words based on frequency across all the documents[9]
   - *the number of unique words = 100,000*
   - *n = 100,000*0.3 = 30,000*
3. Based on a paper released by Google, the vector size of word2vec embeddings can be determined by the following formula:[10]
   - *v = 10M**0.25 ≈ 57*
4. The shape of the final embeddings is: ***(14,30000,57)***

How the model works



SBERT Implementation

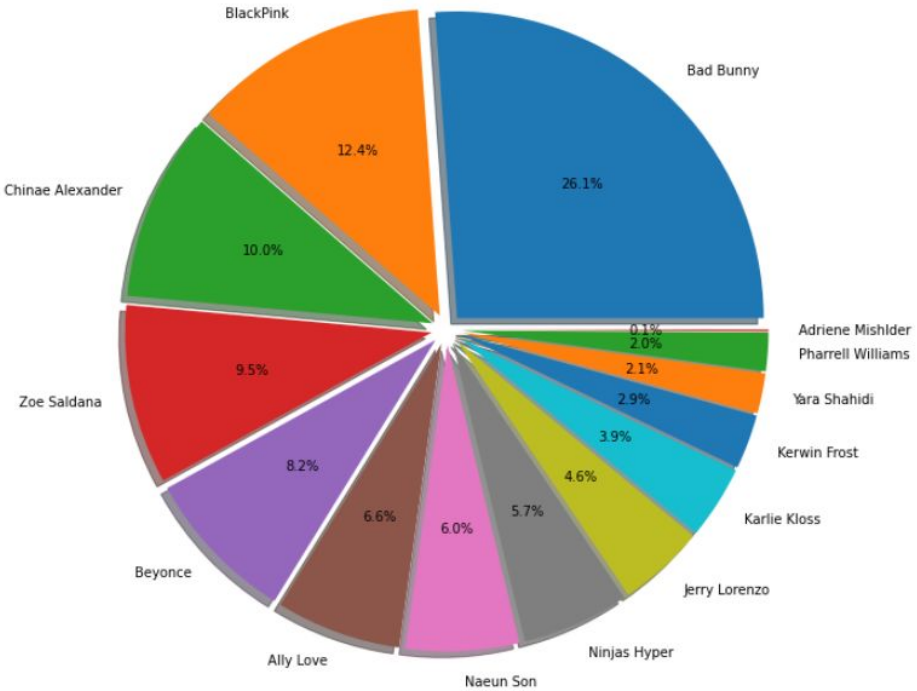
One of the best available models for Semantic Textual Similarity (STS) is:

stsb-bert-large - STSb performance

Top 2 of the most similar influencers to *"BlackPink"*	
Influencers	**Similarity Score**
"Naeun Son"	0.690971255
"Karlie Kloss"	0.671959343

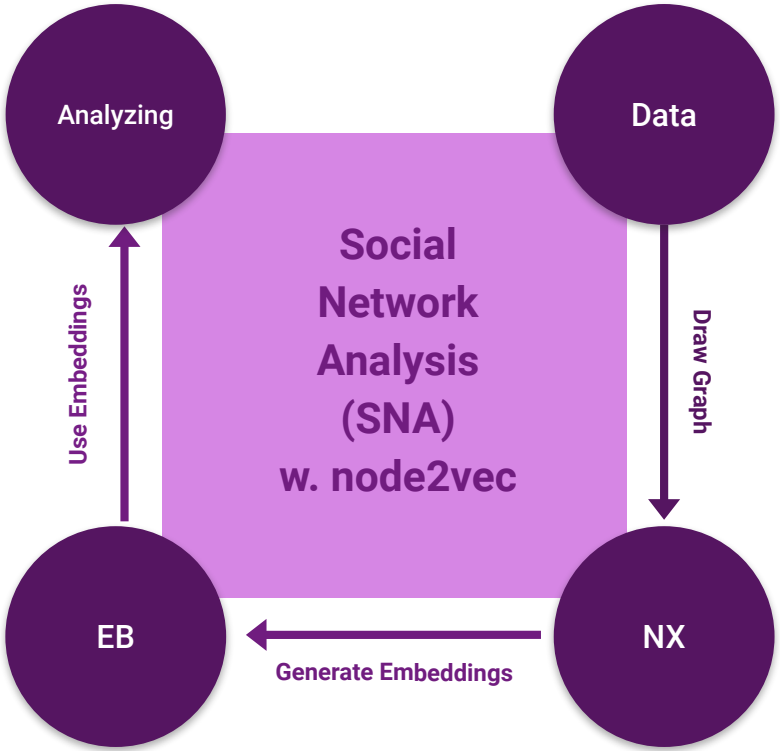
Top 2 of the most similar influencers to *"Naeun Son"*	
Influencers	**Similarity Score**
"BlackPink"	0.690971255
"Beyonce"	0.666431148

SNA Implementation



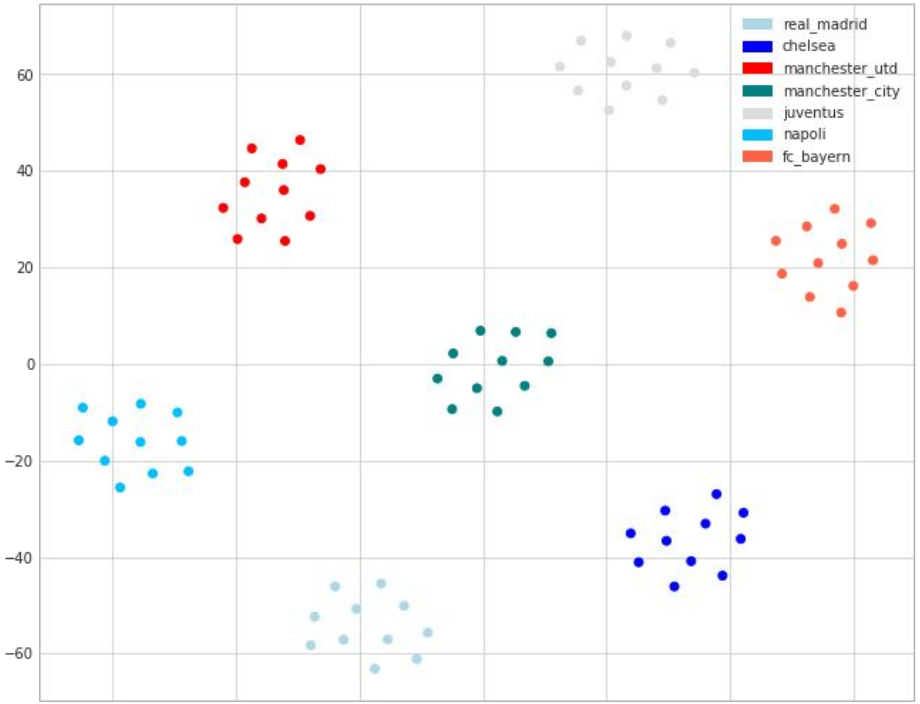

```
model.crosstable()


Celebrity     Celebrity
Ally Love     Adriene Mishlder      3.0
Bad Bunny     Adriene Mishlder      0.0
              Ally Love            41.0
Beyonce       Adriene Mishlder      0.0
              Ally Love            26.0
                                   ...
Zoe Saldana   Kerwin Frost          6.0
              Naeun Son             9.0
              Ninjas Hyper         23.0
              Pharrell Williams    24.0
              Yara Shahidi         82.0
Length: 91, dtype: float64
```

SNA with node2vec



SNA with node2vec

Usage 2: T-SNE Plot



Usage 1: node2vec built-in feature

```
model.wv.most_similar("Celebrity 12",topn=5)
```

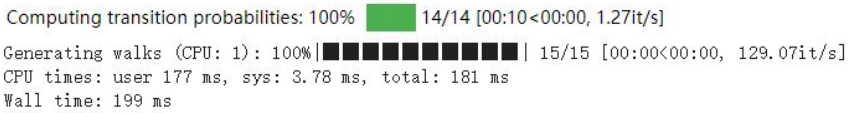
```
[('Celebrity 1', 0.9880407452583313),
 ('Celebrity 8', 0.9570997071266174),
 ('Celebrity 4', 0.9056358766555786),
 ('Celebrity 6', 0.8254547381401062),
 ('Celebrity 5', 0.7046592545509338)]
```

Usage 3: K-means Clustering

```
Group 1
                    0     ...         n
Celebrity 1  -0.019673     ...
0.085747
Celebrity 3  -0.011764     ...
0.091183
[2 rows x n columns]
....
Group k
                    0     ...         n
Celebrity 4  0.022665     ...  0.011859
Celebrity 5  0.018963     ...  0.010145
[2 rows x n columns]
```

SNA Implementation

Average execution time: 150 ms
---
node2vec saved ***99.98%*** of execution time



e.g. One of the outputs of training process

Usage 1: node2vec built-in feature

```
model.wv.most_similar( "Naeun Son",topn=2)
[('BlackPink', 0.9982033967971802),
 ('Beyonce', 0.988198926448822)]
```

Potential Future Work of Text Mining

Remove boilerplate to increase the informativeness and uniqueness of the “documents” for each celebrity.[4]

**In textual analysis, Boilerplate is a combination of words that can be removed from a sentence without significantly changing the original meaning*

```
[(('bot', 'action', 'performed', 'automatically'), 4186),
 (('action', 'performed', 'automatically', 'please'), 3454),
 (('performed', 'automatically', 'please', 'contact'), 3454),
 (('automatically', 'please', 'contact', 'moderator'), 3454)]
```

Case study - node2vec

K-means Clustering

```
Group 1
              0         1         2   ...        13        14        15
Sana   -0.089618 -0.071649 -0.003039  ... -0.043400 -0.102120  0.014032
NCT    -0.109779 -0.097330  0.032881  ... -0.048294 -0.241587  0.031689
Miyeon -0.144210 -0.067727  0.037089  ... -0.058805 -0.197231 -0.010469

[3 rows x 16 columns]
------------------------------------------------------------
Group 2
                0         1         2   ...        13        14        15
BTS       -0.186073 -0.159345  0.043289  ... -0.118391 -0.346044  0.027395
BlackPink -0.186074 -0.117281  0.071182  ... -0.112053 -0.334396 -0.008005
Naeun Son -0.139308 -0.119381  0.069812  ... -0.106206 -0.317578 -0.010998
Seolhyun  -0.220078 -0.148290  0.036138  ... -0.085400 -0.346950  0.023048

[4 rows x 16 columns]
------------------------------------------------------------
Group 3
              0         1         2   ...        13        14        15
GFriend -0.234220 -0.147478  0.064113  ... -0.101717 -0.409747 -0.007356
Solar   -0.260542 -0.194174  0.086498  ... -0.134705 -0.450908  0.024723
iZone   -0.236252 -0.178684  0.092774  ... -0.120792 -0.407976  0.011584

[3 rows x 16 columns]
------------------------------------------------------------
```