



CAPSTONE PROJECT

---

# ARABIC POETRY NLP

Munera AlRajhi

# Table of Contents

---

## ■ INTRODUCTION

Project Scope, Idea, and Motivation.

## ■ DATA COLLECTION AND PROCESSING.

Web Scraping, EDA, and NLP Processing.

## ■ DATA CLASSIFICATION

Multi-class classification and Topic Modeling.

## ■ CONCLUSION

Challenges and Technical Issues, Future Work, Acknowledgment

# INTRODUCTION

---



# WHY ARABIC POETRY?

---



Something I'm  
passionate about



Challenging Topic



Something New!

## PROJECT SCOPE

# WHAT ARE OUR MAIN QUESTIONS?

---

- HOW TO CLASSIFY POEMS BASED ON LABELS?
- HOW TO PREDICT LABELS BASED ON WORDS CLUSTERING?

# DATA COLLECTION & PROCESSING.

---



# DATA COLLECTING

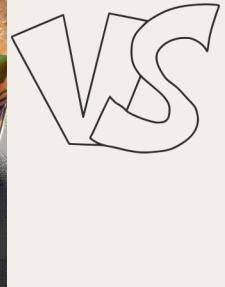
## HOW?

# Web Scraping.

# FROM WHERE?

AlDiwan for Arabic Poetry Webstibe: <https://www.aldiwan.net/>

YES, LET'S DO IT!!!!



The screenshot shows a browser window with several tabs open, all related to Arabic Poetry and Natural Language Processing (NLP). The tabs include:

- Slack - students-only
- Capstone Scheduling
- Misk Skills - Data Sci
- DSI\_Capstone\_NLP
- Arabic-Poetry-NLP()
- الصلة - المدون
- Arabic-Poetry-NLP() -

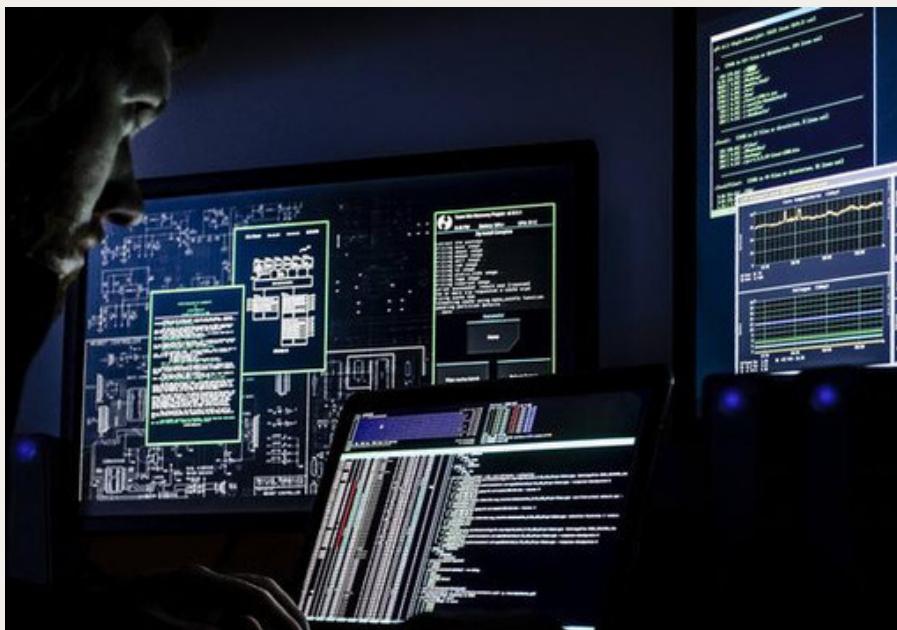
The main content area displays a search interface for poems. A search bar at the top contains the text "أبو بكر الطهري". Below it, a list of poems by this author is shown in a grid format. Each poem entry includes the title, author's name, and a snippet of the poem text.

العنوان	أبو بكر الطهري	مختصر المحتوى
حور شفان قلبي ينبع	أبو بكر الطهري	سورة ناهها أم غلام
أنت ترى السلام وقد تولى	أبو بكر الطهري	يا شيبة البدر حسا
بقرة وجه الملاك	أبو بكر الطهري	عنالك كات موطناً ولادنا
ورود بستان حلية	أبو بكر الطهري	ثغر من
يجالعن الآباء أفرق نورها	أبو الحسن الكلباني	سل حماجر ان جشت أرض حابر
رث بagan غزال اكل	أبو الحسن الكلباني	يا غزال العاقل طوي المصل
وعلج قد بالغ بها	أبو الحسن الكلباني	في حسن وجهك ساخت البرهان
سک العصی هر گندو الحسان	أبو الحسن الكلباني	

On the right side of the screen, the developer tools are open, specifically the Elements and Styles panels. The Styles panel shows a detailed breakdown of the CSS rules applied to the page elements, including media queries and specific class definitions like ".grid-item" and ".grid-item--col-12".

## MY EXPECTATION

---



## REALITY

---



Sitting there and trying  
to understand someone  
else code, be like:



# WEB SCRAPING

```
for page in range(1,32):
    url = requests.get(f"https://www.aldiwan.net/Poems-Topics-%D8%BA%D8%B2%D9%84.html?page={page}")
    soup = BeautifulSoup(url.text, 'html.parser')
    results = soup.find_all('div', attrs={'class':'record col-12'})

    for result in results:
        href = result.find('a')['href']
        GazalPages.append((href))

for page in GazalPages:

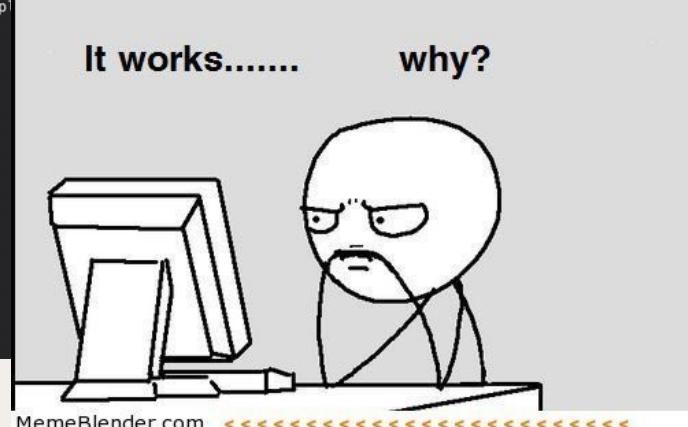
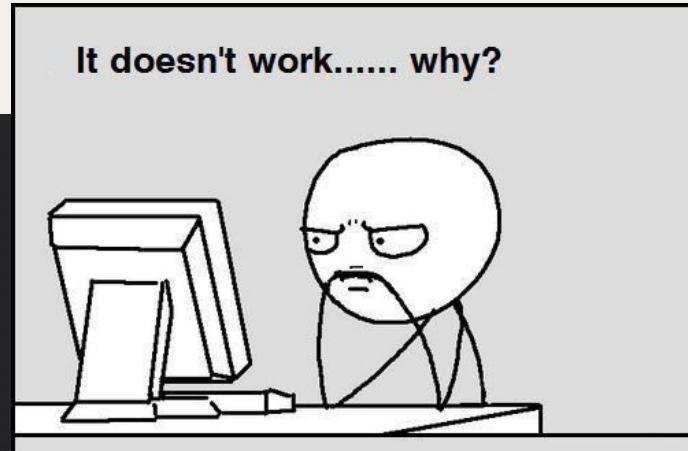
    url = requests.get(f"https://www.aldiwan.net/{page}")
    soup = BeautifulSoup(url.content, 'html.parser')
    poem = str(soup.find('div', attrs={'class':'bet-1 row pt-0 px-5 pb-4 justify-content-center'}).text.strip())
    VersusNumber= soup.find('p', attrs={'class':'d-inline-block px-2 mt-0 mb-1 poem-control main-color'})
    poet = soup.find('h2', attrs={'class':'text-center h3 mt-3 mb-0'}).text.strip()
    century = soup.find('p', attrs={'class':'main-color text-center mb-0 ltr'}).text.strip()

    sss = soup.find_all('div', attrs={'class':'col-6 col-md-3'})
    dummy = []

    for heading in sss:
        info = heading.find('a')
        dummy.append(info.text.strip())

    GazalPoems.append((poem,VersusNumber,poet,century,dummy[0],dummy[1],dummy[2]))
    print(page)

GazalPoems=pd.DataFrame(GazalPoems,columns=['Poem','Number of Versus','Poet',"Century","Label","Type",'Metre'])
```



# life lessons!!

---

- **ALWAYS** KEEP YOUR CODE NEAT AND UNDERSTANDABLE, SO OTHER PEOPLE WON'T MAKE MEMES ABOUT YOU!
- **BE PATIENT**, DON'T JUMP TO CONCLUSIONS AND ONLY FOCUS ON YOUR NEXT STEP

# life lessons!!

---

SAYING **YES** TO **LUJAIN** IS LIKE ORDERING FROM STARBUCKS, BOTH WILL COST YOU SOMETHING!! EITHER EXTRA WORK OR EXTRA MONEY!!

# FINAL DATASET

## FINALLY!!

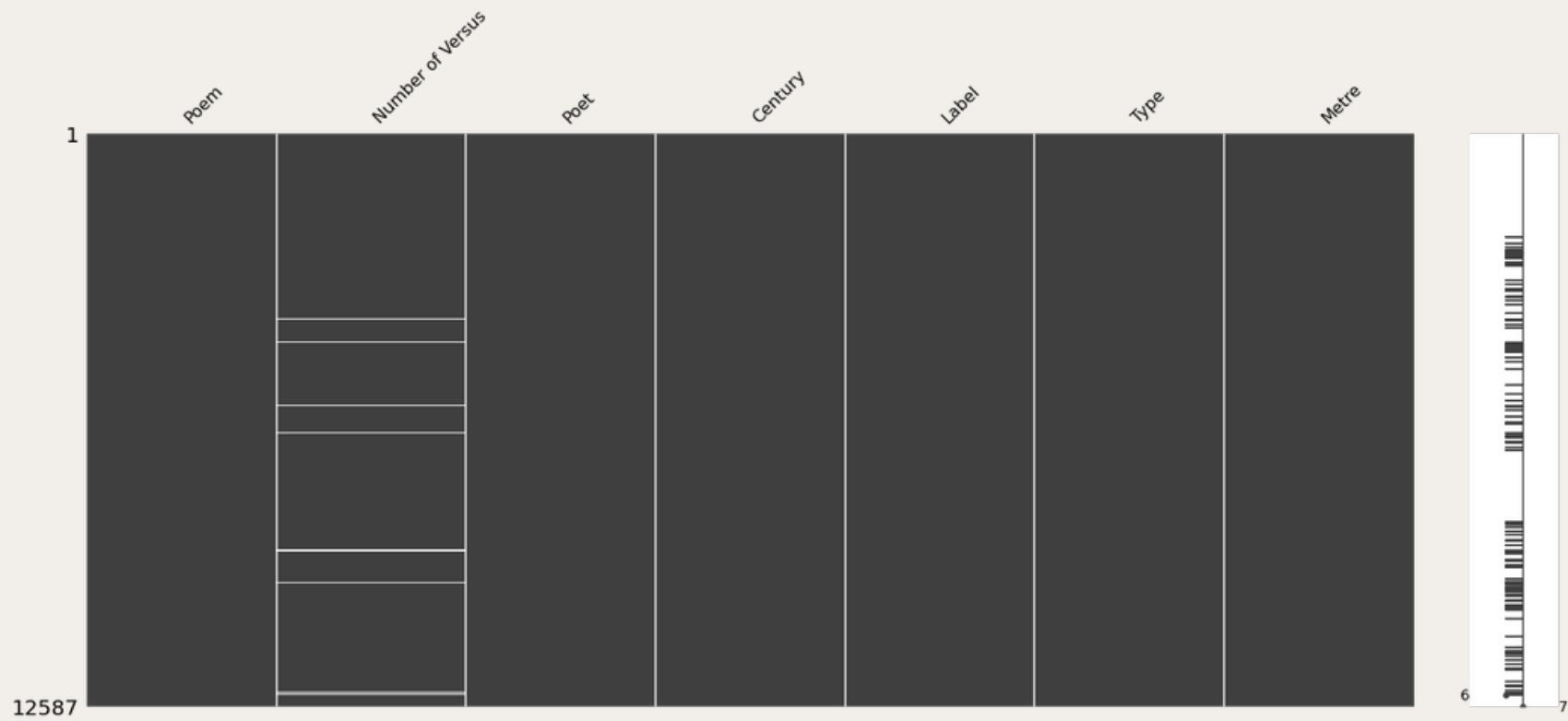
we now have a complete dataset, with a 12587 poem along with 7 features.

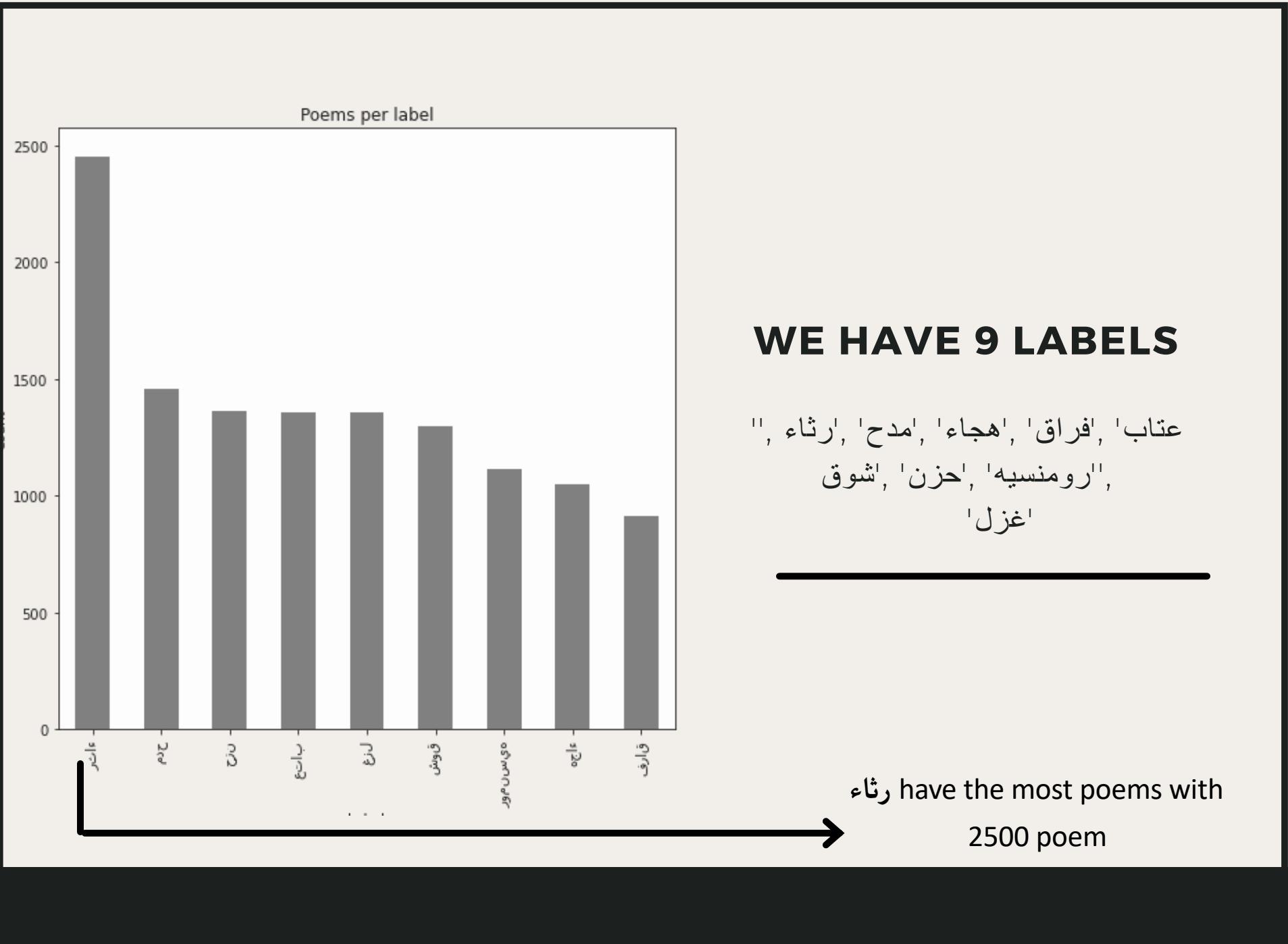
	Poem	Number of Versus	Poet	Century	Label	Type	Metre
12226	...أتلك رياض أم خدود نواعم وفيها أقاح أم ثغور بوا	82	الإرجاني	العصر الأندلسي	غزل	عموديه	بحر الطويل
10107	...لم يبق بينك من جمالك منسما لمن الذهب وفي طلاب	24	سليمان الصولة	سوريا	شوق	عموديه	بحر الكامل
1753	...اسقني واسق يوسفا مزة الطعم قرقفا دع من العيش ك	7	ابو نواس	العصر العباسي	فراق	عموديه	بحر مجزوء الخفيف
4984	...جاورت ريك يا أبا العباس وتركت شبلك رحمة للناس	24	سليمان الصولة	سوريا	رثاء	عموديه	بحر الكامل
6075	...سقى الله تلك الدار هامية القطر مدى الدهر ما نا	19	العشاري	العصر العثماني	رثاء	عموديه	بحر الطويل
2247	...إذا وهى الحب فالهجران يقتله وإن تمكן فالهجران	2	خليل مطران	لبنان	فراق	عموديه	بحر البسيط
7836	...على وجهه بالكاس صرف سلافة سقاني بيمناه فطابت ب	2	المفقى عبداللطيف فتح الله	لبنان	رومنسية	عموديه	بحر الطويل
8799	...أرى جسمى تحط به البلايا وما شارفت معترك المناي	2	المحي	العصر العثماني	حزن	عموديه	بحر الوافر
4190	...سلام يفوق الروض فيه الأزاهر ويزري بعقد الدر في	23	المفقى عبداللطيف فتح الله	لبنان	مدح	عموديه	بحر الطويل
4641	...كساك الصوم أعمار الليالي وأعقبك العنيمة في الم	8	ابن بابك	العصر العباسي	مدح	عموديه	بحر الوافر

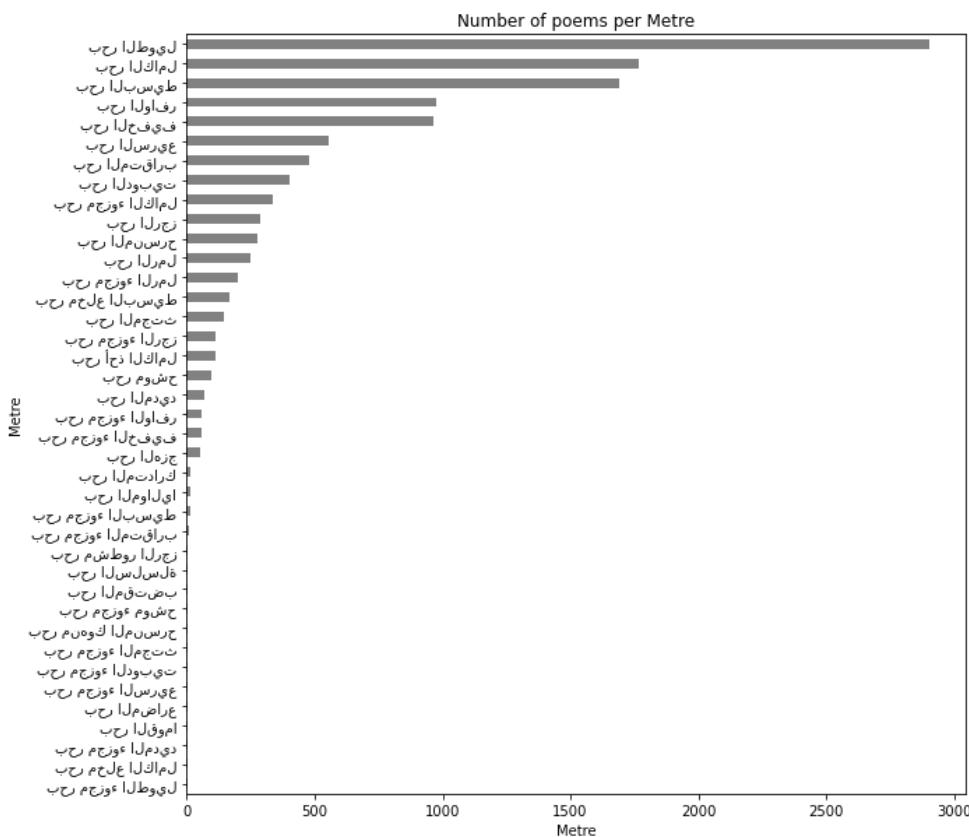
# DATA CLEANING AND PREPARATION

---

## Handling Missing Values

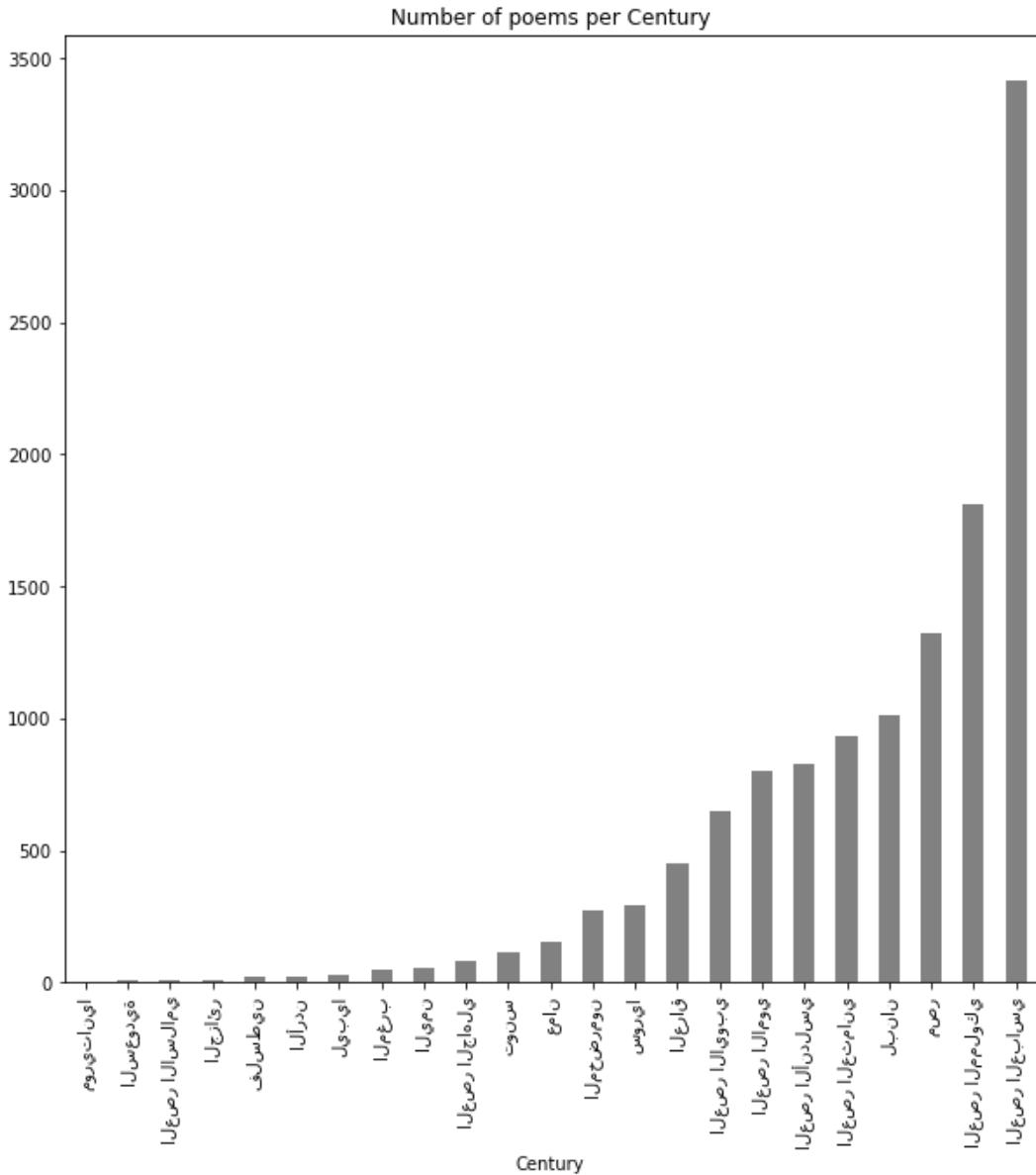






**بحر الطويل** have the most  
poems with 2901 poem

# **WE HAVE 66 METRE**



العَصْرُ الْعَبَاسِيُّ have the most  
poems with 3417 poem

# WE HAVE 23 CENTURY

# DATA PROCESSING

## TOKENIZATION

[إذا، اعتبرت، على، أخ،  
فاستبقة، الغد، ولا، اتهلك، بلا،  
إخوان]

## STEMMING

[إذا، اعتب، على، اخ، سبق،  
الغد، ولا، تهل، بلا، اخو]

## REMOVING STOP WORDS

[اعتبر، سبق، الغد، تهل، بلا]

, (6280 , 'قلب')]  
, (4110 , 'علم')  
, (3827 , 'حسن')  
, (3724 , 'دهر')  
, (3656 , 'كرم')  
, (3639 , 'بعد')  
, (3504 , 'ليل')  
, (3378 , 'دمع')  
, (3336 , 'عين')  
, (3194 , 'نفس')  
, (3081 , 'سلم')  
, (2962 , 'فضل')  
, (2900 , 'ذكر')  
, (2810 , 'امر')  
, (2686 , 'فرق')  
, (2674 , 'قبل')  
, (2657 , 'بدر')  
, (2568 , 'علي')  
, (2565 , 'هوى')  
, (2560 , 'سعد')  
, (2558 , 'حمد')  
, (2540 , 'نظر')  
, (2472 , 'زمن')  
, (2430 , 'خلق')  
, (2372 , 'وجد')  
, (2370 , 'عرف')  
, (2335 , 'جمع')  
, (2314 , 'كنت')  
, (2280 , 'وصل')  
[(2257 , 'قوم')]

# Most frequent 30 Words



Qalab means in Heart.

# DATA CLASSIFICATION.

---



# MULTI-CLASS CLASSIFICATION

---

Can we build an ML classifier to predict poem labels?

The labels are: عتاب, فراق, هجاء, مدح, رثاء, رومنسية, حزن, شوق, غزل

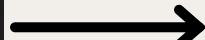
- 1 Text Encoding
- 2 Build and Evaluate Classification Models
- 3 Testing

# MULTI-CLASS CLASSIFICATION

1

# Text Encoding

I used Count Vectorizer to create a Bag of Words and encode them.



```
array([[0, 0, 0, ..., 0, 0, 0],  
       [0, 0, 0, ..., 0, 0, 0],  
       [0, 0, 0, ..., 0, 0, 0],  
       ...,  
       [0, 0, 0, ..., 0, 0, 0],  
       [0, 0, 0, ..., 0, 0, 0],  
       [0, 0, 0, ..., 0, 0, 0]])
```

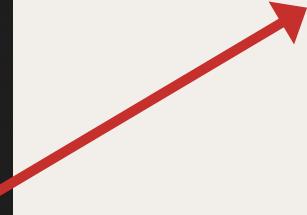
# MULTI-CLASS CLASSIFICATION

2

## Build and Evaluate Classification Models

	precision	recall	f1-score	support
275	0.30	0.25	0.37	حزن
479	0.39	0.47	0.33	رثاء
226	0.35	0.41	0.31	رومانسيه
253	0.49	0.49	0.50	شوق
279	0.19	0.17	0.23	عتاب
260	0.32	0.30	0.34	غزل
187	0.31	0.25	0.41	فراق
293	0.55	0.61	0.51	مدح
218	0.26	0.22	0.31	هجاء
accuracy			0.37	2470
macro avg	0.37	0.35	0.35	2470
weighted avg	0.36	0.37	0.36	2470

We can notice here  
we have low  
Accuracy which  
means the model is  
underfitting



# MULTI-CLASS CLASSIFICATION

---

2

## Build and Evaluate Classification Models

even after trying different model the problem still there!!

	Model	Score
2	Random Forest	0.363158
1	Multinomial Naive Bayes	0.359109
0	Logistic Regression	0.333603
3	LinearSVC	0.301619



# MULTI-CLASS CLASSIFICATION

## 3 Testing

```
#qoute
test_text = ["العمرى لقد أوهيت قلبي عن العز وطاطات رأس وشفة كليب لقد قسمت مني قناعة صلبة ويقسم عود النبع وهو صلبيب"]
# حزن
# encoding
test_vector = count_vector.transform(test_text)
test_vector = test_vector.toarray()

## Perform and Evaluate the Model
text_predict_class = encoder.inverse_transform(RandomModel.predict(test_vector))
print(test_text)
print(text_predict_class)

[("العمرى لقد أوهيت قلبي عن العز وطاطات رأس والفؤاد كليب لقد قسمت مني قناعة صلبة ويقسم عود النبع وهو صلبيب",
  'رومنسيه')]
```

```
#qoute
test_text = ["إن وترت قلبك [الهموم] فـ[ما] مثل [انتصار] بالـ[نتـاي] والـ[لوـتر] وـ[شـادـن] حيرت لـ[واحـظـه] الـ[حـاظـه] عـين [الـغـازـال] بـ[الـحـورـ']"]
# غزل
# encoding
test_vector = count_vector.transform(test_text)
test_vector = test_vector.toarray()

## Perform and Evaluate the Model
text_predict_class = encoder.inverse_transform(RandomModel.predict(test_vector))
print(test_text)
print(text_predict_class)
```

[ إن وترت قلبك الهموم فـما مثل انتصار بالـنتـاي والـلوـتر وـشـادـن حـيرـت لـواحـظـه الـحـاظـه عـين الغـازـال بـالـحـورـ' ]  
[ غـزل ]

# TOPIC MODELING

How to predict labels based on words clustering?

```
,0)]  
' + "بدر" *0.008 + "زمر" *0.007 + "شعر" *0.008 + "لليل" *0.007 + "ما" *0.009'  
' + "بحر" *0.007 + "نور" *0.006 + "سحر" *0.005 + "نظر" *0.006 + "صدر" ()  
.1)  
' + "قلب" *0.014 + "دمع" *0.015 + "هوى" *0.020 + "لحظ" *0.037'  
' + "عشق" *0.009 + "حب" *0.010 + "لحب" *0.008 + "صبيح" *0.011'  
.2)  
' + "حبه" *0.022 + "هوك" *0.016 + "مدح" *0.017 + "رسق" *0.016  
' + "ريل" *0.015 + "عذل" *0.013 + "صب" *0.015 + "ريما" *0.013 + "ربب" ()  
.3)  
' + "حسن" *0.012 + "جمل" *0.008 + "دهر" *0.010 + "كنت" *0.008  
' + "روح" *0.007 + "انس" *0.006 + "عذر" *0.007 + "عرض" *0.007 + "زمن" ()  
.4)  
' + "ابيك" *0.015 + "دمى" *0.022 + "دمى" *0.015 + "فليل" *0.019 + "كلب" *0.015  
' + "نصل" *0.011 + "نول" *0.011 + "نصل" *0.011 + "نول" *0.012 + "رضع" ()  
.5)  
' + "شمس" *0.006 + "طرف" *0.006 + "سيف" *0.006 + "غرب" *0.007 + "كرم" *0.009  
' + "حمد" *0.005 + "سلام" *0.005 + "وجه" *0.006 + "عجب" *0.006 + "سلم" *0.006
```

حب

```
+ "طبى" *0.011 + "يلى" *0.010 + "غزل" *0.009 + "ظبه" *0.010 + "غفل" *0.009  
' + "دجي" *0.008 + "ليل" *0.007 + "بحس" *0.008 + "حيبا" *0.007 + "جين" ()  
.7)  
' + "ملك" *0.010 + "نسم" *0.005 + "ارض" *0.007 + "ربيع" *0.008 + "فخر" *0.005  
' + "حلو" *0.004 + "لخد" *0.005 + "ببل" *0.005 + "جيد" *0.005 + "ملك" *0.010  
.9)  
+ "تعب" *0.012 + "سقط" *0.010 + "رسي" *0.011 + "تفا" *0.012 + "مزن" *0.010 + "تفا" *0.012  
' + "حشر" *0.007 + "أجب" *0.008 + "سير" *0.009 + "ازر" *0.007 + "قدد" *0.008 + "حشر" ()
```

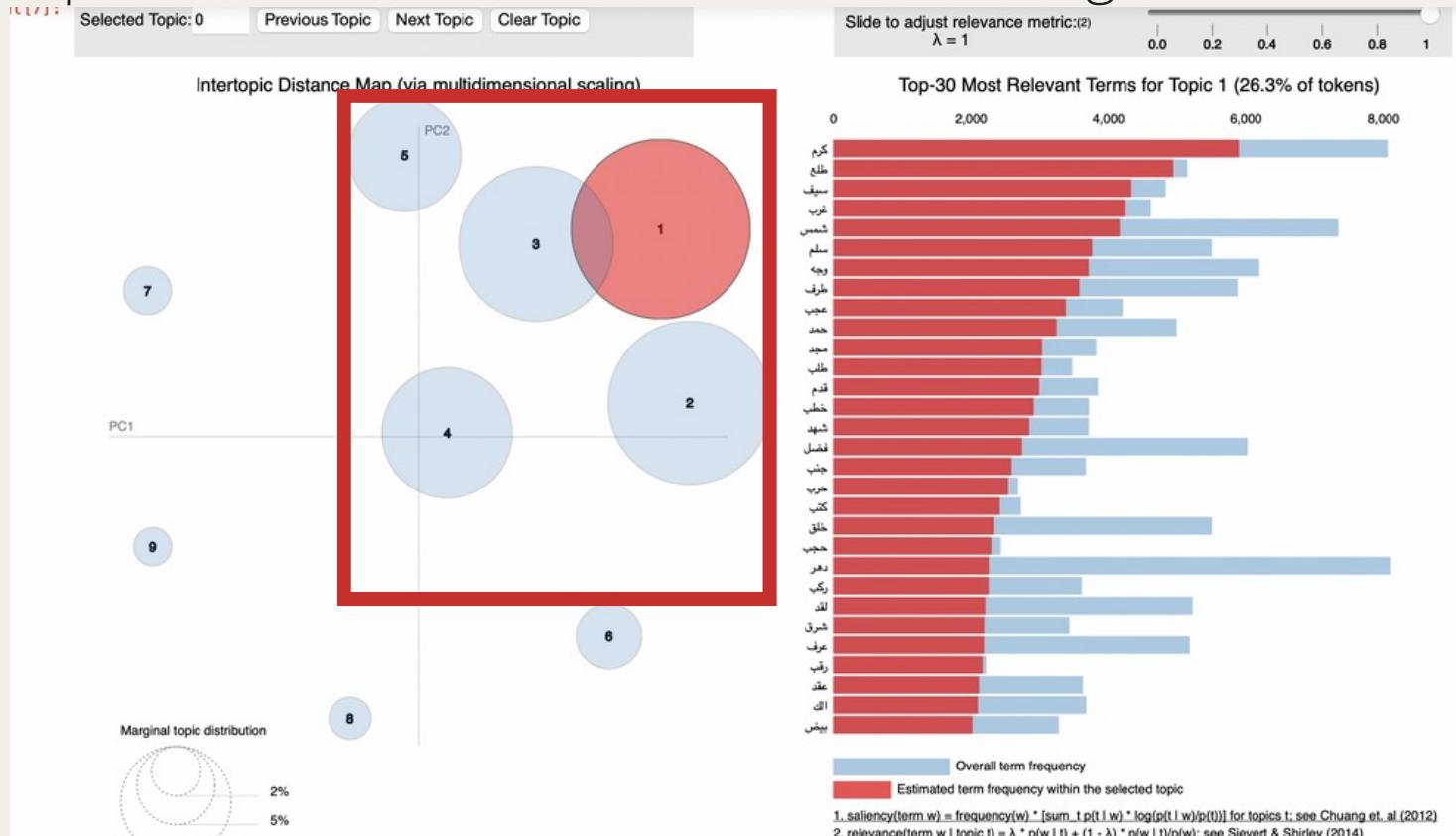
مدح

```
+ "تفا" *0.012 + "سقط" *0.010 + "رسي" *0.011 + "تفا" *0.012 + "مزن" *0.010 + "تفا" *0.012  
' + "حشر" *0.007 + "أجب" *0.008 + "سير" *0.009 + "ازر" *0.007 + "قدد" *0.008 + "حشر" ()
```

فراق

# TOPIC MODELING

How to predict labels based on words clustering?



# TOPIC MODELING

---

How to predict labels based on words clustering?

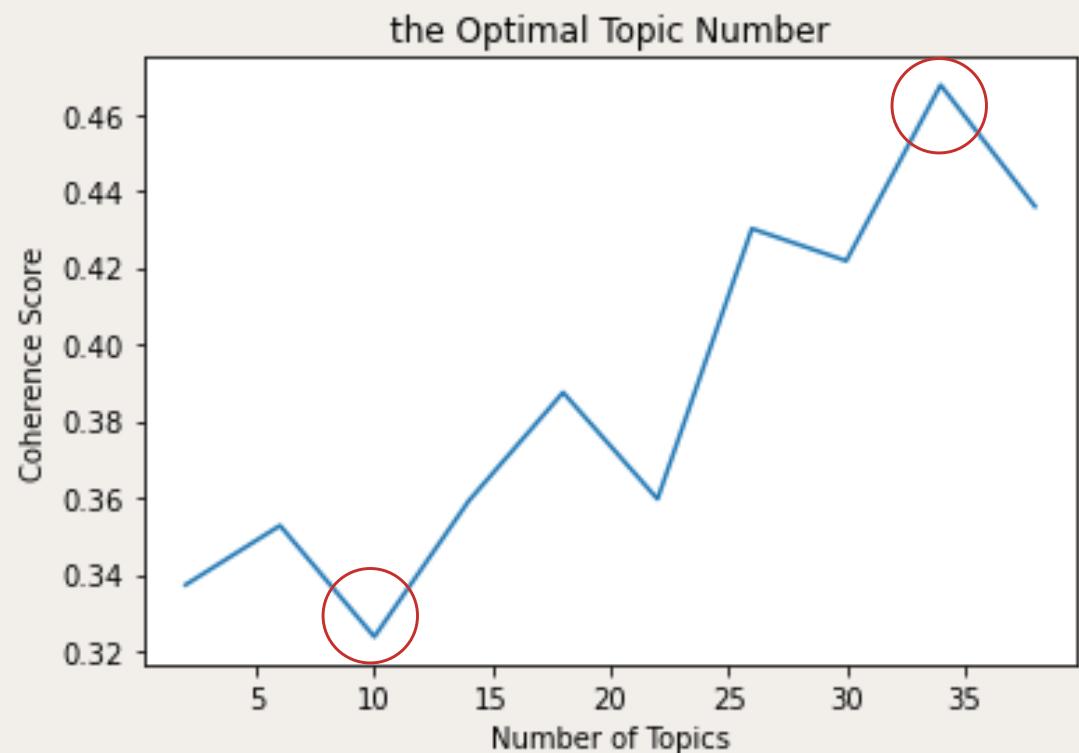
## Evaluation:

The Number of Topics we used: **9**

**Coherence Score: 0.29**

The Optimal Topic Number: **34**

**Coherence score: 0.46**



# CONCLUSION

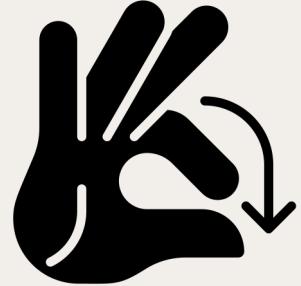
---



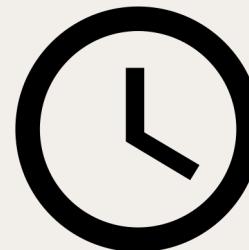
Trường Durmistrang và trường Beauxbatons  
đều là những trường học mà không ai có thể chinh  
phục. Trường Durmistrang và trường Beauxbatons  
đều là những trường học mà không ai có thể chinh  
phục. Trường Durmistrang và trường Beauxbatons  
đều là những trường học mà không ai có thể chinh  
phục. Trường Durmistrang và trường Beauxbatons  
đều là những trường học mà không ai có thể chinh  
phục. Trường Durmistrang và trường Beauxbatons  
đều là những trường học mà không ai có thể chinh  
phục.

# CHALLENGES AND LIMITATION

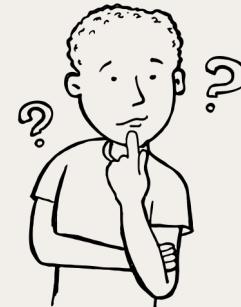
SMALL DATASET



TIME LIMITATION

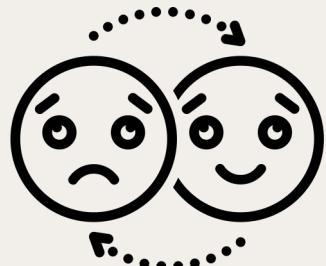


NEW TOPICS

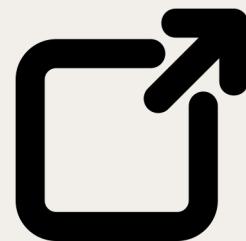


# FUTURE WORK

SENTIMENT  
ANALYSIS



EXPAND THE  
DATASET



IMPROVE THE  
CLASSIFICATION  
MODEL



# KEY TAKEAWAYS

---



BE PATIENT!

ALWAYS TEST  
YOUR LIMITS.

**GOOGLE AND  
YOUTUBE ARE  
YOUR BEST  
FRIENDS.**



DR. RICK SCAVETTA  
Instructor



LUJAIN FELEMBAN  
Assistant Instructor

## ACKNOWLEDGMENT

---

# REFERENCES:

---

- **How to Create an LDA Topic Model in Python with Gensim:**

[https://www.youtube.com/watch?v=TKjjlp5\\_r7o](https://www.youtube.com/watch?v=TKjjlp5_r7o)

- **Misk Skills Data Science Book:**

[http://www.ylz.ncx.mybluehost.me/scavetta.academy/misk/11\\_nlp/materials/code/\\_build/html/04\\_nlp\\_unsupervised\\_learning.html](http://www.ylz.ncx.mybluehost.me/scavetta.academy/misk/11_nlp/materials/code/_build/html/04_nlp_unsupervised_learning.html)

- **NLTK book Chapter1 Language Processing and Python:**

<https://www.nltk.org/book/ch01.html>

- **Count Vectorizer vs TFIDF Vectorizer | Natural Language Processing:**

<https://www.linkedin.com/pulse/count-vectorizers-vs-tfidf-natural-language-processing-sheel-saket/>

- **Product-Categorization-NLP:**

[https://github.com/aniass/Product-Categorization-NLP/blob/master/Text\\_analysis.ipynb](https://github.com/aniass/Product-Categorization-NLP/blob/master/Text_analysis.ipynb)