# Lab 2 : Word embeddings in NLP

In [1]:
```
# Taking the file of corpus
import pandas as pd

file = pd.read_csv('corpus.txt', delimiter='\t', header=None, names=['text'])

text = (file['text'].tolist())
```

## Q1. Frequency-based embedding: Count Vectors

In [2]:
```
from sklearn.feature_extraction.text import CountVectorizer

vectorizer = CountVectorizer()

# Finds the unique words
vectorizer.fit(text)

# prints the unique words
print(vectorizer.vocabulary_)
```

{'lionel': 195, 'andrés': 55, 'leo': 191, 'messi': 206, 'note': 214, 'spanish': 271, 'pronunciation': 239, 'ljoˈnel': 196, 'anˈdres': 57, 'ˈmesi': 324, 'born': 80, '24': 30, 'june': 181, '1987': 10, 'is': 176, 'an': 53, 'argentine': 60, 'professional': 237, 'footballer': 137, 'who': 312, 'plays': 235, 'as': 61, 'forward': 143, 'for': 140, 'and': 54, 'captains': 89, 'both': 81, 'major': 200, 'league': 187, 'soccer': 268, 'club': 95, 'inter': 174, 'miami': 207, 'the': 284, 'argentina': 59, 'national': 211, 'team': 280, 'widely': 313, 'regarded': 245, 'one': 219, 'of': 217, 'greatest': 154, 'players': 233, 'all': 48, 'time': 292, 'set': 260, 'numerous': 215, 'records': 243, 'individual': 170, 'accolades': 40, 'won': 317, 'throughout': 291, 'his': 164, 'footballing': 138, 'career': 90, 'such': 277, 'eight': 119, 'ballon': 68, 'or': 220, 'awards': 67, 'times': 293, 'being': 74, 'named': 209, 'world': 318, 'best': 75, 'player': 232, 'by': 83, 'fifa': 127, 'he': 158, 'most': 208, 'decorated': 108, 'in': 168, 'history': 166, 'football': 136, 'having': 157, '45': 36, 'trophies': 301, 'including': 169, 'twelve': 303, 'big': 78, 'five': 133, 'titles': 295, 'four': 144, 'uefa': 305, 'champions': 93, 'leagues': 188, 'two': 304, 'copa': 102, 'américas': 52, 'cup': 106, 'holds': 167, 'european': 124, 'golden': 153, 'shoes': 262, 'goals': 150, 'single': 265, '672': 38, 'with': 315, 'barcelona': 70, '474': 37, 'hat': 156, 'tricks': 300, '36': 34, 'assists': 62, '192': 9, 'la': 182, 'liga': 193, 'matches': 204, 'played': 231, '39': 35, '18': 7, 'goal': 149, 'contributions': 101, '34': 33, 'américa': 51, '26': 31, '21': 28, 'international': 175, 'appearances': 58, '191': 8, '112': 2, 'south': 269, 'american': 49, 'male': 201, 'second': 258, 'latter': 184, 'category': 91, 'outright': 223, 'prolific': 238, 'goalscorer': 151, 'creative': 104, 'playmaker': 234, 'has': 155, 'scored': 254, 'over': 224, '850': 39, 'senior': 259, 'country': 103, 'rosario': 251, 'relocated': 246, 'to': 296, 'spain': 270, 'join': 178, 'at': 64, 'age': 46, '13': 4, 'made': 199, 'competitive': 99, 'debut': 107, '17': 6, 'october': 216, '2004': 11, 'established': 122, 'himself': 163, 'integral': 173, 'within': 316, 'next': 213, 'three': 290, 'years': 321, 'first': 132, 'uninterrupted': 306, 'season': 256, '2008': 14, '09': 0, 'helped': 160, 'achieve': 42, 'treble': 299, 'that': 283, 'year': 320, 'aged': 47, '22': 29, 'consecutive': 100, 'ballons': 69, 'win': 314, 'it': 177, 'during': 116, '2011': 16, '12': 3, 'while': 311, 'establishing': 123, 'top': 297, 'scorer': 255, 'following': 135, 'seasons': 257, 'finished': 131, 'behind': 73, 'cristiano': 105, 'ronaldo': 250, 'perceived': 229, 'rival': 249, 'before': 72, 'regaining': 244, 'form': 142, '2014': 18, '15': 5, 'campaign': 85, 'where': 309, 'became': 71, 'led': 189, 'historic': 165, 'fifth': 128, '2015': 19, 'assumed': 63, 'captaincy': 88, '2018': 21, 'record': 242, 'sixth': 267, '2019': 22, 'overall': 225, 'tenure': 282, 'ten': 281, 'among': 50, 'others': 221, 'signed': 263, 'french': 147, 'paris': 227, 'saint': 252, 'germain': 148, 'august': 66, '2021': 24, 'would': 319, 'ligue': 194, 'title': 294, 'there': 286, 'joined': 179, 'july': 180, '2023': 26, 'new': 212, 'mark': 203, 'leading': 186, 'capped': 86, 'style': 275, 'play': 230, 'diminutive': 111, 'left': 190, 'footed': 139, 'dribbler': 114, 'drew': 113, 'long': 197, 'comparisons': 97, 'compatriot': 98, 'diego': 110, 'maradona': 202, 'described': 109, 'successor': 276, 'youth': 323, 'level': 192, '2005': 12, 'championship': 94, 'gold': 152, 'medal': 205, 'summer': 278, 'olympics': 218, 'after': 44, 'youngest': 322, 'score': 253, '2006': 13, 'then': 285, 'finals': 130, 'centenario': 92, 'which': 310, 'they': 287, 'lose': 198, 'initially': 172, 'announcing': 56, 'retirement': 247, '2016': 20, 'returned': 248, 'help': 159, 'narrowly': 210, 'qualify': 240, 'again': 45, 'exit': 125, 'early': 117, 'finally': 129, 'broke': 82, '28': 32, 'trophy': 302, 'drought': 115, 'victory': 307, 'was': 308, 'tournament': 298, 'later': 183, 'him': 162, 'seventh': 261, '2022': 25, 'third': 288, 'this': 289, 'followed': 134, 'extending': 126, 'eighth': 120, 'captain': 87, 'came': 84, '2024': 27, 'endorsed': 121, 'sportswear': 274, 'company': 96, 'adidas': 43, 'since': 264, 'according': 41, 'france': 145, 'highest': 161, 'paid': 226, 'out': 222, 'six': 266, 'between': 77, '2009': 15, 'ranked': 241, 'athlete': 65, 'forbes': 141, '100': 1, 'influential': 171, 'people': 228, '2012': 17, '2020': 23, 'laureus': 185, 'sportsman': 273, 'sport': 272, 'bestowed': 76, 'presidential': 236, 'freedom': 146, 'dream': 112, 'surpass': 279, 'billion': 79, 'earnings': 118}

In [3]:
```
# Encode the Document
vector = vectorizer.transform(text)

# Summarizing the Encoded Texts
print("The Count Encoded Vector of the Document is:")
print(vector.toarray())
```

```
The Count Encoded Vector of the Document is:
[[0 0 1 ... 0 0 1]
 [1 0 0 ... 0 0 0]
 [0 0 0 ... 1 2 0]
 [0 1 0 ... 0 0 0]]
```

In [4]:
```
df = pd.DataFrame(vector.todense(), columns=vectorizer.get_feature_names_out())
df
```

Out[4]:

| | 09 | 100 | 112 | 12 | 13 | 15 | 17 | 18 | 191 | 192 | ... | with | within | won | world | would | y |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | ... | 1 | 0 | 2 | 3 | 0 | |
| **1** | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | ... | 0 | 1 | 4 | 0 | 1 | |
| **2** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 4 | 0 | 1 | 6 | 2 | |
| **3** | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 4 | 0 | |

4 rows × 325 columns

In [5]:
```
from sklearn.feature_extraction.text import TfidfVectorizer

tf_idf_vectorizer = TfidfVectorizer()

tf_idf_vectorizer.fit(text)

print(tf_idf_vectorizer.vocabulary_)
```

{'lionel': 195, 'andrés': 55, 'leo': 191, 'messi': 206, 'note': 214, 'spanish': 271, 'pronunciation': 239, 'ljoˈnel': 196, 'anˈdres': 57, 'ˈmesi': 324, 'born': 80, '24': 30, 'june': 181, '1987': 10, 'is': 176, 'an': 53, 'argentine': 60, 'professional': 237, 'footballer': 137, 'who': 312, 'plays': 235, 'as': 61, 'forward': 143, 'for': 140, 'and': 54, 'captains': 89, 'both': 81, 'major': 200, 'league': 187, 'soccer': 268, 'club': 95, 'inter': 174, 'miami': 207, 'the': 284, 'argentina': 59, 'national': 211, 'team': 280, 'widely': 313, 'regarded': 245, 'one': 219, 'of': 217, 'greatest': 154, 'players': 233, 'all': 48, 'time': 292, 'set': 260, 'numerous': 215, 'records': 243, 'individual': 170, 'accolades': 40, 'won': 317, 'throughout': 291, 'his': 164, 'footballing': 138, 'career': 90, 'such': 277, 'eight': 119, 'ballon': 68, 'or': 220, 'awards': 67, 'times': 293, 'being': 74, 'named': 209, 'world': 318, 'best': 75, 'player': 232, 'by': 83, 'fifa': 127, 'he': 158, 'most': 208, 'decorated': 108, 'in': 168, 'history': 166, 'football': 136, 'having': 157, '45': 36, 'trophies': 301, 'including': 169, 'twelve': 303, 'big': 78, 'five': 133, 'titles': 295, 'four': 144, 'uefa': 305, 'champions': 93, 'leagues': 188, 'two': 304, 'copa': 102, 'américas': 52, 'cup': 106, 'holds': 167, 'european': 124, 'golden': 153, 'shoes': 262, 'goals': 150, 'single': 265, '672': 38, 'with': 315, 'barcelona': 70, '474': 37, 'hat': 156, 'tricks': 300, '36': 34, 'assists': 62, '192': 9, 'la': 182, 'liga': 193, 'matches': 204, 'played': 231, '39': 35, '18': 7, 'goal': 149, 'contributions': 101, '34': 33, 'américa': 51, '26': 31, '21': 28, 'international': 175, 'appearances': 58, '191': 8, '112': 2, 'south': 269, 'american': 49, 'male': 201, 'second': 258, 'latter': 184, 'category': 91, 'outright': 223, 'prolific': 238, 'goalscorer': 151, 'creative': 104, 'playmaker': 234, 'has': 155, 'scored': 254, 'over': 224, '850': 39, 'senior': 259, 'country': 103, 'rosario': 251, 'relocated': 246, 'to': 296, 'spain': 270, 'join': 178, 'at': 64, 'age': 46, '13': 4, 'made': 199, 'competitive': 99, 'debut': 107, '17': 6, 'october': 216, '2004': 11, 'established': 122, 'himself': 163, 'integral': 173, 'within': 316, 'next': 213, 'three': 290, 'years': 321, 'first': 132, 'uninterrupted': 306, 'season': 256, '2008': 14, '09': 0, 'helped': 160, 'achieve': 42, 'treble': 299, 'that': 283, 'year': 320, 'aged': 47, '22': 29, 'consecutive': 100, 'ballons': 69, 'win': 314, 'it': 177, 'during': 116, '2011': 16, '12': 3, 'while': 311, 'establishing': 123, 'top': 297, 'scorer': 255, 'following': 135, 'seasons': 257, 'finished': 131, 'behind': 73, 'cristiano': 105, 'ronaldo': 250, 'perceived': 229, 'rival': 249, 'before': 72, 'regaining': 244, 'form': 142, '2014': 18, '15': 5, 'campaign': 85, 'where': 309, 'became': 71, 'led': 189, 'historic': 165, 'fifth': 128, '2015': 19, 'assumed': 63, 'captaincy': 88, '2018': 21, 'record': 242, 'sixth': 267, '2019': 22, 'overall': 225, 'tenure': 282, 'ten': 281, 'among': 50, 'others': 221, 'signed': 263, 'french': 147, 'paris': 227, 'saint': 252, 'germain': 148, 'august': 66, '2021': 24, 'would': 319, 'ligue': 194, 'title': 294, 'there': 286, 'joined': 179, 'july': 180, '2023': 26, 'new': 212, 'mark': 203, 'leading': 186, 'capped': 86, 'style': 275, 'play': 230, 'diminutive': 111, 'left': 190, 'footed': 139, 'dribbler': 114, 'drew': 113, 'long': 197, 'comparisons': 97, 'compatriot': 98, 'diego': 110, 'maradona': 202, 'described': 109, 'successor': 276, 'youth': 323, 'level': 192, '2005': 12, 'championship': 94, 'gold': 152, 'medal': 205, 'summer': 278, 'olympics': 218, 'after': 44, 'youngest': 322, 'score': 253, '2006': 13, 'then': 285, 'finals': 130, 'centenario': 92, 'which': 310, 'they': 287, 'lose': 198, 'initially': 172, 'announcing': 56, 'retirement': 247, '2016': 20, 'returned': 248, 'help': 159, 'narrowly': 210, 'qualify': 240, 'again': 45, 'exit': 125, 'early': 117, 'finally': 129, 'broke': 82, '28': 32, 'trophy': 302, 'drought': 115, 'victory': 307, 'was': 308, 'tournament': 298, 'later': 183, 'him': 162, 'seventh': 261, '2022': 25, 'third': 288, 'this': 289, 'followed': 134, 'extending': 126, 'eighth': 120, 'captain': 87, 'came': 84, '2024': 27, 'endorsed': 121, 'sportswear': 274, 'company': 96, 'adidas': 43, 'since': 264, 'according': 41, 'france': 145, 'highest': 161, 'paid': 226, 'out': 222, 'six': 266, 'between': 77, '2009': 15, 'ranked': 241, 'athlete': 65, 'forbes': 141, '100': 1, 'influential': 171, 'people': 228, '2012': 17, '2020': 23, 'laureus': 185, 'sportsman': 273, 'sport': 272, 'bestowed': 76, 'presidential': 236, 'freedom': 146, 'dream': 112, 'surpass': 279, 'billion': 79, 'earnings': 118}

In [7]:
```
# Encode the Document
vector = tf_idf_vectorizer.transform(text)

# Getting the TF Vector
tf_vectorizer = TfidfVectorizer(use_idf=False, norm=None)
tf_matrix = tf_vectorizer.fit_transform(text)

tf_array = tf_matrix.toarray()
print("The Term Frequency Encoded Vector of the Documents are:")
print(tf_array)
```

```
The Term Frequency Encoded Vector of the Documents are:
[[0. 0. 1. ... 0. 0. 1.]
 [1. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 1. 2. 0.]
 [0. 1. 0. ... 0. 0. 0.]]
```

In [8]:
```
print("The IDF Vector of the Documents are:")
print(tf_idf_vectorizer.idf_)
```

```
The IDF Vector of the Documents are:
[1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.51082562 1.51082562 1.91629073 1.22314355 1.91629073
 1.22314355 1.51082562 1.91629073 1.51082562 1.51082562 1.91629073
 1.51082562 1.51082562 1.22314355 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.51082562 1.51082562 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.22314355 1.51082562 1.51082562 1.51082562 1.91629073 1.22314355
 1.         1.91629073 1.91629073 1.91629073 1.91629073 1.22314355
 1.51082562 1.22314355 1.91629073 1.51082562 1.51082562 1.91629073
 1.91629073 1.91629073 1.         1.91629073 1.51082562 1.22314355
 1.91629073 1.91629073 1.91629073 1.22314355 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.51082562 1.91629073 1.22314355
 1.91629073 1.91629073 1.91629073 1.91629073 1.51082562 1.91629073
 1.         1.91629073 1.91629073 1.51082562 1.91629073 1.51082562
 1.91629073 1.91629073 1.91629073 1.91629073 1.51082562 1.91629073
 1.51082562 1.51082562 1.91629073 1.91629073 1.51082562 1.51082562
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.51082562 1.91629073
 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073 1.91629073
 1.22314355 1.51082562 1.91629073 1.51082562 1.22314355 1.51082562
 1.91629073 1.91629073 1.         1.91629073 1.91629073 1.91629073
 1.51082562 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.51082562 1.51082562 1.91629073 1.91629073 1.91629073 1.51082562
 1.91629073 1.91629073 1.         1.91629073 1.51082562 1.91629073
 1.91629073 1.91629073 1.22314355 1.91629073 1.91629073 1.91629073
 1.         1.51082562 1.91629073 1.91629073 1.91629073 1.91629073
 1.51082562 1.51082562 1.51082562 1.51082562 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.51082562 1.91629073 1.91629073
 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.51082562 1.         1.51082562 1.         1.22314355
 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.         1.91629073 1.91629073 1.         1.91629073
 1.91629073 1.91629073 1.91629073 1.51082562 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.22314355 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.51082562 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073
 1.         1.51082562 1.51082562 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.51082562 1.91629073 1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.91629073 1.91629073 1.22314355 1.91629073
 1.91629073 1.51082562 1.         1.91629073 1.91629073 1.91629073
 1.91629073 1.91629073 1.51082562 1.91629073 1.         1.51082562
 1.91629073 1.51082562 1.22314355 1.91629073 1.91629073 1.91629073
 1.91629073 1.51082562 1.91629073 1.91629073 1.51082562 1.51082562
 1.91629073 1.91629073 1.51082562 1.51082562 1.91629073 1.91629073
 1.51082562 1.91629073 1.22314355 1.51082562 1.91629073 1.22314355
 1.22314355 1.51082562 1.22314355 1.22314355 1.91629073 1.91629073
 1.91629073]
```

In [9]:
```
df = pd.DataFrame(vector.todense(), columns=tf_idf_vectorizer.get_feature_names_out
df
```

Out[9]:

| | 09 | 100 | 112 | 12 | 13 | 15 | 17 | 18 | 191 |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.000000 | 0.000000 | 0.059892 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.059892 | 0.059892 |
| **1** | 0.050598 | 0.000000 | 0.000000 | 0.050598 | 0.050598 | 0.050598 | 0.050598 | 0.000000 | 0.000000 |
| **2** | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| **3** | 0.000000 | 0.069229 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |

4 rows × 325 columns