

Convert text to Numeric Vectors

1. Bag of Words
2. Binary BoW
3. TF-IDF ✓
4. Word2Vec

} n-grams

n=1 unigrams
n=2 bi-grams
n=3 tri-grams

crickets runs score ✓

→ D1 : I love pasta - Pasta is tasty ✓ → Bigram
D2 : I hate pasta - pasta is cheap x

global warming, glaciers → Trigrams

size of the vectors → No. of features

Vocab: list of unique words

Vocab: I love pasta hate is tasty cheap ? 7

D1 : 1 1 2 0 1 1 0
BBOW 1 1 1 0 1 1 1
D2 : 1 0 2 1 1 0 1

Vocab give

	x_1	x_2	x_3	...	x_7	...	x_{7800}	y
D1	1	1	2	...	0	+
D2	1	0	2	1	+

→ Classification / Clustering

TF-IDF

→ Term Frequency - Inverse Document Frequency

$$\text{Term Frequency (TF)} = \frac{\text{No. of times the word occurs in the doc.}}{\text{Total no. of words in the doc.}}$$

$$\text{Document Frequency (DF)} = \frac{\text{No. of documents in which the word occurs.}}{\text{Total no. of documents.}}$$

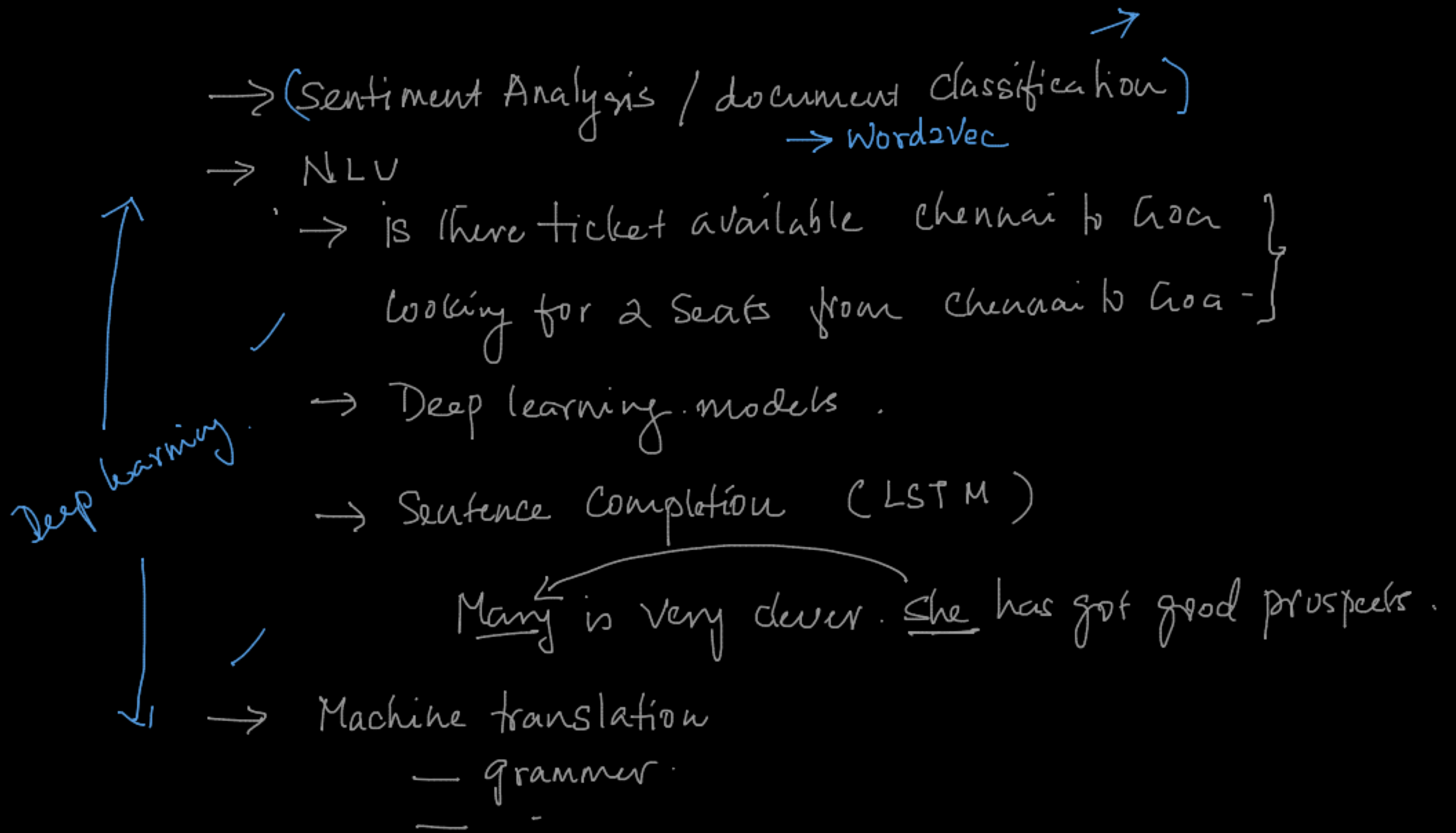
$$\text{IDF} = \frac{1}{\text{DF}}$$

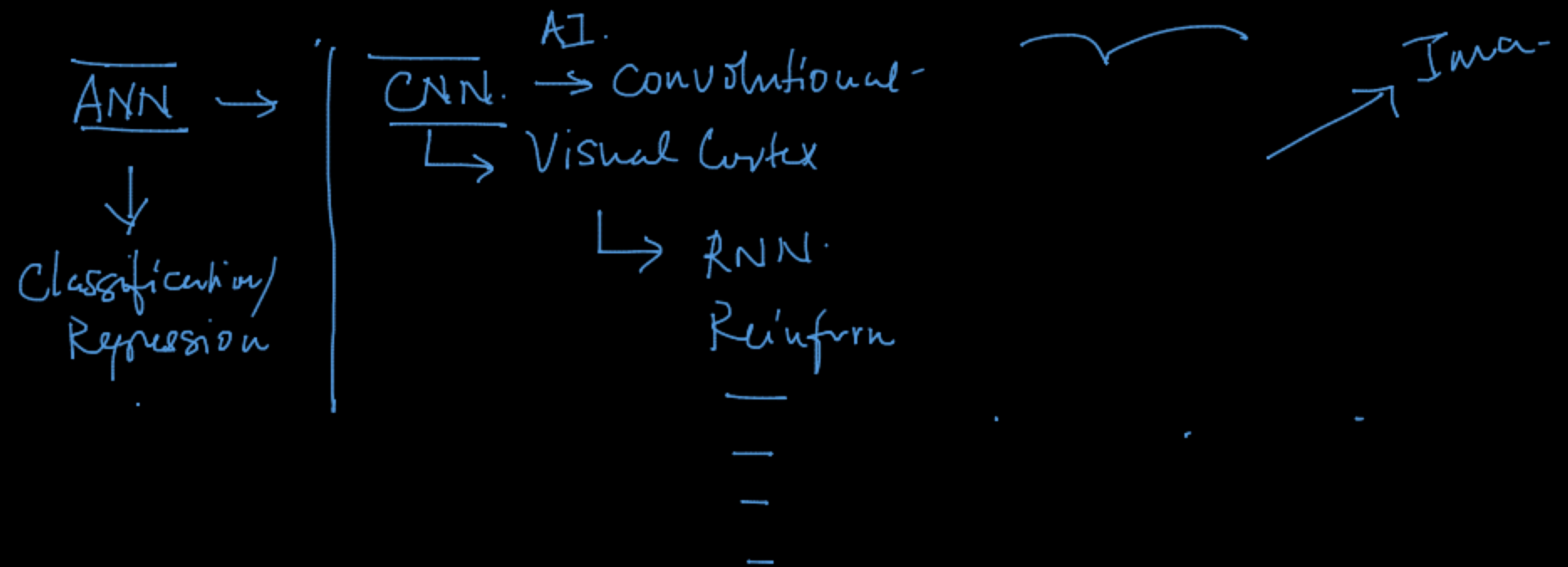
✓ D1: The car is driven on the road. ✓

✓ D2: The truck is driven on the highway

	Term Freq				log(IDF)	TF-IDF	
	D1	D2	DF	IDF		D1	D2
The	2/7	2/7	2/2	1	0	0	0
car	1/7	0	1/2	2	0.3	0.043	0
truck	0	1/7	1/2	2	0.3	0	0.043
is	1/7	1/7	2/2	1	0	0	0
driven	1/7	1/7	2/2	1	0	0	0
on	1/7	1/7	2/2	1	0	0	0
road	1/7	0	1/2	2	0.3	0.043	0
highway	0	1/7	1/2	2	0.3	0	0.043

	x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈	y
→ D1	0	0.043	0	0	0	0	0.043	0	car
D2	0	0	0.043	0	0	0	0	0.043	truck





512 x 512

1024 x 1024 \Rightarrow 10,000,000
 \Rightarrow 1 Million

1000

Text processing

Message	Label.
How r u?	Ham
Trip for two	Spam
⋮	⋮
⋮	⋮

$\text{Vect} = \text{CountVectorizer}()$
 $\Rightarrow \text{Vect.fit_transform}(X_{train})$

x_1	x_2	x_3	...	x_{7082}	Label.
⋮	⋮	⋮	⋮	⋮	⋮

$\text{clf} = \text{LinearSVC}()$
 $= \text{clf.fit}(\text{X_train}, \text{y_train})$
 $= \text{clf.predict}(\text{X_test})$

Pipeline $\Rightarrow \text{model} \cdot (\text{X_train})$

