M00 • SESSION 1 • GENAI 2026

Text to Numbers

How Machines Learn to Read

From raw characters to intelligent understanding.

BoW | TF-IDF | Word2Vec | GloVe | FastText

Foundations | 45 min | GenAI 2026

AGENDA

What We'll Cover Today

01 **The Core Problem**
Why computers only understand numbers — the fundamental constraint

02 **One-Hot Encoding**
Every word a unique slot — simple but semantics-free

03 **Bag of Words & TF-IDF**
Count-based representations powering real search engines

04 **Word Embeddings**
Word2Vec, GloVe, FastText — words become geometry

05 **Cosine Similarity**
Why we measure angle not distance — the universal NLP metric

06 **Live Demo: CineMatch**
1000 movies, 5 algorithms — compare recommendations live

THE CORE PROBLEM

Computers Understand Exactly One Thing

Not letters. Not words. Not sentences. Only numbers.

0 & 1

Binary at the Core

'A' = 01000001
Every character stored as binary.

Tk

Tokens in GPT-4

'ChatGPT' splits into tokens.
Each maps to a unique int ID.

1536

Dimensions

Each token becomes 1,536 floats capturing meaning and context.

?!

The Challenge

'Hello?' must become numbers. But which numbers, and how?

Journey: raw text → tokens → integer IDs → float vectors → meaning

ONE-HOT ENCODING

Every Word Gets One Unique Slot

Vocabulary of N words → N-dim vector with exactly one 1 and N−1 zeros.

VOCABULARY	cat	dog	mat	sat
"cat" →	1	0	0	0
"dog" →	0	1	0	0
"mat" →	0	0	1	0

Problems with One-Hot

- No semantics: 'cat' and 'kitten' equally far (cosine sim = 0)
- Huge sparse vectors: 50K vocab → 50K dims, ONE non-zero
- No relationships: king − man + woman ≠ queen

BAG OF WORDS

Shake the Book — Count What Falls Out

Order doesn't matter — only how often each word appears in the document.

"cat sat on mat"
→ {cat:1, sat:1, on:1, mat:1}

Sentence	cat	dog	sat	mat	crime
The cat sat on the mat	1	0	1	1	0
The dog sat on the mat	0	1	1	1	0
Crime drama: the godfather	0	0	0	0	1

Problem

"dog bites man" = "man bites dog"
Ignores order, grammar, and semantics. "good" and "not good" look identical.

Formula

V[word] = count of word in document
Then cosine similarity measures topic overlap between count vectors.

TF-IDF

Rare Words Matter More

Down-weight 'the', 'a', 'is' — up-weight 'redemption', 'heist', 'godfather'.

TF (Term Frequency)

TF(t,d) = count of t in d / total words in d
How often does this word appear in THIS doc?

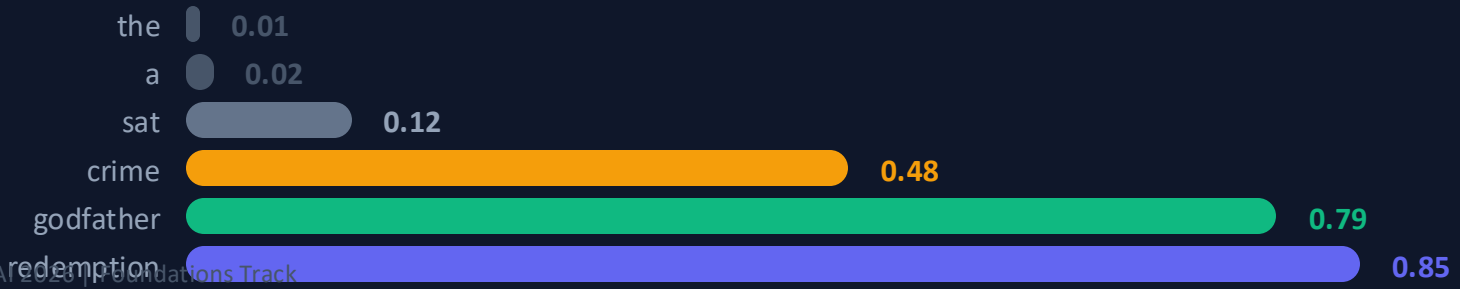
IDF (Inverse Document Frequency)

IDF(t) = log(N / df(t))
Higher score if the word is RARE across all docs.

TF-IDF = TF x IDF

At a party, everyone says "hello" → low IDF, useless. Only one person says "cryptocurrency" → high IDF, very specific!

TF-IDF weights for the Godfather document:



WORD2VEC

Words Learn Meaning from Their Neighbors

"You shall know a word by the company it keeps." — J.R. Firth, 1957

CBOW (Continuous Bag of Words)

Predict the CENTER word from surrounding context.

[The, ___, his, crown] → 'king'

+ Faster training
+ Better for frequent words
+ Smoother embeddings

Skip-gram

Predict surrounding CONTEXT from the center word.

'king' → [The, removed, his, crown]

+ Better for rare words
+ Works on small datasets
+ Used by default in FastText

The Famous Analogy — Words are Points in Space

vector(king) - vector(man) + vector(woman) = vector(queen)

Relationships = directions in 100–300 dimensional space.

GLOVE & FASTTEXT

Standing on Word2Vec's Shoulders

GloVe (2014, Stanford)

Global Vectors for Word Representation

Word2Vec only sees LOCAL windows. GloVe also uses GLOBAL statistics: how often two words co-occur across the ENTIRE corpus.

u.v = log P(word_i | word_j)

+ Pre-trained on 6B Wikipedia words
+ Stable, rich, widely-used embeddings
- One vector per word ('bank' is ambiguous)

FastText (2016, Meta AI)

Subword Character N-Grams

'playing' → character n-grams: pla, lay, ayi, yin, ing, play, ...

word vector = SUM of n-gram vectors

+ Handles typos: 'recieve' still works!
+ OOV words built from character pieces
+ Great for morphology-rich languages
- Larger model; subword noise possible

Both still give ONE static vector per word. Transformers fix this (next session).

COSINE SIMILARITY

Why Angle — Not Distance?

✗ Problem: Euclidean Distance

Doc A mentions 'crime' 4 times = [4]
Doc B mentions 'crime' 1 time = [1]
Distance = 3.0 → WRONG: says "different"

✓ Solution: Cosine Similarity

Measures the ANGLE between vectors.
Length doesn't matter — direction is everything.
cos(0°) = 1.0 → CORRECT: same topic!

$$\cos(\theta) = (A \cdot B) / (||A|| \times ||B||)$$ Range: -1.0 to +1.0

1.00	0.80	0.50	0.00	-1.00
Identical	Very Similar	Related	Unrelated	Opposite

Dividing by ||A|| × ||B|| removes the effect of document length.

COMPARISON TABLE

All 5 Methods — Side by Side

Method	Year	Vector	Semantics?	OOV?	Best Use
Bag of Words	1954	Sparse	No	No	Simple classification
TF-IDF	1972	Sparse	No	No	Search & ranking
Word2Vec	2013	Dense (100-300d)	Word-level	No	NLP analogies
GloVe	2014	Dense (50-300d)	Word-level	No	Transfer learning
FastText	2016	Dense (100-300d)	Word-level	YES!	Multilingual

All methods: text → numbers → cosine similarity
The constant that never changes: convert, compare, decide.

LIVE DEMO

CineMatch — Same Movie, 5 Algorithms

1000 IMDB movies, pick any film, watch five methods disagree (and agree).

Pipeline: movie text (plot + genre + director + cast) → embedding → 1000×1000 similarity matrix → top-10

BoW	TF-IDF	Word2Vec	GloVe	FastText
Finds movies sharing exact words: character names, locations.	Better genre/director matches; rare keywords dominate the score.	Semantic matches — thematically similar with different wording.	Rich pre-trained knowledge from 6B Wikipedia tokens.	Handles unusual names, foreign words, and partial matches.

Try 'The Dark Knight' — which algorithm finds Inception?
Which finds Batman Begins? Which finds The Prestige?

SESSION OVERVIEW

What We Covered

1

One-Hot Encoding

Every word a unique slot — simple but zero semantics

2

Bag of Words

Count word occurrences — fast and surprisingly effective

3

TF-IDF

Rare words weighted higher — backbone of real search engines

4

Word2Vec

Neural network learns: king - man + woman = queen

5

GloVe & FastText

Pre-trained global stats and subword n-grams for OOV words

6

Cosine Similarity

Measure angle not length — the universal text similarity metric

SESSION COMPLETE

You Now Know

How Machines Read

What's MISSING?

These vectors are great — but which words matter most in a long sentence?
How does the model know 'bank' means finance here and river there?
→ We need ATTENTION.

Next Session → Attention is All You Need | How Transformers Work

✓ One-Hot | ✓ Bag of Words | ✓ TF-IDF | ✓ Word2Vec | ✓ GloVe & FastText | ✓ Cosine Sim

Medium: nursnaaz

github.com/nursnaaz