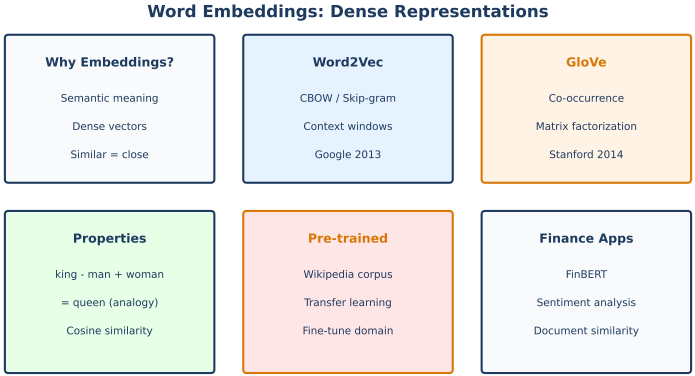
Lesson 39 Summary: Word Embeddings

Data Science with Python – Key Concepts

Data Science Program

Word Embeddings Overview



Dense vectors capture semantic meaning

Why Embeddings?

Limitations of BOW/TF-IDF:

- **Sparse:** High-dimensional, mostly zeros
- **No semantics:** "good" and "great" unrelated
- **Solution:** Dense, low-dimensional vectors

Similar words have similar vectors

Word2Vec

Learn from context:

- **CBOW:** Predict word from context
- **Skip-gram:** Predict context from word
- **Result:** 100-300 dim vectors

Google, 2013 - still widely used

Semantic Arithmetic

Famous analogy:

- **king - man + woman = queen**
- **Paris - France + Germany = Berlin**
- **Similarity:** Cosine distance

Embeddings encode relationships

Pre-trained Models

Transfer learning for NLP:

- **GloVe:** Wikipedia, Common Crawl
- **Word2Vec:** Google News
- **FastText:** Subword information

Fine-tune on domain-specific data

Quick Reference

Essential Commands:

Task	Code
Load GloVe	`gensim.downloader.load('glove-wiki')`
Get vector	`model['word']`
Similarity	`model.most_similar('word')`
Analogy	`model.most_similar(pos=[a,b], neg=[c])`

Embeddings are foundation for modern NLP