# [WORD EMBEDDINGS] -> it's a feature extraction method

Word representation : • 1-hot representation -> it doesn't generalize

• featurized = (word embeddings)

-> great for small training set

**Word Embeddings:** TRANSFER LEARNING

1. Learn word emb. from large text corpus (or download pre-trained WE)
2. Transfer WE to new task with smaller training set
3. Optional: continue to fine tune the WE with new data

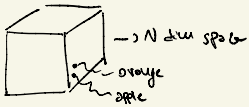
N-features

| | Man (5391) | Woman (9853) | King (4914) | Queen (7157) | Apple (456) | Orange (6257) |
|---|---|---|---|---|---|---|
| Gender | -1 | 1 | -0.95 | 0.97 | 0.00 | 0.01 |
| Royal | 0.01 | 0.02 | 0.93 | 0.95 | -0.01 | 0.00 |
| Age | 0.03 | 0.02 | 0.7 | 0.69 | 0.03 | -0.02 |
| Food | 0.04 | 0.01 | 0.02 | 0.01 | 0.95 | 0.97 |
| ... size, cost, alive | | | | | | |

(very similar -> Generalize better)

N-dimensional vector to represent the word MAN

-> it can be represented in a N dim space :




## Properties :

• Reasoning about analogies

| | Man (5391) | Woman (9853) | King (4914) | Queen (7157) | Apple (456) | Orange (6257) |
|---|---|---|---|---|---|---|
| Gender | −1 | 1 | -0.95 | 0.97 | 0.00 | 0.01 |
| Royal | 0.01 | 0.02 | 0.93 | 0.95 | -0.01 | 0.00 |
| Age | 0.03 | 0.02 | 0.70 | 0.69 | 0.03 | -0.02 |
| Food | 0.09 | 0.01 | 0.02 | 0.01 | 0.95 | 0.97 |

$e_{5391}$ $e_{man}$ $e_{woman}$

Man -> Woman as King -> ? Queen

$e_{man} - e_{woman} \approx e_{king} - e_{?}$

$e_{queen}$

$e_{man} - e_{woman} \approx [-2, 0, 0, 0]$

$e_{king} - e_{queen} \approx [-2, 0, 0, 0]$

## How to learn WE

The distributed representation is learned based on the usage of words. This allows words that are used in similar ways to result in having similar representations, naturally capturing their meaning. This can be contrasted with the crisp but fragile representation in a bag of words model where, unless explicitly managed, different words have different representations, regardless of how they are used.

• You can learn WE through an embedding layer placed in front of your NN, as part of your task (ie: doc classification)

• Word2Vec : statistical method to learn WE.

- CBOW vs SKIP-GRAM Model are 2 learning methods.
- Learn about words using content. The content is defined by a window of surrounding words. The window size is parameter of the model

• Glove (global vector for word representation):

- it doesn't use a window to define local content
- it builds a word-occurrence matrix using statistics across the whole text corpus

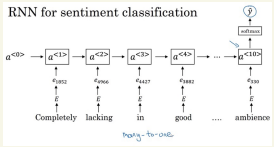
• Usage :

1) Learn an embedding for your problem :
   a) Stand alone: model trained to learn an embedding, which is saved and then used as part of another model for your task later
   b) Jointly : WE learned as part of a large task-specific model

2) Reuse embeddings (Pre trained embeddings):
   a) Static : embedding kept static, used as component of a model
   b) Updated: updated during training of your model (fine tuning)

# [SENTIMENT CLASSIFICATION]

Problem: small labelled dataset
Solution: word embeddings + RNN




# [Embedding layer output]

m = batch size / training set size

ml = max len (sentence max len) ⟶

fs = feature space (dimension of we)