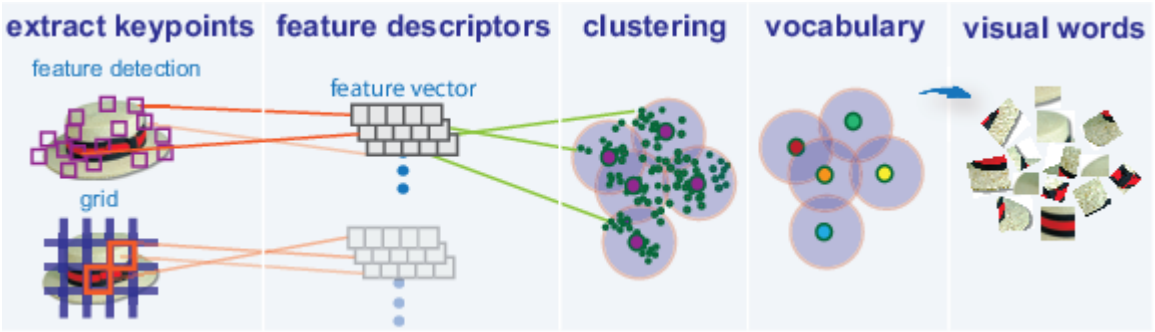
AIML Lecture 4 : Reading Material

1 Bag of Visual Words : Image Feature

A simple approach to classifying images is to treat them as a collection of regions, describing only the appearance of these local patches independently and ignoring their spatial structure. Similar models have been successfully used in the text community for analyzing documents and are known as "bag-of-words" (BoW) models, since each document is represented by a distribution over a fixed vocabulary. That is we can represent a text document as a vector with dimension as the number of words in the vocabulary. Each word in the vocabulary is assigned an index by lexicographic ordering. The value of the vector at an index for a word is just the count of the word occurrences.



To represent an image using the BoW model, an image can be treated as a document. Similarly, "words" in images need to be defined too. To achieve this, it usually includes following three steps: feature detection, feature description, and codebook generation. Feature detection involves, finding edges, corners and other regions of interest in the image. After feature detection, each image is abstracted by several local patches. Feature representation methods deal with how to represent the patches as numerical vectors. These vectors are called feature descriptors. A good descriptor should have the ability to handle intensity, rotation, scale and affine variations to some extent. One of the most famous descriptors is Scale-invariant feature transform (SIFT). SIFT converts each patch to 128-dimensional vector. After this step, each image is a collection of vectors of the same dimension (128 for SIFT), where the order of different vectors is of no importance.

The final step for the BoW model is to convert vector-represented patches to "codewords" (analogous to words in text documents), which also produces a "codebook" (analogy to a word dictionary). A codeword can be considered as a representative of several similar patches. One simple method is performing k-means clustering over all the vectors. Codewords are then defined as the centers of the learned clusters. The number of the clusters is the codebook size (analogous to the size of the word dictionary). Thus, each patch in an image is mapped to a certain codeword through the clustering process and the image can

be represented by the histogram of the codewords. THe BoW representation for images is often called Bag of Visual Words or *BoVW*.

2 Word2Vec: Text Feature

A simple feature representation for a word is 1-of-N (or 'one-hot') encoding every element in the vector is associated with a word in the vocabulary. The encoding of a given word is simply the vector in which the corresponding element is set to one, and all other elements are zero.

Suppose our vocabulary has only five words: King, Queen, Man, Woman, and Child. We could encode the word 'Queen' as the figure on left.

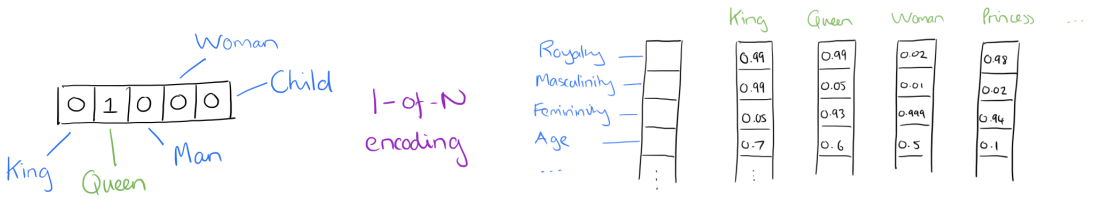


Figure 1: 1-of-N encoding on left. Word2Vec on right.

Using such an encoding, there's no meaningful comparison we can make between word vectors other than equality testing.

In word2vec, a distributed representation of a word is used. Take a vector with several hundred dimensions (say 1000). Each word is representated by a distribution of weights across those elements. So instead of a one-to-one mapping between an element in the vector and a word, the representation of a word is spread across all of the elements in the vector, and each element in the vector contributes to the definition of many words. Such a vector comes to represent in some abstract way the 'meaning' of a word. And as we'll see next, simply by examining a large corpus it's possible to learn word vectors that are able to capture the relationships between words in a surprisingly expressive way.

We find that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way. Specifically, the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship. For example, if we denote the vector for word i as xi, and focus on the singular/plural relation, we observe that

$$x_{\text{apple}} - x_{\text{apples}} \approx x_{\text{car}} - x_{\text{cars}}, \qquad x_{\text{family}} - x_{\text{families}} \approx x_{\text{car}} - x_{\text{cars}},$$

and so on.

3 MFCC : Audio Feature

The most commonly used feature extraction method in automatic speech recognition (ASR) is Mel-Frequency Cepstral Coefficients (MFCC). This feature extraction method was first

mentioned by Bridle and Brown in 1974 and further developed by Mermelstein in 1976 and is based on experiments of the human misconception of words.

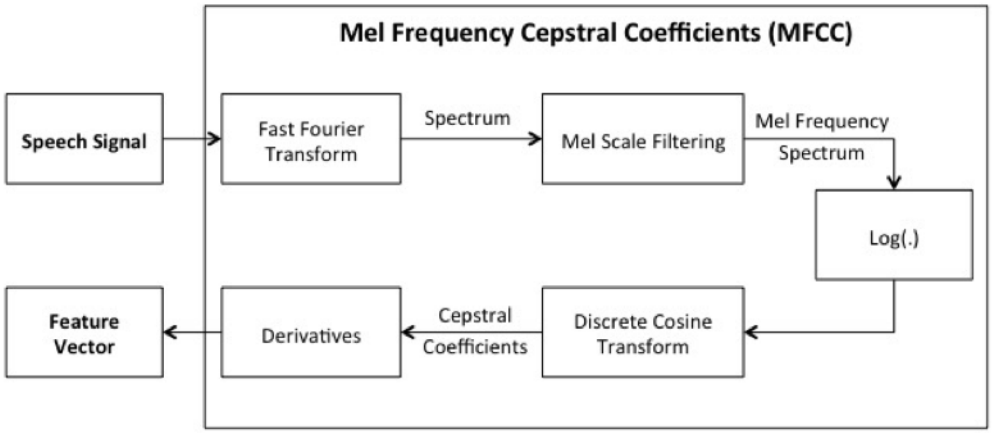


Figure 2: Flowchart with the various stages of computation for getting the MFCC features.

To extract a feature vector containing all information about the linguistic message, MFCC mimics some parts of the human speech production and speech perception. MFCC mimics the logarithmic perception of loudness and pitch of human auditory system and tries to eliminate speaker dependent characteristics by excluding the fundamental frequency and their harmonics. To represent the dynamic nature of speech the MFCC also includes the change of the feature vector over time as part of the feature vector .

4 References

1. Bag of Visual Words
   lecture slides : http://www.robots.ox.ac.uk/~az/icvss08_az_bow.pdf code with explanations:
   http://people.csail.mit.edu/fergus/iccv2005/bagwords.html
   https://kushalvyas.github.io/BOV.html

2. MFCC Audio Feature
   http://www.speech.cs.cmu.edu/15-492/slides/03_mfcc.pdf
   http://recognize-speech.com/feature-extraction/mfcc
   http://www.practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequen

3. Word2Vec
   Article describing many text features: https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/
   https://www.deeplearningweekly.com/blog/demystifying-word2vec
   https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/
   A technical perspective: https://arxiv.org/pdf/1411.2738v3.pdf