**Natural Language Processing (NLP)**

**Session 1**

1. **NLP**: Concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to ‘understand’ its full meaning, complete with the speaker or writer’s intent and sentiment.  
     
   Applications include managing big data i.e., Google Search and analysing social media content, and dialogue systems/chatbots (spoken and typed interfaces).

Read up on Google Translate statistics.

1. **Managing Big Data**:
   1. *Classification*: identify relevant content / quickly assess this content.
   2. *Extraction*: structured information from unstructured textual data.
   3. *Summarize*: compressing the full text into smaller readable summaries either from a single source (single-document summarization) or from a collection of articles (multi-document summarization). Main approaches:
      1. *Extractive*: Select key sentences/phrases for summary.
      2. *Abstractive*: Re-generate a summary based on the meaning of the text.
2. **Sentiment Analysis**: Sentiment analysis is the use of natural language processing (NLP), machine learning, and other data analysis techniques to analyse and derive objective quantitative results from raw text.
   1. *Dictionaries* *method*: Build a dictionary of positive/negative words. Limitations include ambiguity of the text.
   2. *Data-driven method*: Build a model that learns positive and negative words:
      1. *Pre-processing*: Sentence segmentation, work tokenization, normalisation
      2. *Zipf’s Law*: The frequency of any word is approximately inversely proportional to its rank in the frequency table.  
           
         Most of the relevant words can be captured using the *frequent words* method.
   3. *Words*: The simplest way to model interaction between words in texts is characterising a text in terms of the words contained in it, where a word can be considered a **feature** of a text.
   4. *Sentences*:
      1. *Sequence modelling*: tokenizes by pairs of adjacent words rather than by individual ones i.e., not good, very bad, etc. Recurrent Neural Networks work well with this methodology.
      2. *Use of generative grammar*: Encode the sentence with specific rules in line with generative grammar. Helps with syntactic ambiguity.
   5. *Distributional Hypothesis*:
      1. *Words*: Words that occur in similar contexts tend to have similar meanings. Characterise words/sentences as vectors on a vector space i.e., the word is defined as a point on a 2D/3D graph by associating it with the context of its occurrences with other dimensions (words). For example, if we have a graph that has grave on the y-axis, blood on the x-axis, and dead on the z-axis, the characterisation of the word “vampire” can be found. The closeness of the words (vectors) can be derived using distributional semantic (geometric method).
      2. *Sentences*: The meaning of a sentence is thus related to the distribution of the sentences around it. However, the problem is in the sparsity of sentences the steepness of the curve of the Zipfian distribution of sentences is far steeper than that for words. Can use simple vector operations i.e., addition, dot product, etc.
3. **Dialogue Systems**: Breaks things down to *intents* and *entities* and context variables.