Paper Review - From Frequency to Meaning: Vector Space Models of Semantics

Abhirut Gupta

April 5, 2018

Abstract

Survey of Vector Space Models - models for understanding human language. Categorized into three kinds based on the type of matrix used for representation -

- Term-document models - Rows are terms and columns are documents
- Word-context models - Rows are words, and columns are context like phrases, sentences and document (term-document models are a special case)
- Pair-pattern models - Rows are pairs of words and columns are patterns they appear together in

The values in the matrices are frequency statistics gathered from free text. This is used to differentiate from models where values are not frequency statistics and are not treated as VSMs in this paper. The paper presents an overview of these models, an overview of linguistic and mathematical processing that is done to create them from text, an example open-source software for each model, and the broad applications of each of these models. These matrices are an efforts to implement the abstract *distributional hypothesis* - words that occur in similar contexts, have similar meanings. The broad concept is known as *statistical semantics hypothesis* - statistical patterns of human words can be used to figure out what they mean.

1 Models

1.1 The Term-Document Matrix

The rows represent terms (which are usually words) and the columns represent documents. The document vector (column of the matrix) represents the document as a bag of words, and in some sense represents the meaning of the document. The sequential order of words, structure of phrases, and paragraphs is lost, but it has been found to work well in IR. An intuitive explanation is that the topic of the document will influence the author's choice of words. Two documents which are about the same topic, will probably have similar pattern of numbers in the vector.

1.2 The Word-Context Matrix

To measure similarity of words, document is not necessarily the optimal length. The columns of these matrices are the *context* in which a word appears, where the context can be words, phrases, sentences, paragraphs, chapters, or documents. The context can also be sequence of characters of patterns, or grammatical dependencies (Sahlgren?s (2006) thesis). Firth (1957, p. 11) said, "You shall know a word by the company it keeps."

1.3 The Pair-Pattern Matrix

Similarity of relations can be measure with a pair-pattern matrix. Rows correspond to pair of words, and columns correspond to the patterns that represent these pair of words in text. *extended distributional hypothesis* - patterns that frequently co-occur with similar pair of words tend to have similar meanings. *latent relation hypothesis* Pair of words that co-occur in similar patterns tend to have similar semantic relations.

Similarities

Attributional Similarity between two words $sim_a(a, b) \in \mathbb{R}$ (from the word-context matrix) is based on the similarity of their properties. *Relational Similarity* between pair of words $sim_r(a : b, c : d) \in \mathbb{R}$ (from the pair-pattern matrix) is the similarity between the relations $a : b$ and $c : d$. While it might be tempting to reduce the relational similarity in terms of attributional similarity as follows $sim_r(a : b, c : d) = sim_a(a, c) + sim_a(b + d)$, it's not really correct. Consider three pairs of words which are in similar relations $a : b$, $c : d$, and $e : f$, while the attribute similarity between a, c, and e might be high also between b, d, and f might be high, we cannot infer that $a : d$ and $c : f$ are in similar relations. In computational linguistics, the term *semantic relatedness* is used to convey *attributional similarity*, *semantic similarity* is used to refer to words that share a hypernym (a car and a bicycle are *semantically similar*), and *semantically associated* if they co-occur frequently.

1.4 Other Modelss

Higher order tensors are also used to represent word similarities. An example is the word-word-pattern tensor used in Turney (2007).

2 Liguistic Processing for Vector Space Models

Input is assumed to be free text

1. **Tokenization** - Split text into tokens taking care of punctuations, multiword terms, remove stop words. Harder for languages with no space between tokens (Chienese) - use lexicon, but still might not result in unique tokenization.

2. **Normalization** - Different surface forms of same words. Case folding and stemming are common operations. Operations easier in English but might be a problem in other languages. Increases recall for IR, decreases precision.

3. **Annotation** - Same surface form of words might have different meaning based on context (verbs and nouns in English or homonyms). POS tagging, word sense tagging, parsing (tagging roles to words. Reduces recall, increases precision.

3 Mathematical Processing for Vector Space Models

The four broad steps are - generate matrix of frequencies, adjust weights of elements in the matrix (common words have high frequencies but less information than rare words), reduce dimensionality (sparse matrix), calculate similarities. Lowe (2001) gives a good summary of mathematical processing for word?context VSMs.

3.1 Building the Frequency Matrix

Conceptually similar, but engineering challenges on a large corpus. One scan to store events (a word and it's context is one event) and their frequencies in a hash-table, database or a search index. Then use the resulting structure to create the matrix in sparse representation.

3.2 Weighting the Elements

A surprising event has higher information content than an expected event (Shannon, 1948). Use tf-idfs, length normalization (in absence of which search engines prefer longer documents), term weighting to correct for co-related terms, feature selection (some terms get weight of 0 and are effectively removed from the matrix). Pointwise Mutual Information (PMI) is an alternative to tf-idf. Positive Pointwise Mutual Information is often found to work better. It is a measure of how much information the occurrence of one event gives about the other. For independent events PMI is 0. PPMI returns 0 for negative PMI values.

Let **F** be a word context frequency matrix with $\mathbf{F} \in \mathbb{R}^{n_r \times n_c}$. i^{th} row of **F** is the row vector $f_{i:}$ and corresponds to the word w_i, and the j^{th} column of **F** i the column vector $f_{:j}$ and corresponds to the context c_j. f_{ij} is the number of times w_i appears in context c_j

$$p_{ij} = \frac{f_{ij}}{\sum_{a=1}^{n_r} \sum_{b=1}^{n_c} f_{ab}}$$

$$p_{i*} = \frac{\sum_{b=1}^{n_c} f_{ib}}{\sum_{a=1}^{n_r} \sum_{b=1}^{n_c} f_{ab}} \qquad p_{*j} = \frac{\sum_{a=1}^{n_r} f_{aj}}{\sum_{a=1}^{n_r} \sum_{b=1}^{n_c} f_{ab}}$$

$$pmi_{ij} = log(\frac{p_{ij}}{p_{i*} \times p_{*j}})$$

$$\begin{aligned} ppmi_{ij} &= pmi_{ij} \text{ if } pmi_{ij} > 0 \\ &= 0 \text{ otherwise} \end{aligned}$$