Decomposing Similarity and Composing Indpendance

ABSTRACT

Compositional Distributional Semantics (CDS) and Semantic IR (SIR) both work with vectors as semantic representations of terms. In CDS these terms are composed to get semantic representations for phrases, clauses, and sentences of language. In SIR, they are composed to obtain semantic representations for queries and documents. Each approach has its own methods of building the original term vectors and its own methods of composing them.
To evaluate the models, in SIR one computes a retrieval status value (RSV) based on the probabilistic dependance between the representations, whereas in CDS the similarity between the representation is a major measure, this is computed using geometric distances between the representations.
In this paper we show that despite the apparent differences in the methodologies, certain equivalences between the geometric and probabilistic methods can be established.
In particular, we show how and when 1) the bag of word vectors of IR and the entity-relationship vectors of semantic SIR become equivalent to the co-occurrence vectors of CDS, 2) the phrase-based TF-IDF formulae of SIR become equivalent to the vector composition operators of CDS, and 3) the cosine similarity measure of CDS can be decomposed and proven to be equivalent the SIR measure of relevance.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms

Algorithms; Theory

Keywords

NPL, Subject-Verb-Object (SVO), Semantic IR

1. INTRODUCTION & MOTIVATIONS

Depending on the co-occurrence quantification, and depending on the composition for s-v-o, for some expressions, one gets equivalences between geometric and probabilistic (and information-theoretic) expressions.
[**?**]:

$$\vec{d} \cdot \vec{q}$$

$$\vec{d} = \begin{pmatrix} P(t_1|d) \\ P(t_2|d) \\ P(t_3|d) \end{pmatrix}, \qquad \vec{q} = \begin{pmatrix} P(q|t_1) \\ P(q|t_2) \\ P(q|t_3) \end{pmatrix}$$

2. RATHER OBVIOUS I SUPPOSE

2.1 Single Terms

In IR, the main measure of retrieval is TF-IDF. In its most basic form, given a term t, a document d, and a set of documents D, this is the frequency of t in d, multiplied by the logarithm of the total number of documents N divided by the number of documents with t in them.

$$TF(t,d) \cdot IDF(t,D) = freq(t,d) \cdot \log \frac{N}{\mid \{d \in D \mid t \in d\} \mid} \tag{1}$$

In DS, given is a corpus of text in which one works with the frequency of co-occurrence of terms with a set of features, in a window of length k. Given a set of m features $F = \{f_1, f_2, \cdots, f_m\}$, the space is the vector space with F as its basis, that is $V = \{f_i\}_i$. The vector representation of a term t is a linear combination of the basis:

$$\overrightarrow{t} = \sum_i C_i \overrightarrow{f}_i \tag{2}$$

The coordinate C_i over the basis vector $\overrightarrow{f}_i$ is a normalised function of the frequency of co-occurrence of t with f_i in the window of length k. Denoting the latter by $g(freq_k(t, f_i))$, we obtain

$$C_i = g(freq_k(t, f_i)) \tag{3}$$

Forgetting about the i indices for a minute, note that

$$freq_k(t,f) = \frac{\sum_f N(t,f)}{k} \tag{4}$$

for $N(t,f)$ the number of times t and f occurred k words close to each other. So in DS, one works with the frequency of co-occurrence of terms with features rather than with documents. Hence, if one replaces the d's in equation **??** with f's, one obtains:

$$freq_k(t,f) \cdot \log \frac{m}{\mid \{f \in F \mid N(t,f) \neq 0\} \mid} \tag{5}$$

With the right choice of event spaces, one can establish

$$\begin{aligned} P(t) &= \frac{\sum_f N(f,t)}{kL} = \frac{freq_k(t,f)}{L} \quad (6) \\ P(t|f) &= \frac{\mid \{f \in F \mid N(t,f) \neq 0\} \mid}{m} \quad (7) \end{aligned}$$

for L the number of times t occurred in the corpus as a whole. Hence one can rewrite equation **??** as

$$freq_k(t,f) \cdot \log \frac{1}{P(t|f)} \quad (8)$$

This is the same as saying that the normalisation function g in DS is a multiplication by $\log \frac{1}{P(t|f)}$:

$$g(freq_k(t,f)) = freq_k(t,f) \cdot \log \frac{1}{P(t|f)}$$

This is indeed an IDF value in which features are seen as documents, we refer to this by 'TF-IFF', where 'IFF' is for Inverse Feature Frequency. Herein, one obtains a term vector as follows:

$$\overrightarrow{t} = \sum_i freq_k(t,f_i) \cdot \log \frac{1}{P(t|f_i)} \overrightarrow{f}_i \quad (9)$$

It is surprising however, that this is not a familiar normalisation scheme in DS. The most known and worked-with of the latter are Conditional Probability: $\mathrm{P}_k(f|t)$, Likelihood Ratio: $\mathrm{LR}_k(f,t) = \frac{P(f|t)}{P(f)}$, and PMI_k: $PMI_k(f,t) = \log \frac{P(f|t)}{P(f)}$. One simple way to relate these to IR is to say that g is the the multiplication of any of these with the same IFF as above.

From these, the most relevant to IR seems to be PMI. This resembles the Document Term Independence (DTI) measure, where for a single term t we have:

$$\mathrm{DTI}(d,t) := \log \frac{P(d|t)}{P(d)} = \log \frac{P(t|d)}{P(t)} \quad (10)$$

Here $\frac{1}{P(d)}$ is the IDF and $P(d|t)$ is the TF.

Other normalisation schemes such as conditional probability, LMI, BM25 etc ???

2.2 Phrases

In CDS

In IR

$$\begin{aligned} \mathrm{RSV}_{\text{TF-IDF}}(d,q) &:= \sum_{t \in q} \mathrm{score}(t,d,q) \quad (11) \\ \mathrm{score}(t,d,q) &:= \mathrm{TF}(t,d) \cdot \mathrm{IDF}(t) \quad (12) \end{aligned}$$

3. WHY THE INDEPENDENCE MEASURE IS BEST

Multiplication of three vectors: s, v and o.

$$\frac{P(c|s) \cdot P(c|v) \cdot P(c|o)}{P(c) \cdot P(c) \cdot P(c)} = \frac{P(s|c) \cdot P(v|c) \cdot P(o|c)}{P(s) \cdot P(v) \cdot P(o)}$$

Let t_i and t_j be two propositions (e.g. s-v-o).

$\mathrm{sim}(t_i, t_j)$ is high if

1. many features occur within the context of subject, verb, and object
2. the features are rare

cos: corresponds to sum over independence measure. Therefore, pmi is the correct model. Corresponds to

$$\sum_c \log(...) = \log \prod_c (...)$$

Does not work for different spaces for nouns and verbs?

The multiplication yields a probability for the subj-verb-obj sequence where all components are independent of each other.

Multiplication of two conditional probs means then what? Effect of L2 norm?

Relationship to LM - there should be an obvious one.

4. DECOMPOSING SIMILARITY

Decompsoed additive baseline similarity

$$\frac{cos(sbj1, sbj2) + cos(vrb1, vrb2)) + cos(obj1, obj2)}{3}$$

Can any of the NLP sentence similarities decompose? It does not seem so, for example

$$\begin{aligned} cos(sbj1 + verb1, sbj2 + vrb2) &\neq cos(sbj1, sbj2) + cos(vrb1, vrb2) \\ cos(sbj1 \odot verb1, sbj2 + vrb2) &\neq cos(sbj1, sbj2) \odot cos(vrb1, vrb2) \end{aligned}$$

For kronecker composition has to move between vector of different tensor rank, i.e. either from V to $V \otimes V$ or from the latter to the former. Diagonalization might be helpful here, also convolution kernel might help.

In general, may be the sentence cosines, do doecompsoe but not in a direct way as above, that is when left and right have the same operation. The operations might vary from lef to right, or the ratios might be preserved etc.

5. COSINES AGAIN

$$\begin{aligned} \cos(angle(s \odot v \odot o, ...) &= 1/\sum P(s,v,o|c) \cdot P(c) \quad (13) \\ \cos(angle(s + v + o, ...) &= \sum (P(s|c) \cdot ... + P(v|c)... + ...) \cdot .... \quad (14) \\ \cos(angle(s + v + o, ...) &= \sum (\log \Pi_{xins,v,o}(P(x,c)/P(x) \cdot P(c)) \quad (15) \\ & \quad (16) \end{aligned}$$

6. VARIOUS LOG MEASURES

$$\begin{aligned} &\text{NLogP} & N(t,c) \cdot \log 1P(c) \quad (17) \\ &\text{LogNLogP} & log(N(t,c)+1) \cdot \log 1P(c) \quad (18) \\ &\text{PExpN} & P(c)^{-N(t,c)} \quad (19) \\ &\text{PExpLogN} & P(c)^{-\log N(t,c)+1} \quad (20) \\ & & (21) \end{aligned}$$

	NLP	IR
co-occurrence	co-occurrence between the semantic symbol (target word) and feature words	co-occurrence between the semantic symbols (words) themselves
representation of words	distributional	symbolic
single vs set	similarity between two *single phrases*	relationship (implication, entailment) between two *sets of phrases*
symmetric yes/no	similarity is a symmetric function	relationship between sets is not symmetric; moreover, the phrase-based score is not necessarily symmetric
similar/relevant	phrase t_i is similar to phrase t_j	document d (source) is relevant with respect to query q (target)
scores	the similarity score is estimated based on the distance/angle between distributional vectors	the relevance score is estimated based on the retrieval model that computes the implication between the set of document and the set of query propositions
probabilistic semantics	(in)dependence between target word and feature word: $\frac{P(w_t,w_f)}{P(w_t)\cdot P(w_f)}$	(in)dependent between document and query: $\frac{P(d,q)}{P(d)\cdot P(q)}$
	for this work: virtual query has exactly one proposition; virtual document has exactly one condition; therefore, the similarity score $\mathrm{sim}(\text{phrase1}, \text{phrase2})$ can be compared to the retrieval score $\mathrm{RSV}(\text{document: set of phrases, query: set of phrases})$. TODO: how to get this over two columns	