

A note on application of Tensor Space Model for semantic problems

Prashant Singh¹, P.K.Mishra²

¹(Department of Computer Science, Banaras Hindu University, India)

²(Department of Computer Science, Banaras Hindu University, India)

Abstract: A Model is an abstract representation of system that allows the investigation of properties of the system and in some cases prediction of future outcomes We have seen many mathematical models but Tensor Space Model is area of interest for new researchers. It is an effective tool for notational convenience and memory optimization. In this paper I have to tried to prognosticate applicability of some Tensor Space Model for simple and higher dimensional problems and propose a model that can account for meaning of a word with respect to another word in a given text.

Keywords : Vector Space Model, Tensor Space Model(TSM)

I. INTRODUCTION

Tensor is a concept of multi-linear algebra that is an extension of vector which is concept of linear algebra. It can be expressed as sequence of values represented by a function with a tuple valued domain and scalar valued range. Tensors, defined mathematically, are simply arrays of numbers, or functions that transform according to certain rules under a change of coordinates. In physics, tensors demonstrates the properties of a physical system. The most important reason for its application is that “Tensor is independent of frame of reference”.

1.1 SEMANTIC SPACES

Semantic Spaces are popular framework for representation of word meaning as higher dimensional vectors, which can provide a robust model for semantic similarity. Natural language processing is a combined field of computer science and linguistic. Sentences are treated as information vector and language as vector space. A basic set of words can be defined as a linear combination vectors that span a given set of words with operations that are translatable to operands in language and, the original set can be expanded by simply translating definitions composed of only words from original set into some string of operations on basis vectors corresponding to these words .If we talk in terms of linear transformation, then that would be from the language space to information space.

Meaning of a given word depends upon context it appears. Same word may have different meaning and also different words have same meaning depending upon the set of other words with it. As an example we can word *bark* which has two meanings in different context *bark* (the sound of a dog) and *bark* (the skin of a tree). These words are called *homonym*. Heterographic examples include *to*, *too*, *two*, and *there*, *their*, *they're*.

TERM	MEANING	SPELLING	PRONUNCIATION
Homonym	Different	Same	Same
Homograph	Different	Same	Same or different
Homophone	Different	Same or different	Same
Heteronym	Different	Same	Different

Table 1

There are different approaches for solving problem of semantic spaces. One would be treating words as vectors another would be its extension into a space better to call it a vector space approach and my focus is its conversion into more lucrative approach a tensor approach.

1.2 ADDITION OF INFORMATION VECTORS

Nisan Haramati[3] suggested these methods which are very helpful in designing a model for similar problem.

1.2.1 FIRST METHOD

If we denote a sentence as an information vector then by using laws of vectors we can interpret different result. If we take one information vectors “Sachin is a cricketer” and “Cricketers are treated as God”. So by parallelogram law of vector we can interpret that “Sachin is treated like God”.

1.2.2 SECOND METHOD

Another approach of representation of information is by treating subject as a center of a circle and verbs as different radius and datives on circumferences. But for handling a lot of information's these methods are confusing.

1.2.3 THIRD METHOD

By using post script binary notation we can represent information vector .Despite not being able to produce any usable vector space structure from the analysis, it is not inconceivable that such a vector space exists, based on our own ability to communicate and understand one another.

1.3 WORD SENSE DISCRIMINATION

Word sense discrimination is an unsupervised learning clustering problem which seeks to discover which instance of words is used in same meaning. The goal of word sense discrimination is to group multiple instances of words into clusters, where each cluster represents a distinct meaning.

1.3.1 CLUSTERING

Vector Space clustering algorithm directly uses the vector representation of contexts as their input. However, similarity space algorithm requires a similarity matrix that produces the pair wise similarity between given context.

1.3.2 DETECTING SYNONYMY

This is very important task two words with same meaning are called synonyms and detecting it is very crucial.

1.3.3 CONTEXT REPRESENTATION

We maintain separation between the instances to be discriminated (i.e. test data) and data from which features are selected (i.e. training data).This allows to explore variations in the training data while maintaining a consistent test set ,and also avoid any limitation that might be caused by selecting features from test data when discriminating a small number of instances.

1.3.4 DISTRIBUTIONAL HYPOTHESIS

Linguists have faith in the theory that the context in which a word occurs determines its meaning. In word of FIRTH “You shall know a word by company it keeps”. The meaning of a word is defined by the way it is used. This leads to distributed hypothesis about word meaning .The context surrounding a given word provides information about its meaning. Words are similar if they share similar linguistic contexts. Semantic similarity can be considered as distributional similarity.

1.4 APPROACH OF SOLVING WORD SENSE PROBLEM

The meaning of a word x with respect to another word y could be a new vector z that is function of these two vectors. If we define a binary operation then vector z can be written as $z = x * y$. But this can be used only if we ignore syntactic relation.

It depends entirely upon manner of composition which plays decisive role .To make it more clear let us take an example if we talk about set theory, null set is a very common and basic term .The meaning of word “null” with another word “set” is same if we paraphrase word “set” with “void”, “empty” and “vacuous”. Here there is no problem of ambiguity. But if we take three words represented by three vectors a , b , c and the concatenation of these vectors d .

Let $a = \text{boy}$, $b = \text{plays}$, $c = \text{cricket}$. Then, vector $d = a * b * c$ which means cricket plays boy. But if we talk about predicate argument then you can't distinguish between boy plays cricket and cricket plays boy .So, operation plays a vital role.

1.5 DIFFERENT MEASURES OF FINDING RELATION OF WORDS WITH RESPECT TO EACH OTHER

1.5.1 CREATION OF COOCCURENCE MATRIX:

If we have a given text and each word in the text is described by a vector and we get a modified text of vectors. A co-occurrence matrix or co-occurrence distribution (less often co-occurrence matrix or co-occurrence

distribution) is a matrix or distribution that is defined over an image to be the distribution of co-occurring values at a given offset.

1.5.2 MEASURE OF DISTRIBUTED SIMILARITY

Compare word vectors using cosines

$$\text{Cos}(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} = \frac{\sum X_i Y_i}{\sqrt{\sum X_i^2} \sqrt{\sum Y_i^2}}, \text{ where } i \in \mathbb{N}.$$

Euclidean Distance of two vectors is:

$$\|X - Y\| = \sqrt{\sum (X_i - Y_i)^2}$$

The distance between two word vector determines how much they are related in a given text.

The meaning of words can be represented by vectors.

Problem occurs when two words have same meaning then the cooccurrence matrix will contain 1 at more than one place .So difficulty will occur in determining cosine angles and also in calculating distance

1.6 Limitation of Vector Space Model:

All context Words are equally important.

Distributional similarity is sometimes ignored.

All words within sample space are important while those which are in population are obsolete.

Syntactic dependencies are ignored.

1.6.1 Semantic Priming:

The processing of target word is facilitated by previous processing of semantically related word(the prime).Facilitation means faster and more accurate processing.e.g. Man primes living body while vechile does not prime living object. We can use Vector Space Model to simulate semantic priming but lexical relations can't be distinguished.

II. Tensor Space Model for meaning of a word in a context with another word in a given text

Tensor product-based models:

Smolensky (1990) uses tensor product to combine two word vectors a and b into a vector c representing the expression $a + b$.

Mitchell and Lapata framework:

They proposed a framework to represent the meaning of combination $p+a$ as function $F(p, a, R, K)$, where p and a are predicate and argument , R is the relation between them and K is the additional knowledge. This model is better called as predicate argument model.

Vector Space models :

These models are efficient for problems of low dimensions. For high dimensional problem these models are quite weak.

Proposed Tensor Space Model

Tensor space model is capable of dealing with all problems which have been discussed earlier. It considers documents as tensor instead of vector as we did earlier.

Let T be the tensor space (set of all possible tensors). Let R some set of relations.

Meaning of a word x can be expressed as three tuple relation $x=(x', R, R^{-1})$, where

X' represents a vector describing word x .

$\Phi: \Phi \rightarrow T$: describing x selection preference

$\Phi^{-1}: \Phi \rightarrow T$: describing inverse selection preference

Both Φ and Φ^{-1} are partial function.

Let $(a', \Phi_a, \Phi_a^{-1})$ and $(b', \Phi_b, \Phi_b^{-1})$ be representation of two words.

We define the meaning of a and b in context as a pair (a'', b'') of vectors where a'' is meaning of a wrt b and b'' meaning of b wrt a . Let Γ be the relation linking a to b .

$$a'' = (a' \otimes \Phi_b^{-1}(\Gamma), \Phi_a(\Gamma), \Phi_a^{-1})$$

$$b'' = (b' \otimes \Phi_a(\Gamma), \Phi_b, \Phi_b^{-1} - (\Gamma))$$

Let there be a paragraph p , a line $l \in p$ and let $a, b \in l$.

Writing in terms of predicate argument structure, where *belongsto*, *part of*, *meaning wrt* are the relations belonging to $\Psi \in R$.

$$\text{belongsto}(a,l) + \text{belongsto}(b,l) + \text{partof}(l,p) + \text{meaning wrt}(a,b) + \text{meaning wrt}(b,a).$$

Each proposition in tensor memory can be viewed as relation. Next step is memory probing.

$$T = \xi_{\text{belongsto}} \otimes \xi_a \otimes \xi_l + \xi_{\text{belongsto}} \otimes \xi_b \otimes \xi_l + \xi_{\text{partof}} \otimes \xi_l \otimes \xi_p + \xi_{\text{meaningwrt}} \otimes \xi_a \otimes \xi_b + \xi_{\text{meaningwrt}} \otimes \xi_b \otimes \xi_a$$

T is tensor memory. If we want to probe from memory then we will find following result.
 $\xi.T = \text{Rel}$, $\text{Rel} = \text{belongsto}(l) + \text{part of}(p) + \text{meaningwrt}(a) + \text{meaningwrt}(b)$.

We can redefine this predicate tensor to get a good mathematical form. Let τ_c represents predicate tensor for belongs to relation and similarly τ_p and τ_m represents predicate tensor for part of and meaning with respect to relation.

So tensor memory can be rewritten as

$$T = (\tau_b * \xi_a * \xi_l) + (\tau_b * \xi_b * \xi_l) + (\tau_p * \xi_l * \xi_p) + (\tau_m * \xi_a * \xi_l) + (\tau_m * \xi_b * \xi_l)$$

III GRAPHICAL REPRESENTATION OF TENSOR

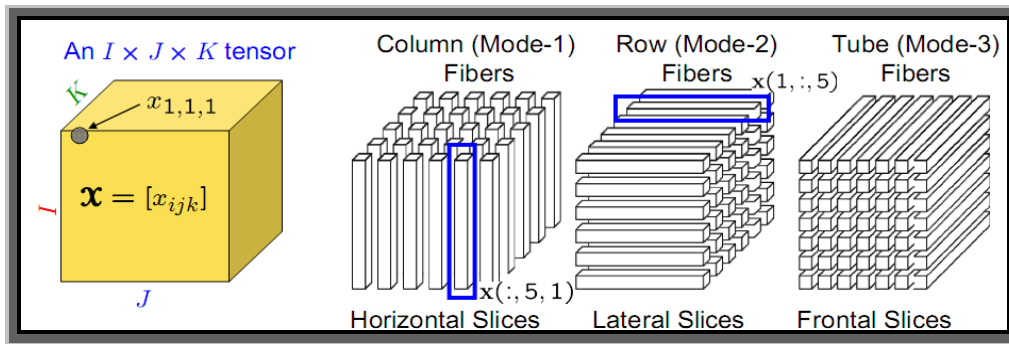


Figure 1

We can view it in terms of slices. Dimensionality is very vital issue. For higher order tensor we can't make graphs because we can not represent higher order object in two or three dimension.

Conversion from tensor to matrix and matrix to tensor can be done easily.

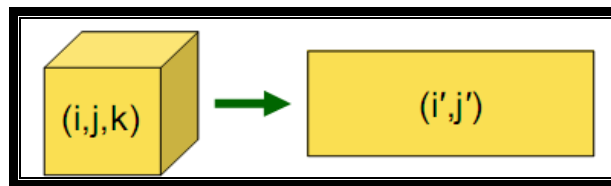


Figure 2

Tensor Multiplication to matrix:

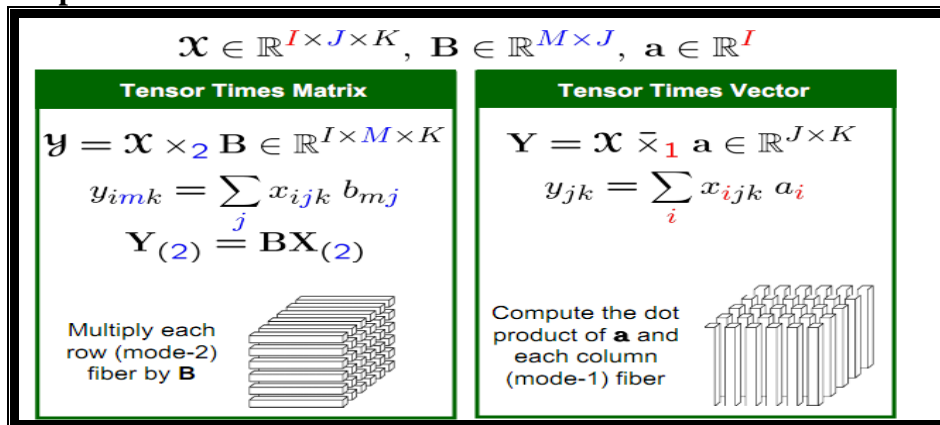


Figure 3

III. Conclusion And Future Work

If a Vector Space Model has expression of complexity of the form $\Phi(\alpha, \beta, \gamma)$ written as $\alpha^i \beta^j \gamma^k$ for the same problem which TSM solves applying same algorithm. Then, TSM will have complexity of the form $\alpha^i \beta^j \gamma^k$. This is appropriate model for higher order problems and problems involving too much of calculations. Already TSM has been proposed for hypertext representation, authorship identification and similar type of complex problem. Its new area of application is in the field of medical sciences, forensic sciences, life sciences and most importantly in computer application. Future work could be in terms of a robust toolbox which can make its application easier for lay man.

References

- [1] Introduction to Riemannian Geometry and Tensor Calculus , C.E.Weatherburn
- [2] Differential Geometry , N.J. Hicks
- [3] Nisan Haramati, Natural Language Processingwith Vector Spaces, PHYSICS 210 ,University of British Columbia
- [4] Croft ,W.B. and Lafferty, Language Modeling For Information Retrieval, Kluwer Academics,2003
- [5] K.Erk ,A simple similarity based model for selection preferences
- [6] Deng Cai, Xiaofei He, Jiawei Han, Tensor Space Model for document analysis
- [7] Baeza Yates and R Neto, Modern Information Retrieval.Addison-Wesley,1999