Word Embedding

Ramaseshan Ramachandran

WHAT IS CONTEXT?

- All words within a window or ideally within a sentence
- All content words within a window or sentence that fall in a certain frequency range
- All content words which stand in closest proximity to the word in question in the grammatical schema of each window or sentence

CONTEXT

- Context influences the word meaning
- Small boy, small car, small house, small island
- Words that occur in similar contexts will tend to have similar meanings
- Semantic similarity beween two words (w_x, w_y) is a function of how frequently they appeared in similar linguistic contexts
  - $\vec{w_x} \approx \vec{w_y}$ when the frequency of the context $(f_{C_{xy}(k)})$ with a window of size k in which both words w_x and w_y appeared is higher
- If $f_{C_{xy}}(k)$ is higher, then the semantic relationship of (w_x, w_y) is stronger
- Extending to multiple similar words for w_x:

  $\vec{w_x} \approx \vec{w_{y_i}}$ when the frequency of $C_{xy_i}(k)$ is higher, where $i = 1 \ldots n$

  Note: Here $approx$ represents similarity

SEMANTIC SPACE

- A space where the similar words (synonyms, hyponyms, hypernyms) are classified and arranged in various axes
    - Colour (hypernym) - $\underbrace{Red, Green, Orange}_{co-hyponyms}$ (hyponym) - Attributional Similarity
- A space where the similar words (synonyms, hyponyms, hypernyms) are classified and arranged in various axes
- A model or models that automatically find similar words are known as Distributed Semantic Models (DSM)
- Semantically similar words are found automatically using co-occurrences/co-locations/context

  OR
- Words connected by similar patterns are probably semantically similar

DISTRIBUTED SEMANTIC MODELS

- Extract the meaning of the words using distributed linguistic properties
- Compute lexical **co-occurrence** of every word (co-locates with certain distance) with every other word in the Vocabulary
    - Linear proximity of words within a window is considered
    - They need not represent any relations

Example He drove the car through a red bridge.
The verb drove relates to red and bridge only through the proximity,
but carries no relations with red and bridge in terms of semantics

- Build a co-occurrence matrix using co-occurrence statistics
- Rows/columns in the matrix represent distributed semantic information of words

DISTRIBUTED SEMANTIC MODELS

I cook dinner every Sunday

…

I cooked dinner last Sunday

…

I am cooking dinner today

…

My son cooks dinner every Sunday

…

- The words in this corpus are related by association
- The verb cook, cooked, cooks and cooking are related due to its co-occurrence statistics - semantic relationship
- The words dinner and Sunday are similar due to associative relationship and due to co-occurrence

- In the COVID19 research corpus it iis difficult to search and find the phrase *needle in a haystack*
- You will find needle related to $\underbrace{pain, illness, blood, drugs, syringe}_{Associative relationship}$

  and not to thread, knitting, cloth

WORD SIMILARITY

> You shall know a word by the company it keeps
>
> \- Firth, 1957

VECTOR REPRESENTATION OF WORDS

Let V be the unique set of terms and $|V|$ be the size of the vocabulary. Then every vector representing the word $\mathscr{R}^{|\mathrm{V}|\mathrm{x}1}$ would point to a vector in the V-dimensional space

ONE-HOT VECTOR - 1

Consider all the ≈39000 words (estimated tokens in English is ≈ 13M) in the Oxford Learner's pocket dictionary. We can represent each word as an independent vector quantity as follows in the real space $\mathscr{R}^{|V|X1}$

$$t^{a} = \begin{pmatrix} 1 \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix} t^{aback} = \begin{pmatrix} 0 \\ 1 \\ \dots \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix} \dots t^{zoom} = \begin{pmatrix} 0 \\ 0 \\ \dots \\ 0 \\ \dots \\ 1 \\ 0 \end{pmatrix} t^{zucchini} = \begin{pmatrix} 0 \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ 1 \end{pmatrix}$$

This is a very simple codification scheme to represent words independently in the vector space. This is known as ***one-hot vector***.

ONE-HOT VECTOR - 2

In one-hot vector, every word is represented independently. The terms, *home, house, apartments, flats* are independently coded. With one-hot vector based model, the dot product

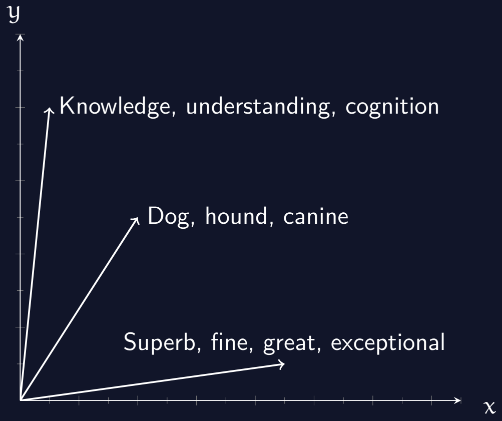
$$\left(\mathrm{t}^{\mathrm{House}}\right)^{\mathsf{T}} \cdot \mathrm{t}^{\mathrm{Apartment}} = 0 \tag{1}$$

$$\left(\mathrm{t}^{\mathrm{Home}}\right)^{\mathsf{T}} \cdot \mathrm{t}^{\mathrm{House}} = 0 \tag{2}$$

With one-Hot vector, there is no notion of similarity or synonyms.

RELATIONSHIP AMONG TERMS - SYNONYMS

We could represent all the synonyms of a word in one axis


y
Knowledge, understanding, cognition
Dog, hound, canine
Superb, fine, great, exceptional
x

POLYSEMOUS WORD - BANK

Synset('bank.n.01')	sloping land (especially the slope beside a body of water)
Synset('depository-financial-institution.n.01')	a financial institution that accepts deposits and channels the money into lending activities
Synset('bank.n.03')	a long ridge or pile
Synset('bank.n.10')	a flight maneuver; aircraft tips laterally about its longitudinal axis (especially in turning)
Synset('trust.v.01')	have confidence or faith in

Bank

appears in different word senses - or the meaning of the word is determined by the context in which appears

POLYSEMOUS WORD - PROGRAM

Synset(’plan.n.01’)	a series of steps to be carried out or goals to be accomplished
Synset(’program.n.02’)	a system of projects or services intended to meet a public need
Synset(’broadcast.n.02’)	a radio or television show
Synset(’platform.n.02’)	a document stating the aims and principles of a political party
Synset(’program.n.05’)	an announcement of the events that will occur as part of a theatrical or sporting event
Synset(’course_of_study.n.01’)	an integrated course of academic studies
Synset(’program.n.07’)	(computer science) a sequence of instructions that a computer can interpret and execute
Synset(’program.n.08’)	a performance (or series of performances) at a public presentation
Synset(’program.v.01’)	arrange a program of or for
Synset(’program.v.02’)	write a computer program

SYNONYMS

small.a.01	['small', 'little']
minor.s.10	['minor', 'modest', 'small', 'small-scale', 'pocket-size', 'pocket-sized']
humble.s.01	['humble', 'low', 'lowly', 'modest', 'small']
little.s.07	['little', 'minuscule', 'small']
belittled.s.01	['belittled', 'diminished', 'small']
potent.a.03	['potent', 'strong', 'stiff']
impregnable.s.01	['impregnable', 'inviolable', 'secure', 'strong', 'unassailable', 'hard'] He has such an impregnable defense (Cricket-Very hard to find the gap between the bat and the pad)
solid.s.07	['solid', 'strong', 'substantial']
strong.s.09	['strong', 'warm']
firm.s.03	['firm', 'strong'] - firm grasp of fundamentals

CONTEXTUAL UNDERSTANDING OF TEXT

You shall know a word by the company it keeps - (Firth, J. R. 1957)

- In order to understand the word and its meaning, it not enough if we consider only the individual word
- The *meaning* and *context* should be central in understanding word/text
- Exploit the context-dependent nature of words
- Language patterns cannot be accounted for in terms of a single entity
- The *collocation*, a particular word consistently co-occurs with the other words, gives enough clue to understand a word and its meaning

UNDERSTANDING A WORD FROM ITS CONTEXT

The view from the top of the mountain was
The view from the summit was
La vue du sommet de la montagne était
Mtazamo wa juu wa mlima huo ulikuwa

awesome/(*impressionnante*, *impressionnant*)
breathtaking
amazing, அற்புதமான/അത്ഭുതകരമായ/ಅದ್ಭುತ/అద్భుతమైన
stunning/(*superbe*)
astounding अद्भुत/चमकप्रद
astonishing
awe-inspiring
extraordinary
incredible/(*incroyable*)
unbelievable
magnificent शानदार/ഗംഭീരമായ/ભવ્ય
wonderful/(*ajabu*)
spectacular
remarkable/(*yakuvutia*)

SEMANTICALLY CONNECTED VECTORS

- Identify a model that enumerates the relationships between terms
- Identify a model that tries to put similar items closer to each other in some space or structure
- Build a model that discovers/uncovers the semantic similarity between words and documents in the latent semantic domain
- Develop a distributed word vectors or dense vectors that captures the linear combination of word vectors in the transformed domain
- Transform the term-document space into a synonymy and a semantic space

METHODS TO CREATE WORD VECTORS

- Brown clustering - statistical algorithms for assigning words to classes based on the frequency of their co-occurrence with other words
- Hyperspace Analogue to Language - HAL
- Correlated Occurrence Analogue to Lexical Semantic - COALS
- Latent Semantic Analysis or Latent Semantic Indexing
- Global Vectors - GloVe
- Neural networks using skip grams and CBOW
  - CBOW - uses surrounding words to predict the center of words
  - Skip grams use center of words to predict the surrounding words

WORD SIMILARITY

- Sparse vectors are too long and not very convenient as features machine learning
- Abstracts more than just frequency counts
- It captures neighborhood words that are connected by synonyms

WORD SIMILARITY

You shall know a word by the company it keeps

- Firth, 1957

ATTRIBUTIONAL SIMILARITY

Attributional Similarity

- Two words are similar if they shared similar attributes - cat and kitten, dog and puppy
- Refers to the degree of similarity between two words or phrases in terms of their shared attributes
- Words that share many collocates denote concepts that share many attributes

Relational Similarity

- Related by concepts/roles - King and queen are related by the roles in the monarchy
- Blood relationships - siblings, aunts, uncles, parents, etc.

Techniques - Distributional Semantic Models

VECTOR BASED MODELS

Assumption Context words within a certain distance from the target word are semantically relevant

- Ability to represent word meaning simply by using distributional statistics
- The context surrounding a given word provides important information about its meaning
    - Small number of words surrounding the target word is known as context
- Distributional patterns of co-occurrence with their neighboring words provide semantic properties of words

TECHNIQUES TO CAPTURE CO-OCCURRENCE INFORMATION

Let A denote the word-word co-occurrence matrix where each row corresponds to a unique/target word, and each column represents a context.
$a_i j$ denotes every element of the A

a_i denotes the number of times the word i co-occurring with the word j, $a_i = \sum_j a_{ij}$

Now, we can fill the co-occurrence using:

1. **Raw Frequency Count:** Each element, a_{ij} denotes the raw frequency count of word i co-occurring with the word j
2. **Probability:** $p_{ij} = P(w_j \mid w_i) = \dfrac{a_{ij}}{a_i}$
3. **Point-wise Mutual Information(PMI)**: $P(w_i, w_j) = max\left(\log_2\left(\dfrac{p_{ij}}{p_i * p_j}\right), 0\right)$

The co-occurrence statistics are captured by scaning the entire corpus once using a windowing approach

A SAMPLE CO-OCCURRENCE MATRIX WITH RAW FREQUENCY COUNT

	w_0	w_1	w_2	$\dots$	w_{n-3}	w_{n-2}	w_{n-1}	w_n
w_0	0	33	29	$\dots$	33	37	39	39
w_1	33	1	45	$\dots$	0	27	21	10
w_2	29	45	0	$\dots$	37	40	19	23
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
w_{n-3}	33	0	37	$\dots$	0	24	26	49
w_{n-2}	37	27	40	$\dots$	24	0	22	31
w_{n-1}	39	21	19	$\dots$	26	22	1	38
w_n	39	10	23	$\dots$	49	31	38	0

SEMANTIC SPACE

- A semantic space model is method of assigning each word in a language to a point in a real finite dimensional vector space. Formally it is a quadruple $\mathbb{A}, \mathbb{B}, \mathbb{S}, \mathbb{M}$[1].
- $\mathbb{B}$ is the set $b_{1...D}$ of basis elements, the dimensions of the space. $\mathbb{B}$ can be a set of words.
- The dimensionality of the matrix is k where k will represent k most frequent words (minus the stop words) in a corpus.
- A word vector can be defined as $\vec{v} = \langle A(f(t, b_1)), A(f(t, b_2)), ..., A(f(t, b_n)) \rangle$ where A is an association function. A is an identity matrix, if raw frequencies are used. t is the target word and b is the basis element[2].
- $\mathbb{S}$ is a similarity measure that maps pairs of vectors onto a continuous valued quantity that represents contextual similarity
- $\mathbb{M}$ is a transformation that takes one semantic space and maps it onto another, for example by reducing its dimensionality

SIMILARITY MEASURES

A similarity measure [3][3] is a real-valued function that quantifies the similarity between two objects - in this case words. Some of the similarity measures are given below.

$$\textbf{Euclidean Distance - } \mathbb{E}(\vec{w_1}, \vec{w_2}) = \sqrt{w_1^2 - w_2^2} \tag{3}$$

$$\textbf{Cosine Similarity} = \frac{\vec{w_1}.\vec{w_2}}{\|\vec{w_1}\|\,\|\vec{w_2}\|} \tag{4}$$

$$\textbf{Cosine distance} = 1 - \textbf{Cosine Similarity} \tag{5}$$

$$\textbf{Cluster similarity-}\mathscr{L}(\vec{w_1}, \vec{w_2}) = \frac{\vec{w_1}.\vec{w_2}}{\|\vec{w_1}\|} \tag{6}$$

WORD VECTOR EXAMPLES

```
Similar words for apple
’apple’,  0
’iphone’,  0.266
’ipad’,  0.287
’apples’,  0.356
’blackberry’,  0.361
’ipod’,  0.365
’macbook’,  0.383
’mac’,  0.391
’android’,  0.391
’google’,  0.395
’microsoft’,  0.418
’ios’,  0.433
’iphones’,  0.445
’touch’,  0.446
’sony’,  0.447
```

Similar words for - american

```
’american’, 0
’america’, 0.255
’americans’, 0.312
’u.s.’, 0.320
’british’, 0.323
’canadian’, 0.329
’history’, 0.356
’national’, 0.364
’african’, 0.374
’society’, 0.375
’states’, 0.386
’european’, 0.387
’world’, 0.394
’nation’, 0.399
’us’, 0.399
```

VECTOR DIFFERENCE BETWEEN TWO WORDS

$$vec(apple) - vec(iphone)$$

```
(’raisin’, 0.5744591153088133)
(’pecan’, 0.576061737414159)
(’cranberry’, 0.5840016172254104)
(’butternut’, 0.5882322018694753)
(’cider’, 0.5910795032086132)
(’apricot’, 0.6036644437522422)
(’tomato’, 0.6073715970323961)
(’rosemary’, 0.6150986936477657)
(’rhubarb’, 0.6157884153793192)
(’feta’, 0.6183016129045151)
(’apples’, 0.6226003361980218)
(’avocado’, 0.6235366677962004)
(’fennel’, 0.6306016018912576)
(’chutney’, 0.6312524337590703)
(’spiced’, 0.6327632200841328)
```

VECTOR ARITHMETIC ON WORD VECTORS...

840B words and 300 elements word vectors used for this computation

$\vec{apple}$	$\vec{apple} - \vec{iphone}$	$\vec{apple} - \vec{fruit}$
('apple', 0)	('apples', 0.39)	('ipad', 0.412)
('apples', 0.25)	('fruit', 0.43)	('iphone', 0.433)
('blackberry', 0.31)	('grape', 0.44)	('macbook', 0.435)
('Apple', 0.35)	('tomato', 0.44)	('ipod', 0.445))
('iphone', 0.37)	('pecan', 0.45)	('imac', 0.465
('fruit', 0.37)	('rhubarb', 0.45)	('3gs', 0.473)
('blueberry', 0.38)	('pears', 0.45)	('Ipad', 0.490)
('strawberry', 0.38)	('cranberry', 0.452)	('itouch', 0.512)
('ipad', 0.39)	('raisin', 0.453)	('ipad2', 0.514)
('pineapple', 0.39)	('apricot', 0.459)	('Iphone', 0.514)
('pear', 0.39)	('carrot', 0.461)	('ios', 0.520)
('cider', 0.39)	('candied', 0.462)	('Macbook', 0.524)
('mango', 0.40)	('blueberry', 0.463)	('ibook', 0.534)
('ipod', 0.40)	('apricots', 0.466)	('IPhone', 0.541)
('raspberry', 0.40)	('tomatoes', 0.466)	('32gb', 0.545)

REFERENCES

[1] Will Lowe. "Towards a theory of semantic space". In: *Proceedings of the annual meeting of the cognitive science society*. Vol. 23. 23. 2001.

[2] Sebastian Padó and Mirella Lapata. "Dependency-Based Construction of Semantic Space Models". In: *Computational Linguistics* 33.2 (2007), pp. 161–199. DOI: 10.1162/coli.2007.33.2.161. URL: https://aclanthology.org/J07-2002.

[3] Lillian Lee. "Measures of Distributional Similarity". In: *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, Maryland, USA: Association for Computational Linguistics, June 1999, pp. 25–32. DOI: 10.3115/1034678.1034693. URL: https://aclanthology.org/P99-1004.

[4] Kevin Lund and Curt Burgess. "Producing high-dimensional semantic spaces from lexical co-occurrence". en. In: *Behavior Research Methods, Instruments, & Computers* 28.2 (1996), pp. 203–208. ISSN: 0743-3808, 1532-5970. DOI: 10.3758/BF03204766. URL: http://link.springer.com/article/10.3758/BF03204766 (visited on 09/09/2015).

[5] Douglas LT Rohde, Laura M Gonnerman, and David C Plaut. "An improved model of semantic similarity based on lexical co-occurrence". In: *Communications of the ACM* 8.627-633 (2006), p. 116.