Semantic Similarity Using Word2Vec

Rakshit Sharma (CSE)
Advisor: Professor Amitabha Mukherjee (CSE)

Abstract

Finding semantic similarity touches wide range of problems in real life scenarios. It may be used to summarize the news articles, to find similarities in the reviews of some product so as to get pros and cons of a product, etc. For doing semantic similarity, the problem with Indian language is that there is no proper parser available available for the Indian languages. Hence we proceed through WordVector implementation.

Motivation

- No proper parser available for Indian language.
- Local language user makes up to large user base on the internet.

	Internet	Language	user	Penetration
ALL INDIA	269Mn	127Mn	47%	
URBAN INDIA	188Mn	81Mn	51%	No 57% / Yes 43%
RURAL INDIA	81Mn	46Mn	41%	No 43% / Yes 57%

Dataset

Hindi news dataset of about 300 news articles

Project Goals/Objectives

- Collect and create dataset in Hindi language.
- Word level similarity using WordVector.
- Sentence level similarity using word level similarity.
- Semantic Vector for sentences.
- Being able to merge news articles using the above sentence similarity

Methods/Process

The approach is similar to Yuhua Li et al[2] for "Sentence Similarity Based on Semantic Nets and Corpus Statistics".

Get hierarchical knowledge base by using Word2Vec and then getting a raw word similarity based on common synsets.
We take the joint word set of the two sentences that we have to compare then we make an n-dimensional vector each for sentence T1 and T2 where n = #words in T1 + #words in T2 followed by calculating weights of each dimension as:-

$$\begin{Bmatrix} S1[i] = 1 & \forall w[i] \text{ in the joint word set} \in T1 \\ S1[i] = \text{similarity score} & \forall w[i] \text{ in the joint word set} \notin T1 \end{Bmatrix}$$

$$\begin{Bmatrix} S2[i] = 1 & \forall w[i] \text{ in the joint word set} \in T2 \\ S2[i] = \text{similarity score} & \forall w[i] \text{ in the joint word set} \notin T2 \end{Bmatrix}$$

$$S_s = \frac{\mathbf{s_1} \cdot \mathbf{s_2}}{\|\mathbf{s_1}\| \cdot \|\mathbf{s_2}\|}.$$

$$\begin{Bmatrix} R1[w[i]] = index[w[i]] & \forall w[i] \text{ in the joint word set} \in T1 \\ R1[w[i]] = index[\text{similar word}] & \forall w[i] \text{ in the joint word set} \notin T1 \end{Bmatrix}$$

$$\begin{Bmatrix} R2[w[i]] = index[w[i]] & \forall w[i] \text{ in the joint word set} \in T2 \\ R2[w[i]] = index[\text{similar word}] & \forall w[i] \text{ in the joint word set} \notin T2 \end{Bmatrix}$$

$$S_r = 1 - \frac{\|\mathbf{r_1} - \mathbf{r_2}\|}{\|\mathbf{r_1} + \mathbf{r_2}\|}.$$

$$S(T_1, T_2) = \delta S_s + (1 - \delta) S_r$$

Once we have the results, recognize the threshold to classify similar and dissimilar by ROC curve.
The binary classifier can then be used to merge the news articles based on what are similar and dissimilar sentences in the articles.

TABLE 1
Process for Deriving the Semantic Vector

	RAM	keeps	things	being	worked	with	The	CPU	uses	as	a	short-term	memory	store
RAM	1												0.8147	0.8147
keeps		1												
things			1					0.2802	0.4433					
being				1										
worked					1									
with						1								
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
$\vec{s}$	1	1	1	1	1	1	0	0.2802	0.4433	0	0	0	0.8147	0.8147
Weight	I(RAM) / I(RAM)	I(keeps) / I(keeps)	I(things) / I(things)	I(being) / I(being)	I(worked) / I(worked)	I(with) / I(with)	I(The) / I(The)	I(CPU) / I(things)	I(uses) / I(things)	I(as) / I(as)	I(a) / I(a)	I(short-term) / I(short-term)	I(memory) / I(RAM)	I(store) / I(RAM)

Results

S.No.	Sentence Pairs	Similarity	S.No.	Sentence Pairs	Similarity
1	बीजेपी की बिहार में करारी हार महागठबंधन ने बीजेपी को बिहार में 100 सीटों से हराया	0.650628	5	नरेंद्र मोदी अपने मंत्रियों के बड़बोलेपन से परेशान नरेंद्र मोदी का मंत्रियों के साथ विदेश दौरा टला	0.558449
2	बीफ की अफवाह पर भीड़ ने एक इंसान को जान से मार डाला दादरी में भीड़ ने एक मासूम को जान से मारा	0.633209	6	उमा जी कहती है नरेंद्र मोदी विनाश पुरुष नरेंद्र मोदी के गुजरात में विकास नहीं सिर्फ दंगे हुए हैं	0.319521
3	यह एक जूता है मैं कल जूता लेने गया	0.219979	7	छपरा के मढ़ौरा में प्रधानमंत्री नरेंद्र मोदी की रैली बिहार चुनाव के मद्देनज़र पार्टियों की रैली जोर-शोर पर	0.436039
4	आज NLP की कक्षा है यह एक जूता है	0.0	8	नीतीश कुमार सन्निपात रोगी की तरह परस्पर विरोधी बातें कर रहे हैं नीतीश कुमार प्रधानमंत्री नरेंद्र मोदी की यात्रा से भी परेशान होते हैं	0.4790112

Conclusions/Recommendations

An approach was given to calculate sentence similarity score with the help of Word2Vec. Sentences having similar words tend to give more score irrespective of the overall sentence meaning, hence considered the impact of word order on sentence meaning. However, still without proper word disambiguation, similarity score was much higher then human score in some cases like in example 5 and vice-versa as in example 6. However in real life scenarios, sentences having many similar words tend to convey same meaning most of the time.

References

- Yuhua Li et al "Sentence Similarity Based on Semantic Nets and Corpus Statistics" (AUGUST 2006) (IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 18, NO. 8) http://www.aicit.org/AISS/ppl/AISS1666PPL.pdf
- Poster Template:- https://goo.gl/dd5j74
- Userbase information:- http://goo.gl/vqnhNU
- Userbase information:- http://goo.gl/6nTxEP
- List of hindi stop words:- https://goo.gl/a0gRzd