Single or Multiple? Combining Word Representations Independently Learned from Text and WordNet

AAAI 2016

Josu Goikoetxea, Eneko Agirre, and Aitor Soroa

巴斯克大学

outline

- 作者简介
- 论文简介
- 相关工作：transH, retrofitting
- 词向量的生成与合并 (random walk)
- 实验效果、对比、多组词向量合并效果

作者简介

- Eneko Agirre
- Processing of Basque, 语义相似度/关联度，WSD,SRL，IE，。。。
- 比较厉害

- Josu Goikoetxea

- Aitor Soroa
- NLP，CL，AI

论文简介

论文简介

我有很多不同类型的语料，怎么训练出一个好的词向量？

论文简介

我有很多不同类型的语料，怎么训练出一个好的词向量？

直接在不同的语料上跑 word2vec, 然后把这些词向量拼接起来/PCA 就好了！

相关工作

— 不同的语料注入到一个词向量

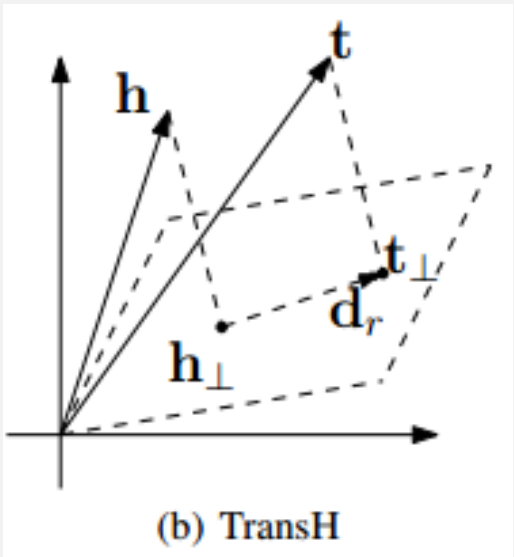
相关工作

2012.SIGKDD

将 WordNet 的信息加入目标函数中训练，要求 WordNet 中有关联的点词向量近

2014.AAAI transH

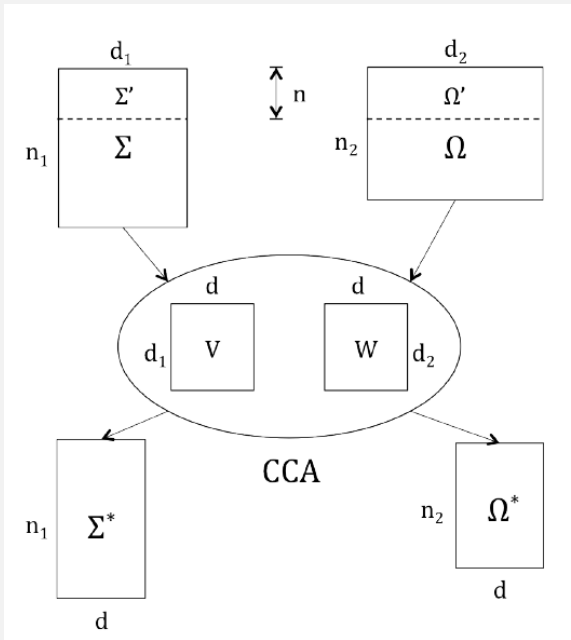
$$f_r(\mathbf{h},\mathbf{t}) = \|(\mathbf{h} - \mathbf{w}_r^\top \mathbf{h}\mathbf{w}_r) + \mathbf{d}_r - (\mathbf{t} - \mathbf{w}_r^\top \mathbf{t}\mathbf{w}_r)\|_2^2.$$



(b) TransH

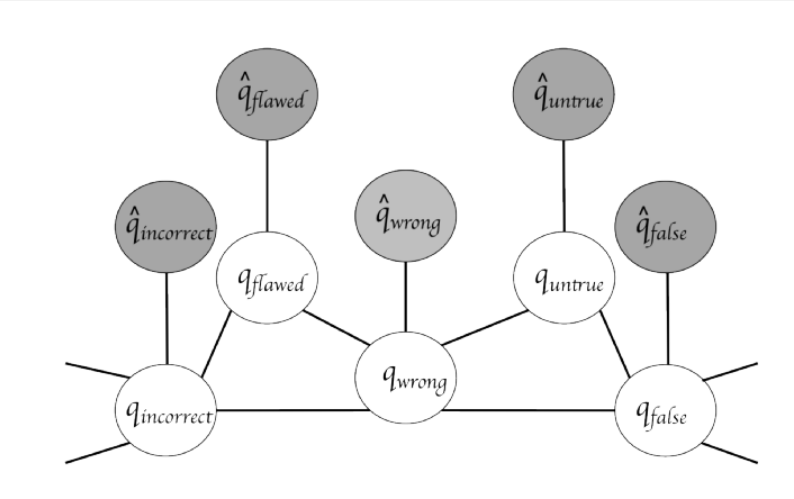
2014.EACL

将不同语言的词向量映射到同一个平面，然后使用CCA找到关联，将不同语言的词向量通过不同的矩阵映射到同一个空间



相关工作

2015.EMNLP Retrofitting with Semantic Lexicons-顺序模型



1. 传统工具 (word2vec) 生成初始向量空间 $\hat{Q}$ (左图灰色点)
2. 根据语义字典生成 Ω （左图的边）
3. 最优 (小) 化 $\Psi(Q) =$

$$\sum_{i=1}^{n}\left[\alpha_i\left\|q_i-\hat{q}_i\right\|^2+\sum_{(i,j)\in E}\beta_{ij}\left\|q_i-q_j\right\|^2\right]$$

2015.NACCL Multiview LSA: Representation Learning via Generalized CCA

使用 Generalized CCA 将 Wikipedia，PPDB，WordNet，FrameNet，CatVar 生成的不同的词向量合并成一个向量

实验方法

— 词向量生成

— 词向量合并

实验方法. 词向量生成

从 text 生成的词向量

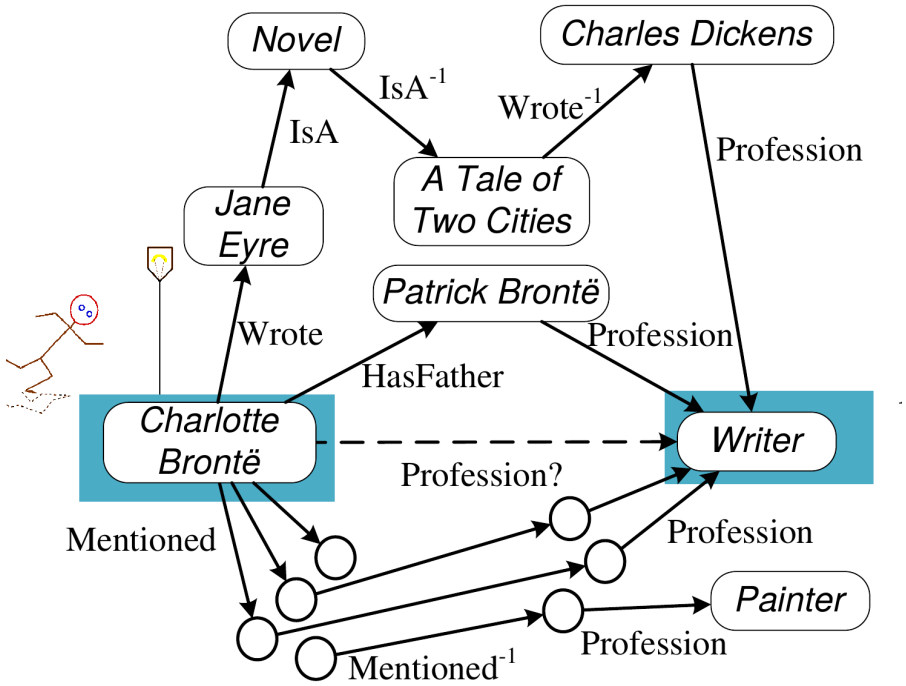
（Wikipedia+British National Corpus+ukWaC）+word2vec.skipgram

从 WordNet（WN）生成的词向量

1. 在 WN 上做 random walk，得到一些路径，每条路径就是一个“句子”
   - 一个例子：yucatec(尤卡坦语) mayan quiche(火腿起司蛋卷) kekchi(克奇人) speak sino-tibetan（汉藏语系）tone language west chadic（乍得）talk
2. 对这些“句子”生成的“假文本”+word2vec.skipgram 得到向量

random walk(RW)

- random walk = 双向 path ranking algorithm (PRA) + sampling



$$score(s,t) = \sum_{P \in \mathbf{P}} f_P(s,t)\theta_P$$

- $\mathbf{P}$ is the set of all relation paths with length $\leq L$
- $f_P(s,t) = \text{Prob}(s \rightarrow t; P)$

实验方法. 词向量合并-1

词向量直接组合

1. CAT(concatenating): 2 个词向量拼接（300 维->600 维）
2. CEN(centroid): 词向量平均
3. CMP(complex): 组合成一个复数，$(v_1 + v_2 i)$

相关性分析

1. PCA: CAT->300 维
2. CCA: 将 2 个词向量映射到一个新的、一致的向量空间

语料组合

（Wikipedia+British National Corpus+ukWaC+**random walk 生成的“假文本”**）+word2vec.skipgram

实验方法. 词向量合并-2

相似度组合

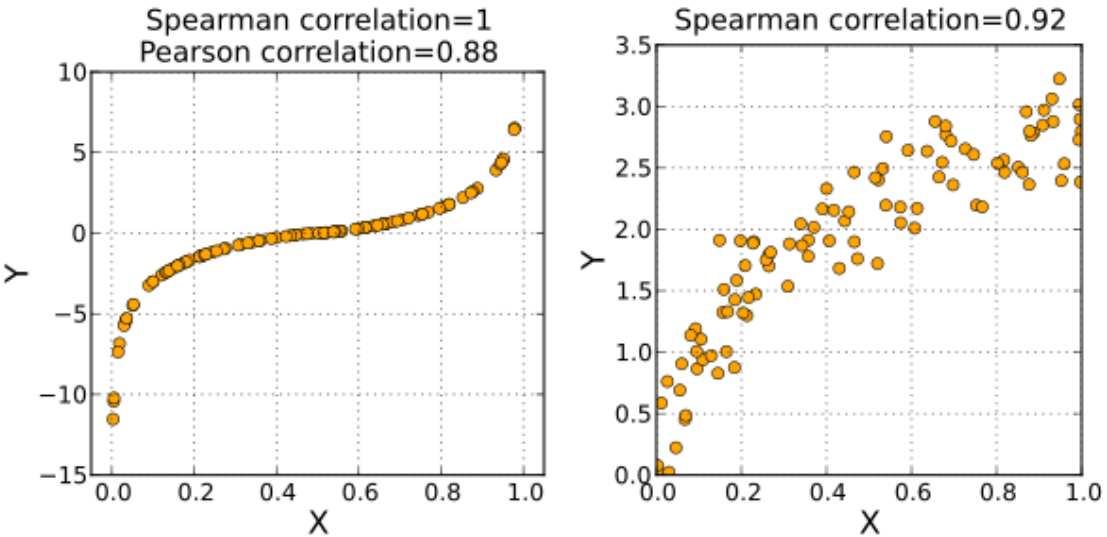
1. AVG: 不合并 2 组向量，计算相似度的时候使用 2 个相似度的均值
2. RNK: 对于每组向量：对于实验中的数据计算相似度并排序。用 2 个排名的平均值作为每组数据的最终排名

实验方法. 词向量合并-2

相似度组合

1. AVG: 不合并 2 组向量，计算相似度的时候使用 2 个相似度的均值
2. RNK: 对于每组向量：对于实验中的数据计算相似度并排序。用 2 个排名的平均值作为每组数据的最终排名

评价结果好坏采用斯皮尔曼等级相关系数



$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \in [-1, 1]$$

实验

— 0. 数据集 — 1. 不同词向量组合的效果

— 2. 和 NACCL2015 Retrofitting 的对比

— 3. 简单对比其他的词向量组合/提升模型

— 4. 多种（>2) 语料的词向量混合

实验. 数据集

WordSim353 Similarity(WSS)

- 353 个英语单词对（200 个 13 个人标，153 个 16 个人标），相似度：0-10
- 定义：Word 1 Word 2 Human (mean)
- 例子：love sex 6.77

WordSim353 Relatedness(WSR)

- 252 个英语单词对，相关度：0-10
- 定义：Word 1 Word 2 Human (mean)
- 例子：planet galaxy 8.11

RG-65(RG)

MEN

- 3000 对单词（共现 700 次）

MTURK287(MTU)

SimLex-999(SL)

- 999 对相同词性的单词对，0-10 的相似度（0 表示完全不相似）
- 数据格式：word1 word2 POS SimLex999 conc(w1) conc(w2) concQ Assoc(USF)...
- 例子：smart intelligent A 9.2 1.75 2.46 1 7.11 ...

实验. 不同词向量组合的效果

	RG	SL	WSS	WSR	MTU	MEN	WS	sim	rel	all
RWwn	82.3	52.5	76.2	58.7	62.1	75.4	68.7	70.3	65.4	68.2
WBU	76.4	39.7	76.6	61.5	64.6	74.6	67.3	64.2	66.9	64.5
CAT	7.8	**12.5**	6.7	6.5	7.5	**6.0**	8.0	9.0	**6.7**	8.4
CEN	4.6	9.6	2.7	-1.1	1.3	3.2	2.3	5.6	1.2	4.2
CMP	-3.4	-1.2	-2.9	-8.9	-7.4	-0.9	-6.9	-2.5	-5.7	-4.0
PCA	**10.8**	**12.5**	5.7	5.3	**8.3**	5.6	6.9	**9.6**	6.5	**8.9**
CCA	6,8	2,7	-0,4	-0,2	11,7	-6,1	-3,5	6,0	-3,3	2,3
COR	6.6	8.2	**7.2**	8.8	3.3	4.1	**8.6**	7.4	5.4	6.2
AVG	8.0	12.1	5.5	6.5	7.0	6.2	7.4	8.5	6.6	8.2
RNK	7.3	11.3	0.2	**11.7**	-14.7	-14.7	6.6	6.2	-5.9	-0.8

- CAT: 拼接，AVG：相似度均值，COR：语料混合，CEN：向量平均，RNK：相似度排名均值，CMP：复数表示
- $PCA > CAT \approx AVG > COR > CEN > CMP$

实验. 和 NACCL2015 Retrofitting 的对比

	RG	SL	WSS	WSR	MTU	MEN	WS	sim	rel	all
FAR	74.8	43.7	74.1	61.0	69.9	68.0	65.6	64.2	66.5	64.4
+WN_{sh}	5.0	7.4	4.0	-1.1	-0.6	2.6	1.9	5.5	0.3	3.3
+WN_{all}	4.9	2.5	2.6	4.3	2.4	5.7	3.7	3.3	4.1	3.9
WBU	76.4	39.7	76.6	61.5	64.6	74.6	67.3	64.2	66.9	64.5
+WN_{sh}	4,6	-12,2	-4,8	-18,6	8,0	-4,9	-2,7	2,6	-4,3	1,3
+WN_{all}	6,3	0,9	2,3	0,2	2,4	0,9	0,9	3,7	0,3	2,1
PCA	**10.8**	**12.5**	**5.7**	**5.3**	**8.3**	**5.6**	**6.9**	**9.6**	**6.5**	**8.9**

WN_{sh}: 上位、同义词

WN_{all}: 还用“part-of”，“gloss relation”等关系

- 前三行是原文章的效果，下面 4 行是本文的效果
- 2,3 行说明：上位、同义信息对于相似度判断有帮助，WordNet 的全部关系对判断单词 relation 有帮助
- 说明作者的 WordNet 向量还是很有竞争力的
- 为什么作者的好：（physics-proton）在 WN 没有关联边，retrofit 作用没效果；但是作者训练出来的 WN-vec 中却是非常相似的 2 个向量
- 作者实验的缺点：如果把 WordNet 换成 PPPD，作者的实验不能得到提升，反而下降

实验. 简单对比其他的词向量组合/提升模型

	RG	SL	WSS	WSR	MTU	MEN	WS
txt	—	—	—	—	69.2	—	74.4
CLEAR gain	—	—	—	—	-0.5	—	2.3
txt	71.2	34.5	76.8	60.1	59.1	71.4	68.0
MVLSA gain	9.6	9.4	2.4	3.4	3.8	4.4	2.1
txt	—	—	—	—	—	—	64.7
FREEBASE gain	—	—	—	—	—	—	3.7

CLEAR

Yahoo! Answers corpus + WordNet 上位词、同义词、meronyms（部分名词）

MVLSA

LSA+WordNet

FREEB

freebase relation 优化

- 本文和上面三个都没有可比性，因为 baseline 不一样，不过第三个使用 Freebase 的信息引导作者从 Wikipedia 上面挖更多的语料

实验. 多种（>2) 语料的词向量混合

	RG	SL	WSS	WSR	MTU	MEN	WS	sim	rel	all
(a) WBU	76.4	39.7	76.6	61.5	64.6	74.6	67.3	64.2	66.9	64.5
(b) GOOG	76.0	44.2	77.8	60.0	65.5	74.6	68.1	66.0	66.5	65.6
(c) RWwn	82.3	52.5	76.2	58.7	62.1	75.4	68.7	70.3	65.4	68.2
(d) PPVwn	85.7	49.3	69.4	44.1	54.5	66.1	56.9	68.1	54.9	62.5
(e) RWwiki	79.6	32.3	67.5	48.2	43.9	60.9	59.3	59.8	51.0	55.2
(f) PPVwiki	88.6	29.2	80.7	62.1	64.5	74.1	72.7	66.2	66.9	65.8
CAT(ac)	84.2	52.2	83.3	68.0	72.1	80.6	75.3	73.2	**73.6**	72.9
CAT(ace)	**91.2**	51.4	80.4	64.0	66.4	78.4	73.6	74.3	69.6	72.2
CAT(abce)	**91.2**	51.6	80.7	64.2	66.7	78.6	73.8	74.5	69.4	72.4
AVG(ac)	84.4	51.7	82.1	68.0	71.6	80.8	74.7	72.8	73.5	72.7
AVG(ace)	89.5	52.6	82.4	68.2	71.2	81.4	75.9	74.8	73.6	74.1
AVG(abce)	89.0	52.1	83.5	68.2	73.4	81.7	76.5	74.9	74.4	74.5
AVG(-f)	89.4	54.1	84.0	68.6	73.7	82.1	76.9	75.8	74.8	75.2
AVG(-e)	86.4	53.8	83.8	**69.3**	**74.0**	81.8	76.3	74.6	75.0	74.4
AVG(-d)	89.9	52.9	84.0	68.8	73.5	82.0	77.1	75.6	74.7	75.1
AVG(-c)	89.6	51.4	83.9	66.8	70.8	80.6	76.2	75.0	72.7	73.3
AVG(-b)	89.9	55.3	83.7	69.1	71.6	82.0	77.0	76.3	74.3	75.2
AVG(-a)	90.4	**56.6**	83.2	62.7	71.8	81.6	77.1	**76.8**	72.0	75.5
AVG(ALL)	90.2	54.7	**84.3**	69.1	73.7	**82.8**	**77.4**	76.4	**75.1**	**75.7**
s-o-t-a	86.0	55.2	80.0	*70.0*	*75.1*	80.0	*85.0*	73.7	75.0	*76.3*

6 种语料/词向量

1. WBU: 之前的文本语料
2. GOOG: google news 训练好的
3. RWwn: 之前的“伪语料
4. PPVwn: 在 WrodNet 上使用 Personalized PageRank 得到
5. RWwiki: 在维基百科页面上仿照 WordNet 做 Random walk 生成语料。2 个页面之间有超链接则相连
6. PPVwiki: 在 Wikipedia 上使用 Personalized PageRank 得到

- CAT 对于超过 2 个来源就不管用了；“-1”实验说明每个语料都有用

总结

- 简单的词向量拼接就很不错了
- 对于 WordNet 这样的资源，在上面训练一个还不错的词向量要比把 WordNet 的信息嵌入到文本训练中更有用

问题？

我的问题

1. AVG 和 CEN 不是一样的吗？

我的问题

1. AVG 和 CEN 不是一样的吗？
2. 对于 WordNet 的信息，下面两个数据用 word2vec 训练哪个效果好？

窗口大小 5

- yucatec mayan quiche kekchi
- speak sino-tibetan tone language west chadic talk

窗口大小 1 或 2

- s yucatec WN-synonymy mayan e
- s mayan WN-category quiche e
- s quiche WN-category-rev kekchi e
- ...