深度之眼
deepshare.net

# Efficient Estimation of Word Representations in Vector Space part 1

导师：Pvop

# 《Efficient Estimation of Word Representations in Vector Space》

## 基于向量空间中词表示的有效估计

作者：Tomas Mikolov（第一作者）

单位：Google

# 前期知识储备

Pre-knowledge reserve

## 数学知识

高等数学中偏微分，线性代数中矩阵基本运算，概率论中的条件概率

## 机器学习

- 机器学习中基本的原理及概念，如逻辑回归分类器、梯度下降方法等

## 神经网络

了解神经网络基本知识，知道前馈神经网络和循环神经网络的概念，知道语言模型概念

## 编程

了解PyTorch基本使用方法，如数据读取、模型构建等

# 学习目标

Learning objectives

### 了解词向量的背景知识

- 历史背景
- 数学基础

### 了解前人工作

- SVD
- 前馈神经网络
- Rnn语言模型

### 掌握词向量的评价方法

- Cosine/Analogy
- 论文中数据集

### 掌握模型结构

- CBOW
- Skip-gram

### Word2vec中的关键技术

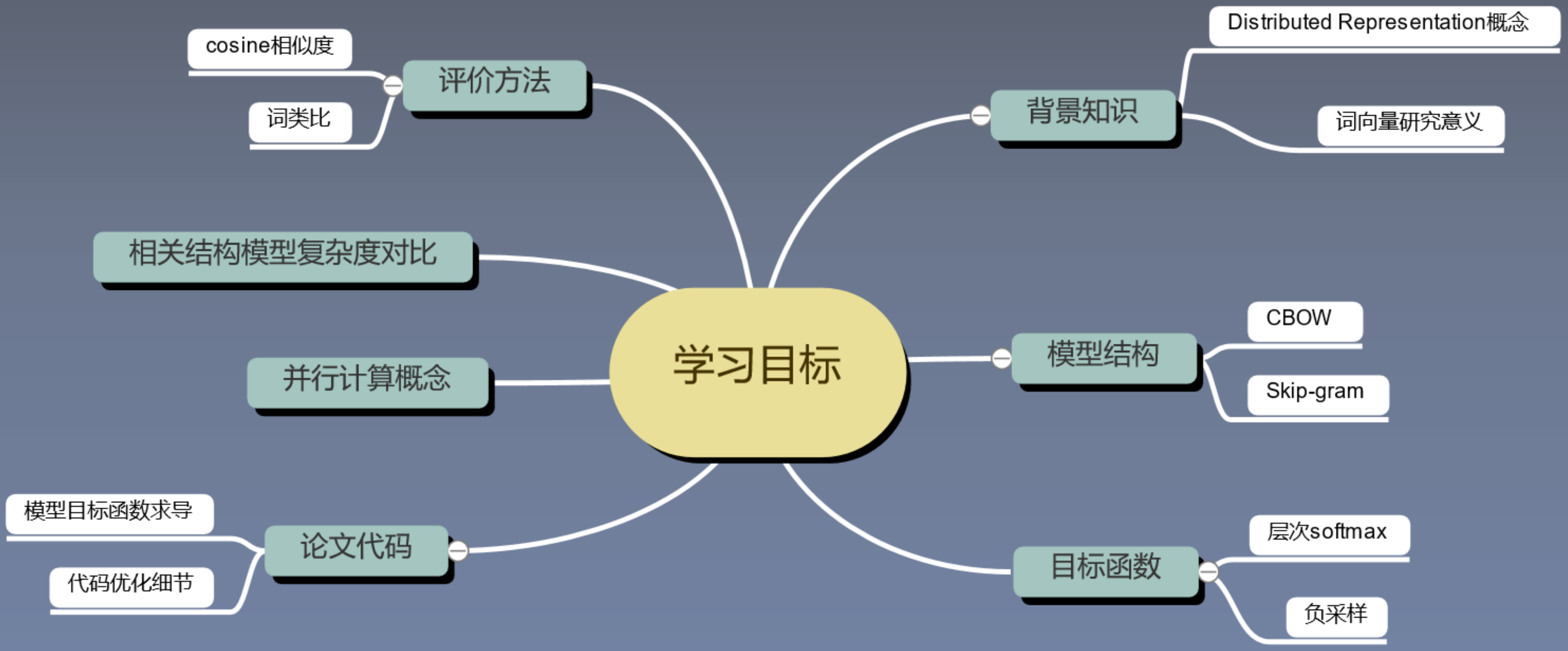
- 层次softmax
- 负采样

### 掌握Word2Vec代码

- 关键代码
- 提高效率细节

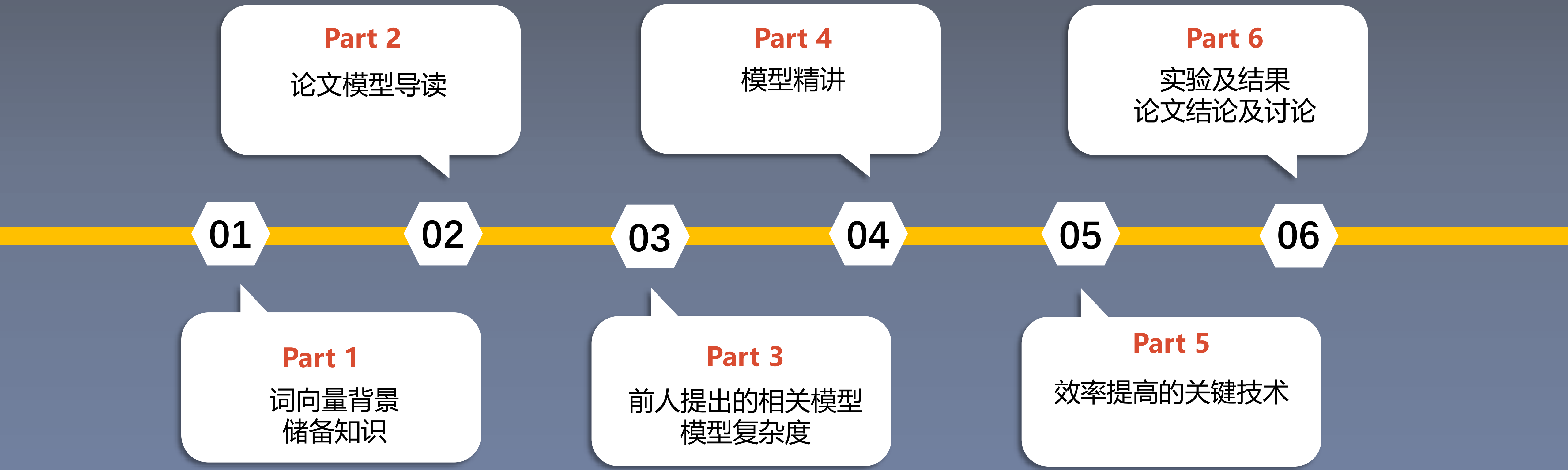
# 学习目标

Learning objectives



# 课程安排

The schedule of course



01

**Part 1**

词向量背景
储备知识

02

**Part 2**

论文模型导读

03

**Part 3**

前人提出的相关模型
模型复杂度

04

**Part 4**

模型精讲

05

**Part 5**

效率提高的关键技术

06

**Part 6**

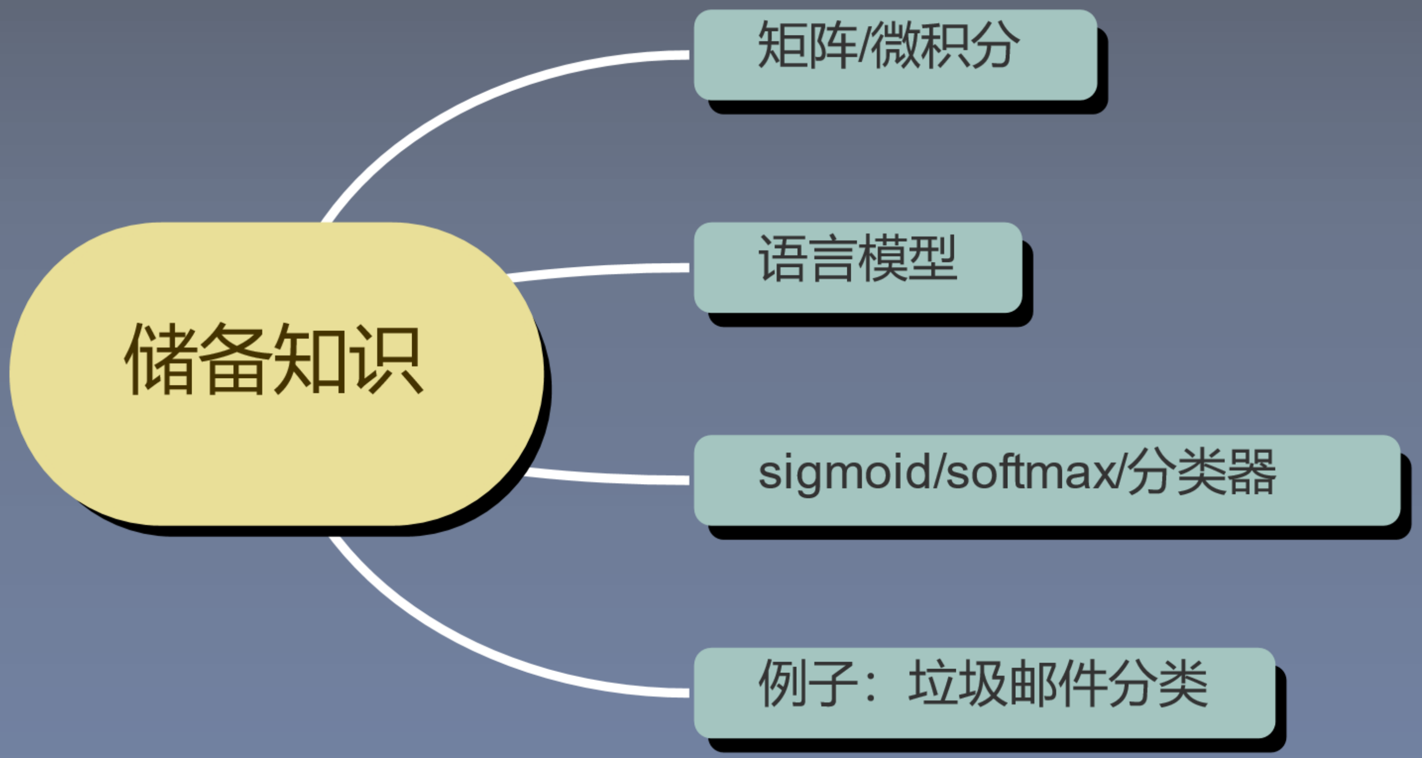
实验及结果
论文结论及讨论

深度之眼
deepshare.net

# 论文储备知识

Pre-knowledge

# 本章知识树



# 论文储备知识

Pre-knowledge

## 语言模型

重点　重点来了！

概率：语言模型是计算一个句子是是句子的概率。

深度之眼的论文课真的很好！　0.8

深度之眼的论文课真的很一般！　0.01

论文课的深度之眼很真好的！　0.000001

# 论文储备知识

Pre-knowledge

重点　重点来了！

## 语言模型

概率：语言模型是计算一个句子是是句子的概率。

| | |
|---|---|
| 深度之眼的论文课真的很好！ | 0.8 |
| 深度之眼的论文课真的很一般！ | 0.01 |
| 论文课的深度之眼很真好的！ | 0.000001 |

zi ran yu yan chu li对应的中文

| | |
|---|---|
| 自然语言处理 | 0.9 |
| 子然预言出力 | 0.01 |
| 紫然玉眼储例 | 0.0001 |

# 论文储备知识

Pre-knowledge

## 基于专家语法规则的语言模型

重点　重点来了！

语言学家企图总结出一套通用的语法规则

笑skr人!

这件事雨女无瓜。

# 论文储备知识

Pre-knowledge

重点　重点来了！

## 统计语言模型

通过概率计算来刻画语言模型

$$P(s) = P(w_1, w_2, \ldots, w_n) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1 w_2) \ldots \mathrm{P}(w_n|w_1 w_2 \ldots w_{n-1})$$

# 论文储备知识

Pre-knowledge

## 统计语言模型

重点　重点来了！

通过概率计算来刻画语言模型

$$P(s) = P(w_1, w_2, \dots, w_n) = \boldsymbol{P(w_1)}\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1w_2) \dots \mathrm{P}(w_n|w_1w_2 \dots w_{n-1})$$

**求解方法**：用语料的频率代替概率（频率学派）

$$p(w_i) = \frac{\mathrm{count}(w_i)}{N}$$

# 论文储备知识

Pre-knowledge

重点 重点来了！

## 统计语言模型

通过概率计算来刻画语言模型

$$P(s) = P(w_1, w_2, \ldots, w_n) = P(w_1)\mathbf{P(w_2|w_1)}\mathrm{P}(w_3|w_1w_2) \ldots \mathrm{P}(w_n|w_1w_2 \ldots w_{n-1})$$

**求解方法**：频率学派+条件概率

$$p(w_i|w_{i-1}) = \frac{p(w_{i-1},w_i)}{p(w_{i-1})}$$

$$p(w_{i-1}, w_i) = \frac{\mathrm{count}(w_{i-1}, w_i)}{N}$$

$$p(w_{i-1}) = \frac{\mathrm{count}(w_{i-1})}{N}$$

$$p(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$$

# 论文储备知识

## Pre-knowledge

重点 重点来了！

### 语言模型

$sentence \ = \{w_1, w_2, \ldots, w_n\}$

P(张三 很 帅)=P(张三)*P(很|张三)*P(帅|张三, 很)

P(张很帅 很 帅)=P(张很帅)*P(很|张很帅)*P(帅|张很帅, 很)

P(张三 很 漂亮)=P(张三)*P(很|张三)*P(漂亮|张三, 很)

# 论文储备知识

Pre-knowledge

## 统计语言模型中的平滑操作

重点 重点来了！

有一些词或者词组在语料中没有出现过，但是这不能代表它不可能存在。

平滑操作就是给那些没有出现过的词或者词组也给一个比较小的概率。

Laplace Smoothing也称为加1平滑：每个词在原来出现次数的基础上加1。

| | | | | | |
|---|---|---|---|---|---|
| A： | 0 | P(A)=0/1000=0 | A： | 1 | P(A)=1/1003=0.001 |
| B： | 990 | P(B)=990/1000=0.99 | B： | 991 | P(B)=991/1003=0.988 |
| C： | 10 | P(C)=10/1000=0.01 | C： | 11 | P(C)=11/1003=0.011 |

# 论文储备知识

Pre-knowledge

重点  重点来了！

## 平滑操作的问题

P(张三 很 帅)=P(张三)P(很|张三)P(帅|张三, 很)

P(张三 很 桌子)=P(张三)P(很|张三)P(桌子|张三, 很)

$$P(s) = P(w_1, w_2, \ldots, w_n) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1 w_2)\ldots\mathrm{P}(w_n|w_1 w_2 \ldots w_{n-1})$$

参数空间过大

数据稀疏严重

# 论文储备知识

Pre-knowledge

重点 重点来了！

## 马尔科夫假设

下一个词的出现仅依赖于前面的一个词或几个词

$$P(s) = P(w_1, w_2, \ldots, w_n) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1w_2)\ldots\mathrm{P}(w_n|w_1w_2\ldots w_{n-1})$$

unigram: $P(s) = P(w_1)\mathrm{P}(w_2)\mathrm{P}(w_3)\ldots\mathrm{P}(w_n)$

bigram: $P(s) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_2)\ldots\mathrm{P}(w_n|w_{n-1})$

trigram: $P(s) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1w_2)\ldots\mathrm{P}(w_n|w_{n-3}w_{n-2}w_{n-1})$

k-gram: $P(s) = P(w_1)\mathrm{P}(w_2|w_1)\mathrm{P}(w_3|w_1w_2)\ldots\mathrm{P}(w_n|w_{n-k}\ldots w_{n-1})$

# 论文储备知识

Pre-knowledge

## 语言模型

例子 我 今天 下午 打 羽毛球

$$\mathrm{p}(sentence) = p(\text{我，今天，下午，打，羽毛球})$$

$$= p(\text{我})p(\text{今天}|\text{我})p(\text{下午}|\text{今天，我})p(\text{打}|\text{下午，今天，我})$$
$$p(\text{羽毛球}|\text{我，今天，下午，打})$$

$$= p(\text{我})p(\text{今天}|\text{我})p(\text{下午}|\text{今天})p(\text{打}|\text{下午})p(\text{羽毛球}|\text{打})$$

# 论文储备知识

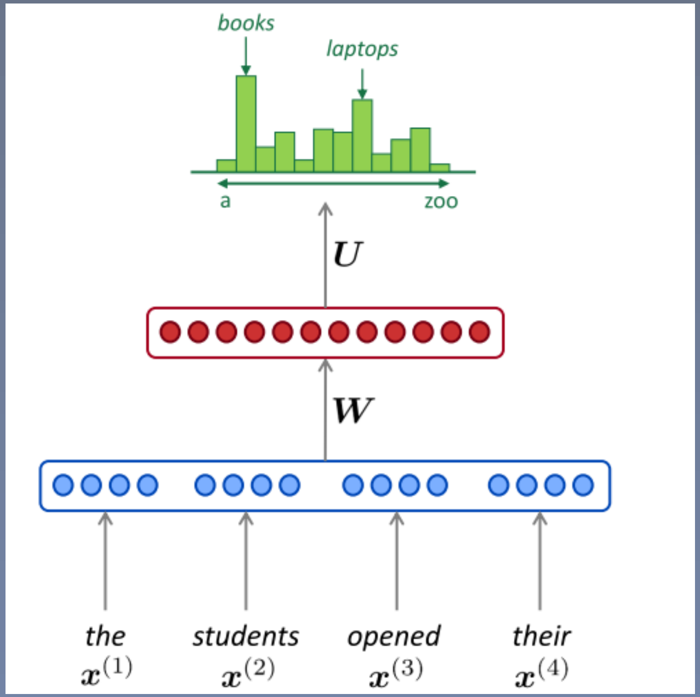
Pre-knowledge

## 语言模型评价指标：困惑度(Perplexity)

$$P(s) = P(w_1, w_2, \dots, w_n) = P(w_1)\mathrm{P}(w_2|w_1)\dots\mathrm{P}(w_n|w_1 w_2 \dots w_{n-1})$$

$$PP(s) = P(w_1, w_2, \dots, w_n)^{-\frac{1}{n}} = \sqrt[N]{\frac{1}{P(w_1, w_2, \dots, w_n)}}$$

句子概率越大，语言模型越好，困惑度越小。



# 第一课：论文导读

The first lesson: the paper guide

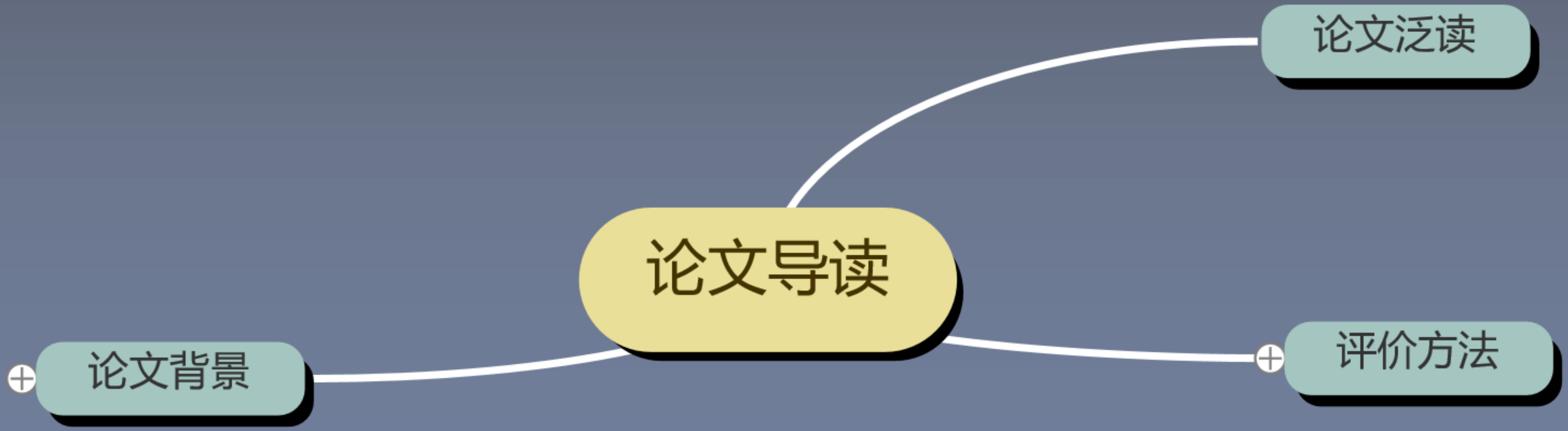
# 目录

1 论文背景知识

2 论文泛读

3 本节回顾及下节预告

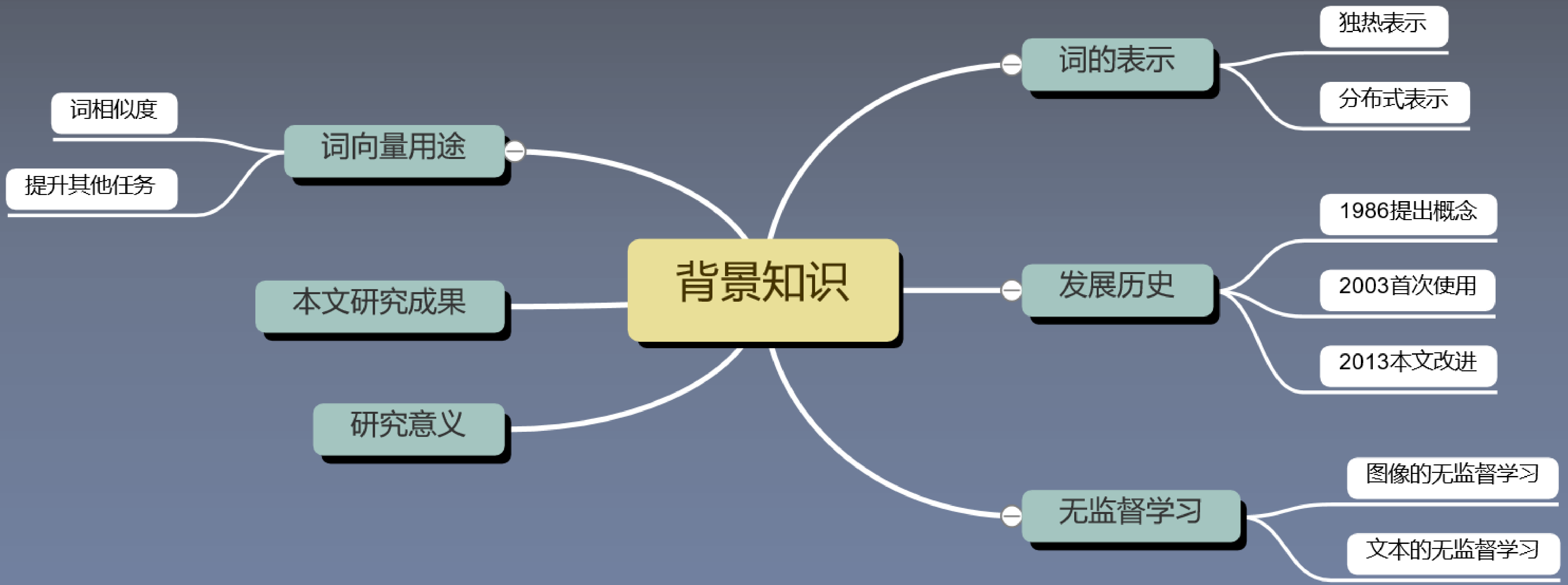
# 本课知识树



# 论文背景知识

background

# 论文背景知识树



背景知识

词向量用途
- 词相似度
- 提升其他任务

本文研究成果

研究意义

词的表示
- 独热表示
- 分布式表示

发展历史
- 1986提出概念
- 2003首次使用
- 2013本文改进

无监督学习
- 图像的无监督学习
- 文本的无监督学习

# 论文背景知识

background

## 词的表示方式
## Word representation

One-hot Representation

独热表示

“话筒”表示为 [0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ...]
“麦克”表示为 [0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 ...]

表示简单

问题：词越多，维数越高（词表大小V）
无法表示词和词之间关系

稀疏矩阵表示方法

“话筒”3:1
“麦克”8:1

# 论文背景知识

background

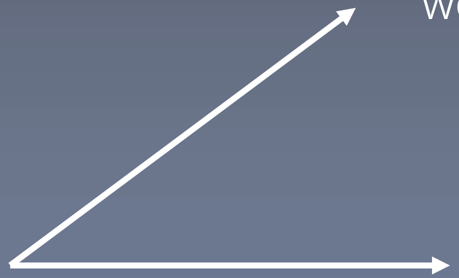
## 词的表示方式
## Word representation

word

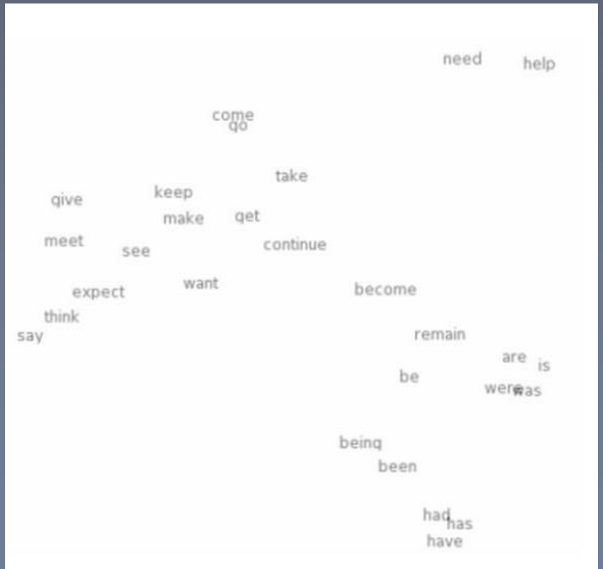
维度D

D<<V

$$\begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$



可以表示词和词之间的关系



Distributed Representation
分布式表示/稠密表示

Word embedding
词向量/词嵌入

# 论文背景知识

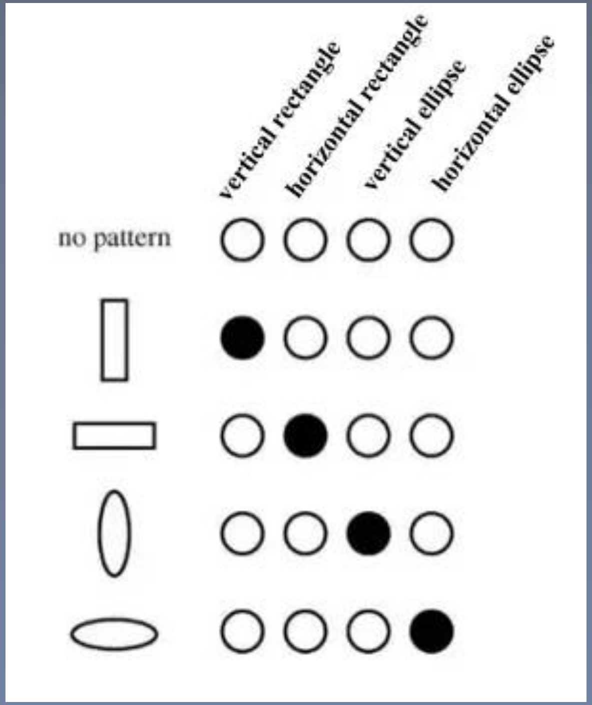
background

**词的表示方式**

**Word representation**

## One-hot Representation

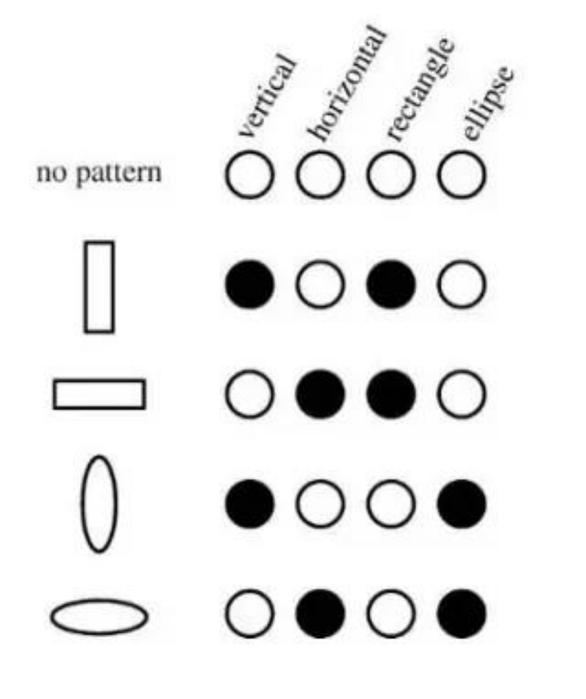
独热表示



|  | vertical rectangle | horizontal rectangle | vertical ellipse | horizontal ellipse |
|---|---|---|---|---|
| no pattern | ○ | ○ | ○ | ○ |
|  | ● | ○ | ○ | ○ |
|  | ○ | ● | ○ | ○ |
|  | ○ | ○ | ● | ○ |
|  | ○ | ○ | ○ | ● |

## Distributed Representation

分布式表示/稠密表示



|  | vertical | horizontal | rectangle | ellipse |
|---|---|---|---|---|
| no pattern | ○ | ○ | ○ | ○ |
|  | ● | ○ | ● | ○ |
|  | ○ | ● | ● | ○ |
|  | ● | ○ | ○ | ● |
|  | ○ | ● | ○ | ● |

https://www.quora.com/Deep-Learning-What-is-meant-by-a-distributed-representation

# 论文背景知识

background

## 发展历程

1986

Hinton
1986年提出
Distributed
Representation

2003

首次使用词向量
A Neural Probabilistic
Language Model

2013

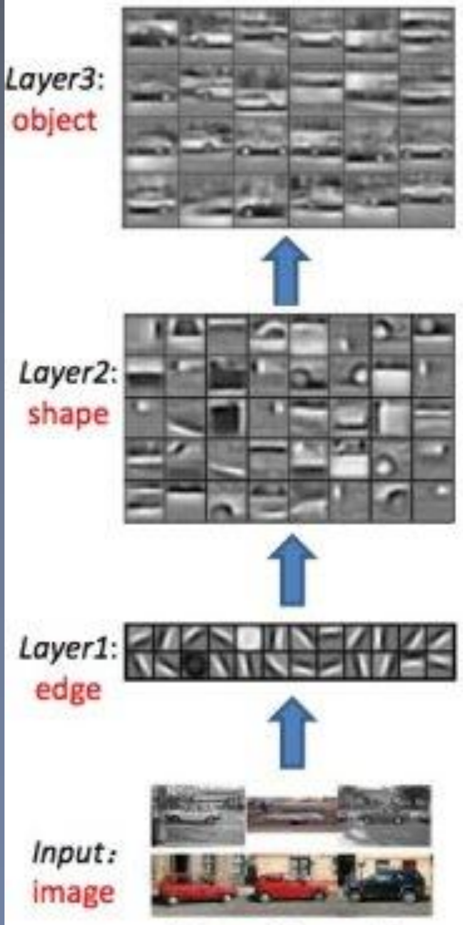
本文改进

开源word2vec

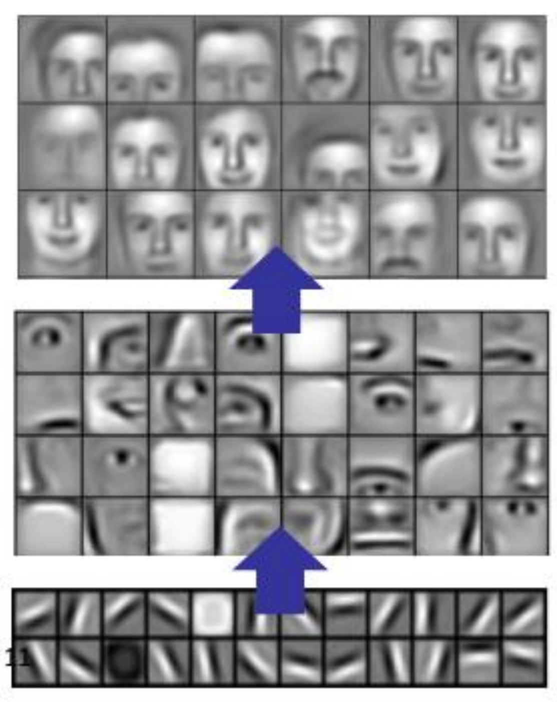
# 论文背景知识

background

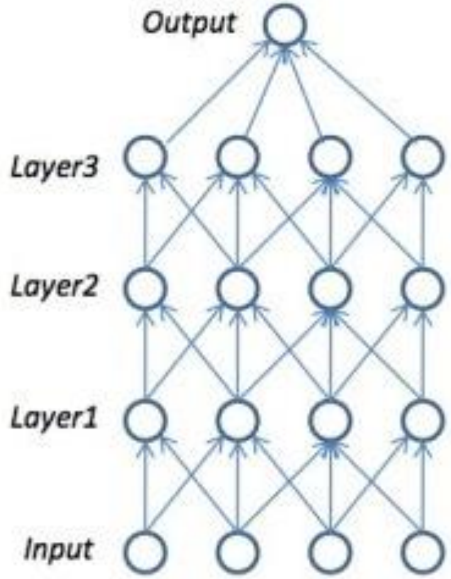
无监督学习：图像



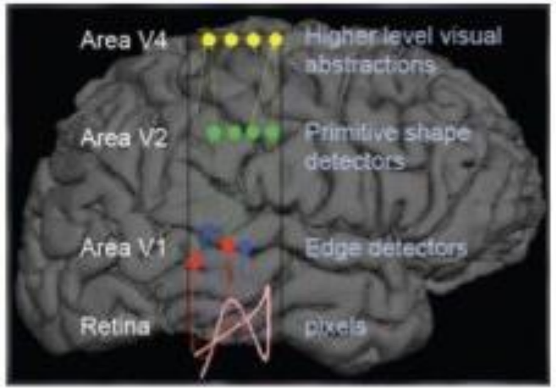
Deep Learning
输入、中间结果



there are approximately $10^{11}$ neurons in the human brain and between $10^{14}$ and $10^{15}$ synapses.



Deep Learning
数学表示



人类视觉

Lee, Grosse, Ranganath, Ng. ICML 09.

# 论文背景知识

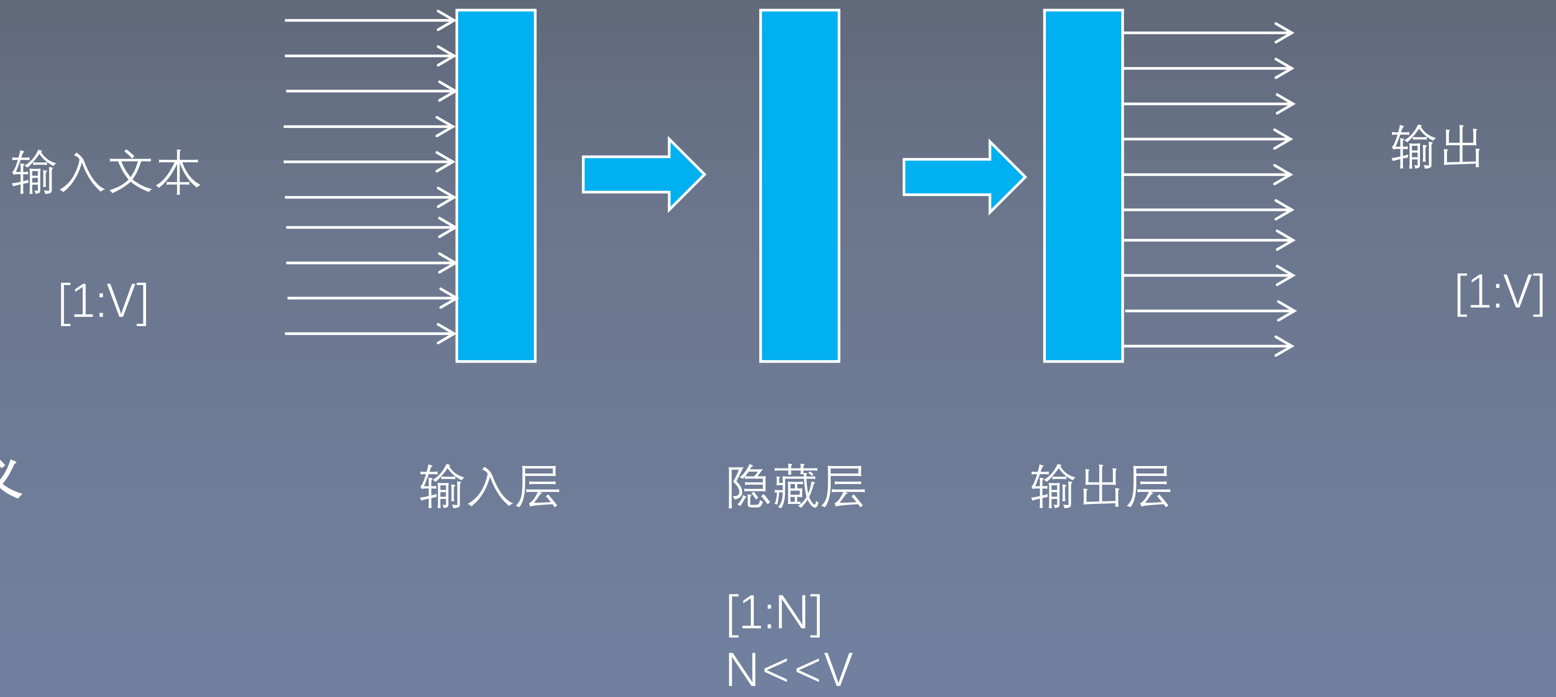
background

**无监督学习：文本**

**无监督学习**

**压缩自编码**

**结合上下文语义**



尝试自我学习，得到词向量

# 研究成果

- 提出新的模型结构
- 提出优化训练的方法，使得训练速度加快
- 给出训练代码word2vec，使得单机训练成为可能
- 成果：训练的词向量，又快又好

# 研究意义

## Research Meaning

**衡量词向量之间的相似程度**

$$\mathrm{sim}(word1, word2) = \cos(wordvec1, vordvec2)$$

*1.frog*
2.frogs
3.toad
4.litoria
5.leptodactylidae
6.rana
7.lizard
8.eleutherodactylus

3. litoria

4. leptodactylidae

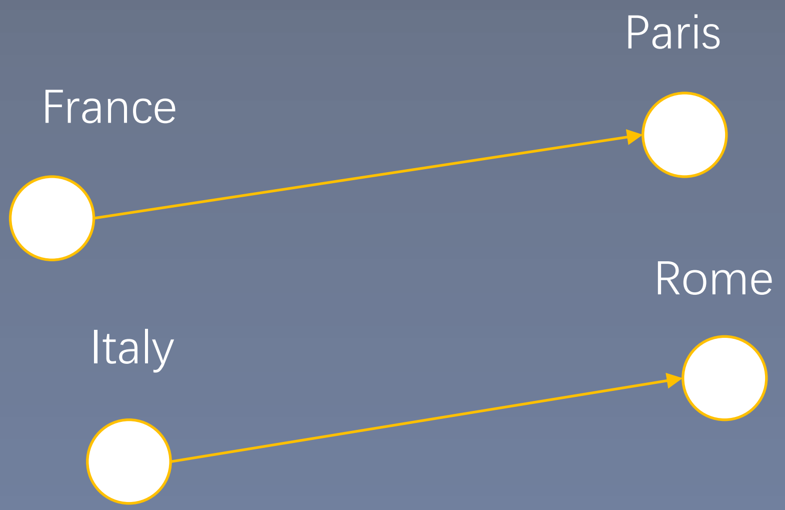
5. rana

7. eleutherodactylus

https://nlp.stanford.edu/projects/glove/

**词类比analogy**

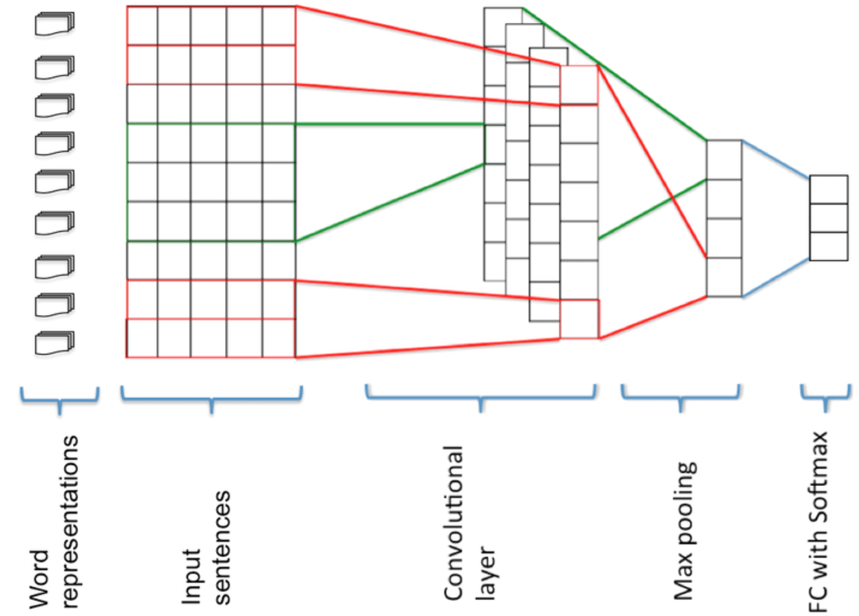
$$\cos(word1 - word2 + word3, vordvec4)$$



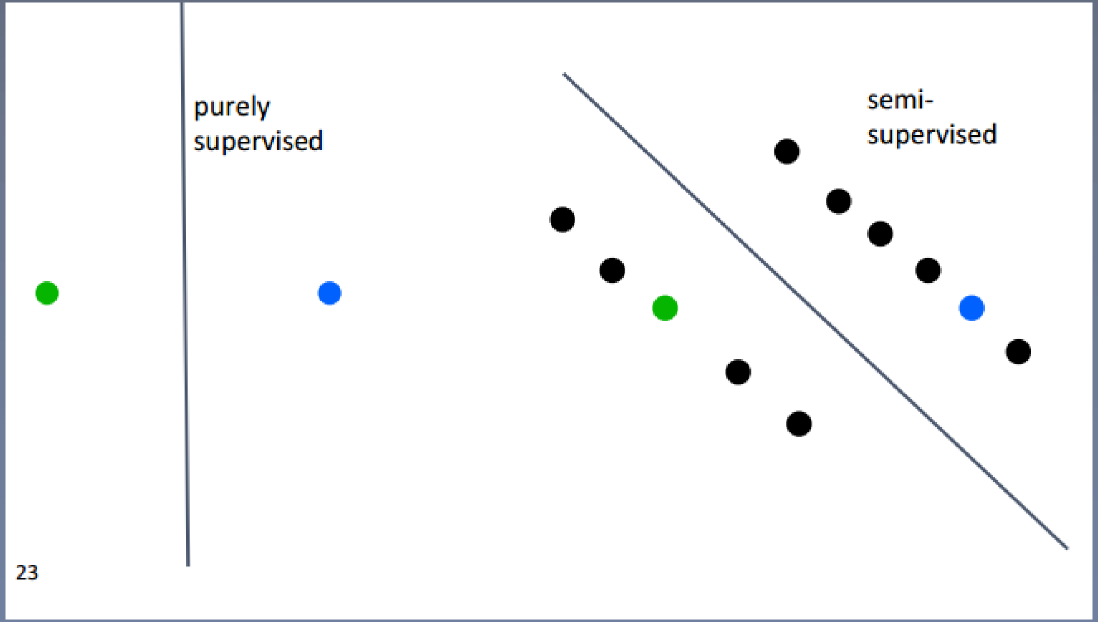
# 研究意义

Research Meaning

**作为预训练模型提升nlp任务**



外部任务比如命名实体识别、文本分类



应用到其他NLP任务上 相当于半监督训练

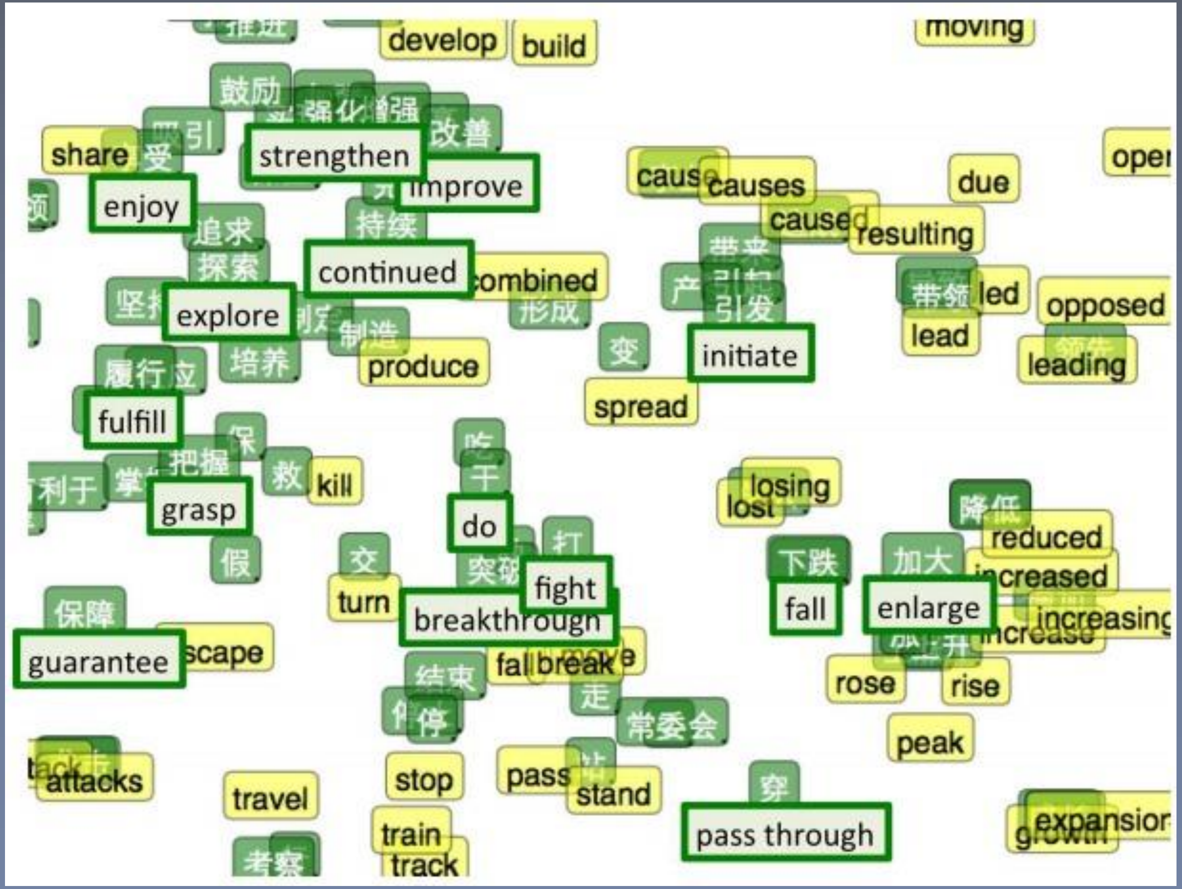
图片来源：清华刘知远分享ppt

# 研究意义

## Research Meaning

双语单词嵌入

Word embedding



Socher *et al.* (2013a)

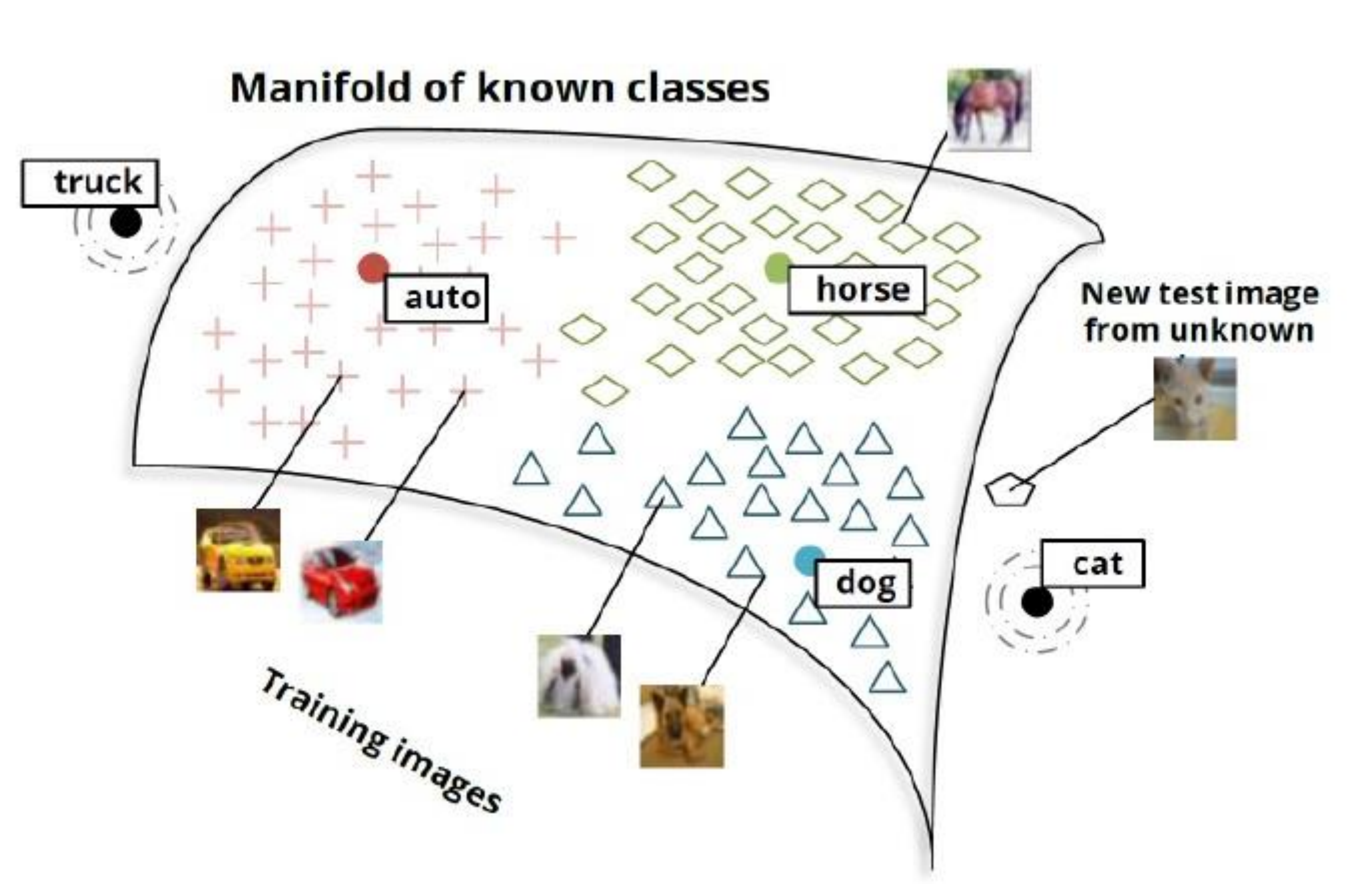
https://nlp.stanford.edu/~socherr/SocherGanjooManningNg_NIPS2013.pdf

# 研究意义

Research Meaning

图像-文字嵌入

Manifold 流形



Socher *et al.* (2013a)

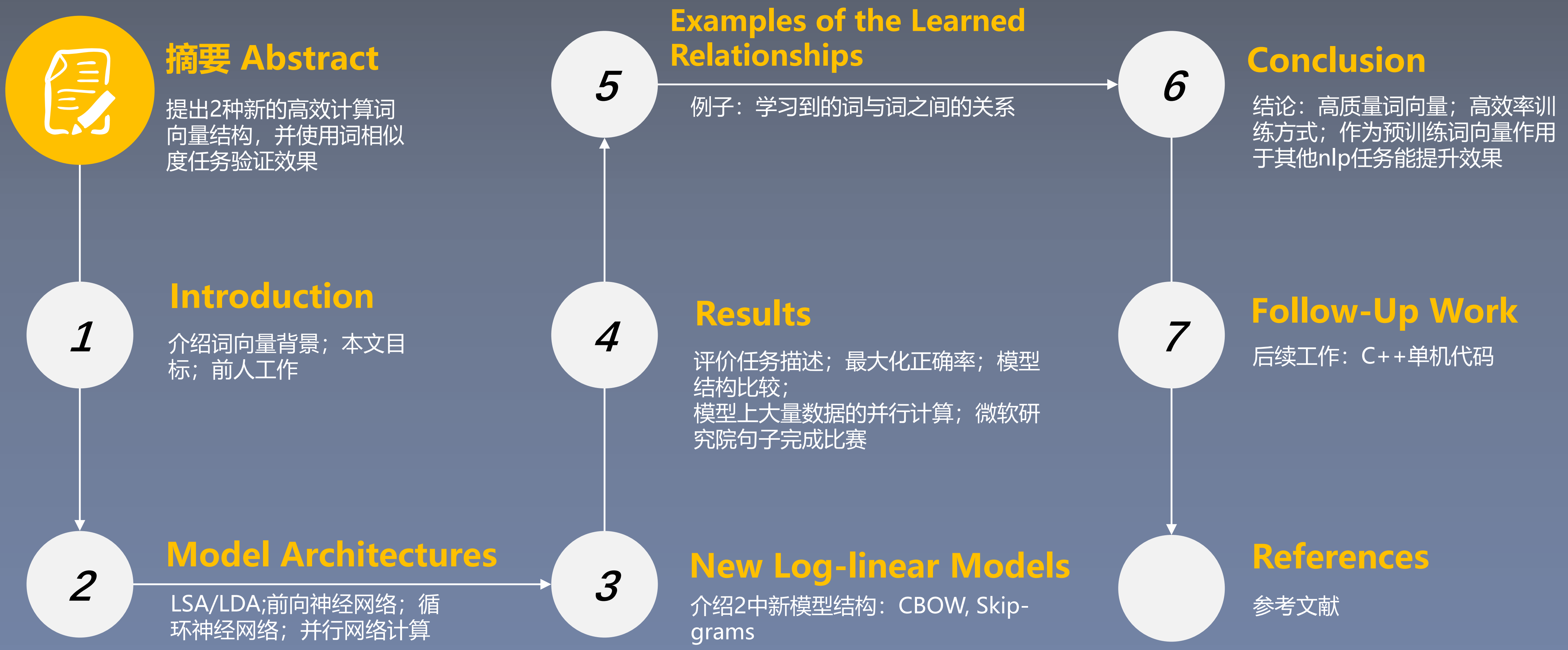
深度之眼
deepshare.net

# 论文泛读

fast guide

# 论文总览

## Summary of Papers



### 摘要 Abstract

提出2种新的高效计算词向量结构，并使用词相似度任务验证效果

1

### Introduction

介绍词向量背景；本文目标；前人工作

2

### Model Architectures

LSA/LDA;前向神经网络；循环神经网络；并行网络计算

3

### New Log-linear Models

介绍2中新模型结构：CBOW, Skip-grams

4

### Results

评价任务描述；最大化正确率；模型结构比较；
模型上大量数据的并行计算；微软研究院句子完成比赛

5

### Examples of the Learned Relationships

例子：学习到的词与词之间的关系

6

### Conclusion

结论：高质量词向量；高效率训练方式；作为预训练词向量作用于其他nlp任务能提升效果

7

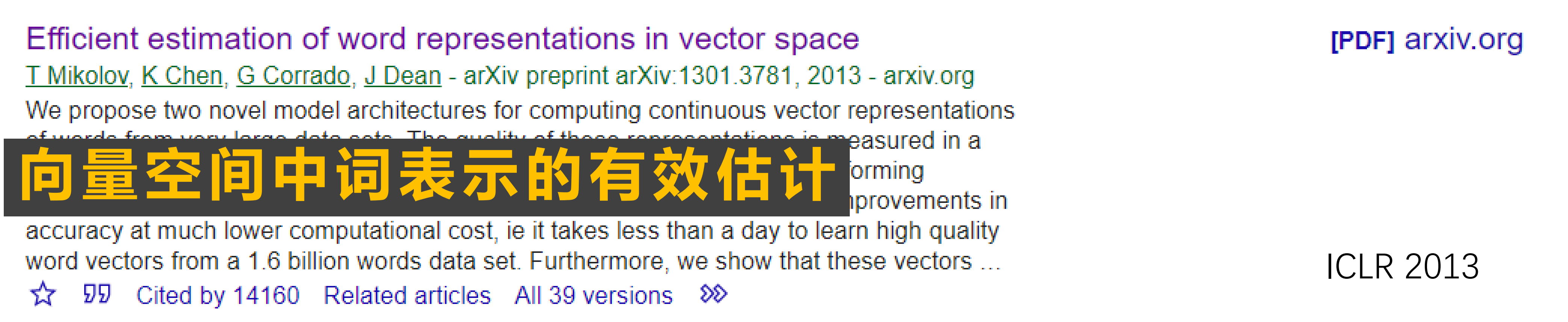
### Follow-Up Work

后续工作：C++单机代码

### References

参考文献

# 实际讲2篇论文



Efficient estimation of word representations in vector space

[PDF] arxiv.org

T Mikolov, K Chen, G Corrado, J Dean - arXiv preprint arXiv:1301.3781, 2013 - arxiv.org

We propose two novel model architectures for computing continuous vector representations of words from very large data sets. The quality of these representations is measured in a

向量空间中词表示的有效估计

forming

nprovements in

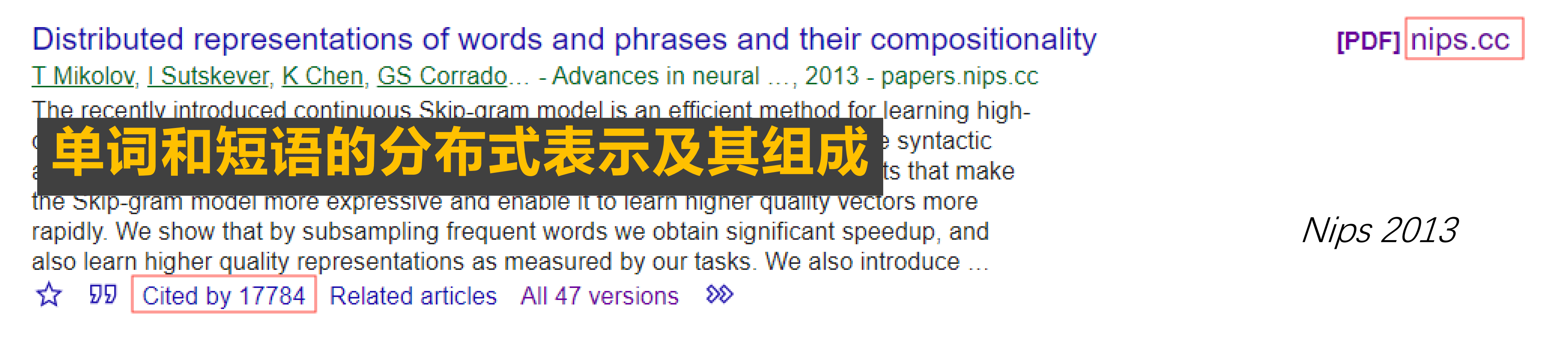
accuracy at much lower computational cost, ie it takes less than a day to learn high quality word vectors from a 1.6 billion words data set. Furthermore, we show that these vectors …

☆ 99 Cited by 14160 Related articles All 39 versions »

ICLR 2013

背景

其他人的工作



Distributed representations of words and phrases and their compositionality

[PDF] nips.cc

T Mikolov, I Sutskever, K Chen, GS Corrado… - Advances in neural …, 2013 - papers.nips.cc

The recently introduced continuous Skip-gram model is an efficient method for learning high-

单词和短语的分布式表示及其组成

syntactic

ts that make

the Skip-gram model more expressive and enable it to learn higher quality vectors more rapidly. We show that by subsampling frequent words we obtain significant speedup, and also learn higher quality representations as measured by our tasks. We also introduce …

☆ 99 Cited by 17784 Related articles All 47 versions »

*Nips 2013*

数学原理

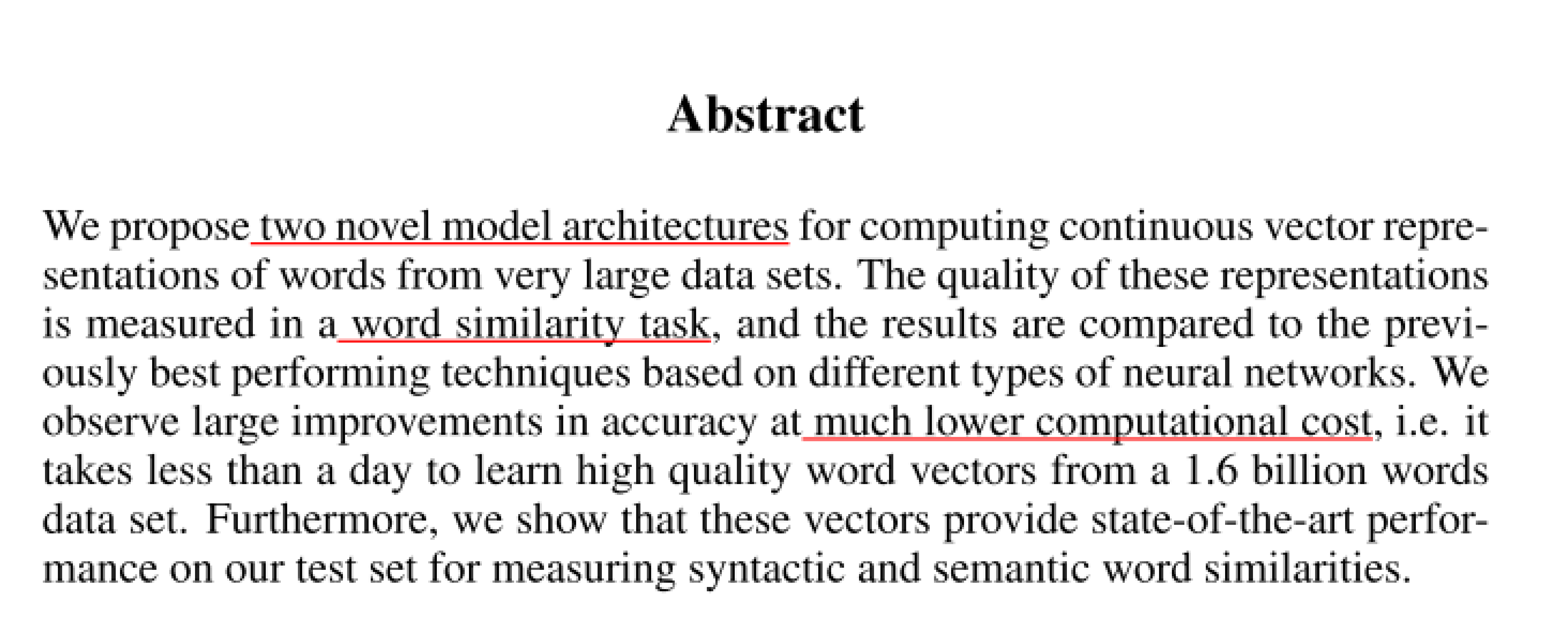
# 摘要

abstract

## 摘要核心

1. 提出了两种新颖的模型结构用来计算词向量
2. 采用一种词相似度的任务来评估对比词向量质量
3. 大量降低模型计算量可以提升词向量质量
4. 进一步，在我们的语义和句法任务上，我们的词向量是当前最好的效果

# 摘要



## Abstract

We propose two novel model architectures for computing continuous vector representations of words from very large data sets. The quality of these representations is measured in a word similarity task, and the results are compared to the previously best performing techniques based on different types of neural networks. We observe large improvements in accuracy at much lower computational cost, i.e. it takes less than a day to learn high quality word vectors from a 1.6 billion words data set. Furthermore, we show that these vectors provide state-of-the-art performance on our test set for measuring syntactic and semantic word similarities.

# 论文小标题

## Paper title

1. Introduction

    1.1 Goals of the paper

    1.2 Previous Work

2. Model Architectures

    2.1 Feedforward Neural Net Language Model(NNLM)

    2.2 Recurrent Neural Net Language Model(RNNLM)

    2.3 Parallel Training of Neural networks

3 New Log-linear Models

    3.1 Continuous Bag-of-Words Model

    3.2 Continuous Skip-gram Model

4 Results

    4.1 Task Description

    4.2 Maximization of Accuracy

    4.3 Comparison of Model Architectures

    4.4 Large Scale Parallel Training of Models

    4.5 Microsoft Research Sentence Completion Challenge

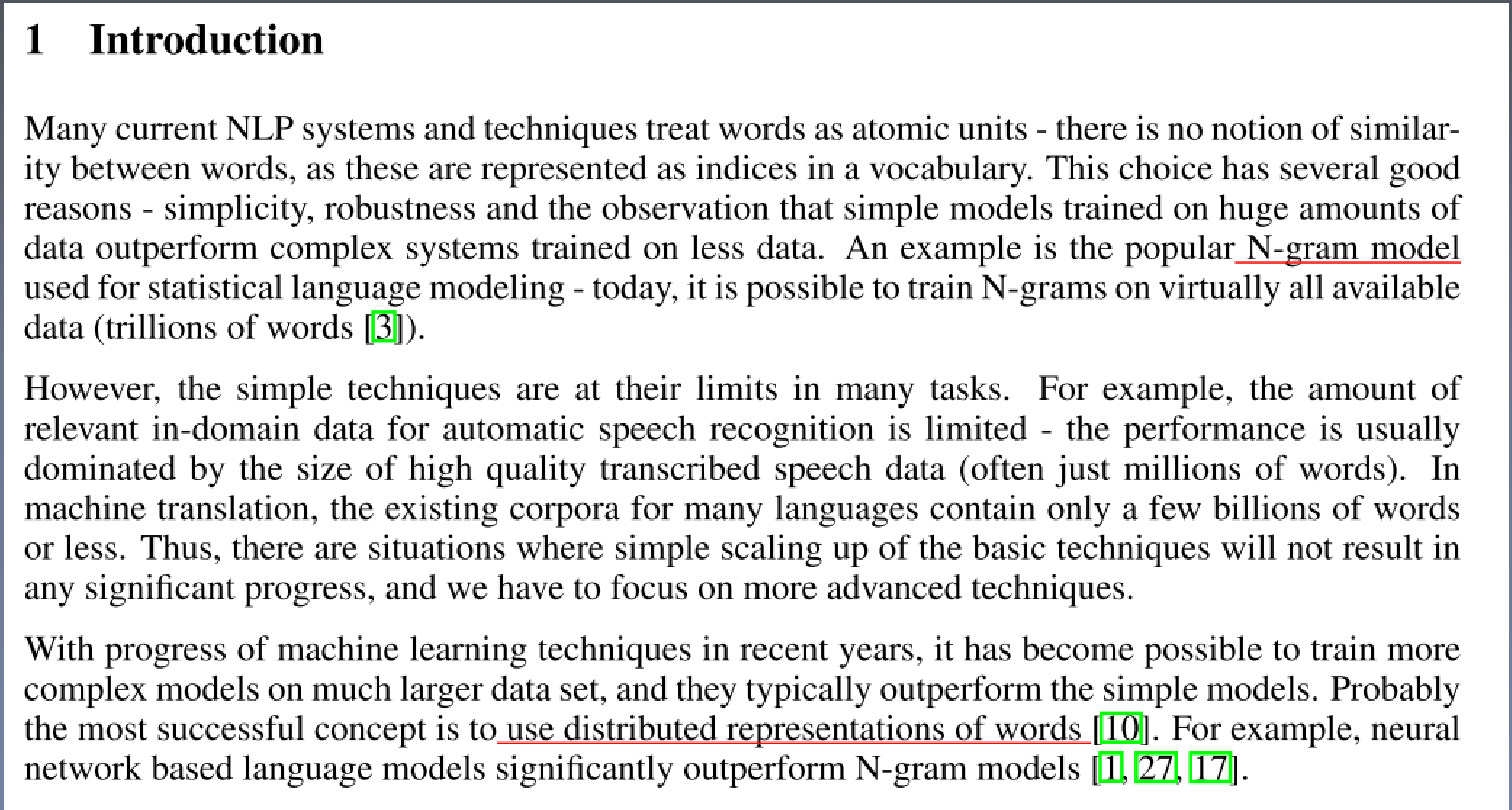
5 Examples of the Learned Relationships

6 Conclusion

# 介绍

## Introduction

1. 传统NLP把词当成最小单元处理，其中一个例子是N-gram
2. 然而这种方法在许多任务中有其局限性，如语音识别、机器翻译
3. 数据量较大时，可以采用分布式表示方法，如语言模型的分布式表示效果会超过N-gram



## 1 Introduction

Many current NLP systems and techniques treat words as atomic units - there is no notion of similarity between words, as these are represented as indices in a vocabulary. This choice has several good reasons - simplicity, robustness and the observation that simple models trained on huge amounts of data outperform complex systems trained on less data. An example is the popular N-gram model used for statistical language modeling - today, it is possible to train N-grams on virtually all available data (trillions of words [3]).

However, the simple techniques are at their limits in many tasks. For example, the amount of relevant in-domain data for automatic speech recognition is limited - the performance is usually dominated by the size of high quality transcribed speech data (often just millions of words). In machine translation, the existing corpora for many languages contain only a few billions of words or less. Thus, there are situations where simple scaling up of the basic techniques will not result in any significant progress, and we have to focus on more advanced techniques.

With progress of machine learning techniques in recent years, it has become possible to train more complex models on much larger data set, and they typically outperform the simple models. Probably the most successful concept is to use distributed representations of words [10]. For example, neural network based language models significantly outperform N-gram models [1, 27, 17].

# Word2vec 评价方法

## 内部任务评价

衡量词向量之间的相似程度

$$\mathrm{sim}(word1, word2) = \cos(wordvec1, vordvec2)$$

***1.frog***
**2.frogs**
**3.toad**
**4.litoria**
**5.leptodactylidae**
**6.rana**
**7.lizard**
**8.eleutherodactylus**

3. litoria

4. leptodactylidae

5. rana

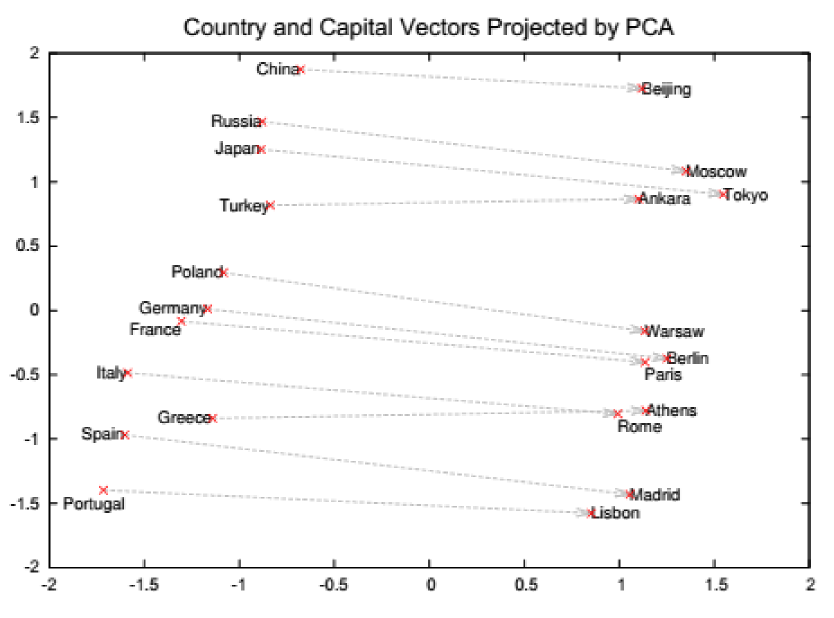
7. eleutherodactylus

https://nlp.stanford.edu/projects/glove/

### 词类比analogy

$$\cos(word1 - word2 + word3, vordvec4)$$

W(“woman")−W(“man") ≃ W(“aunt")−W(“uncle")
W(“woman")−W(“man") ≃ W(“queen")−W(“king")


Country and Capital Vectors Projected by PCA

# Word2vec 评价方法

## 内部任务评价

衡量词向量之间的相似程度

$$\mathrm{sim}(word1, word2) = \cos(wordvec1, vordvec2)$$

***1.frog***

**2.frogs**

**3.toad**

**4.litoria**

**5.leptodactylidae**

**6.rana**

**7.lizard**

**8.eleutherodactylus**

3. litoria

4. leptodactylidae

5. rana

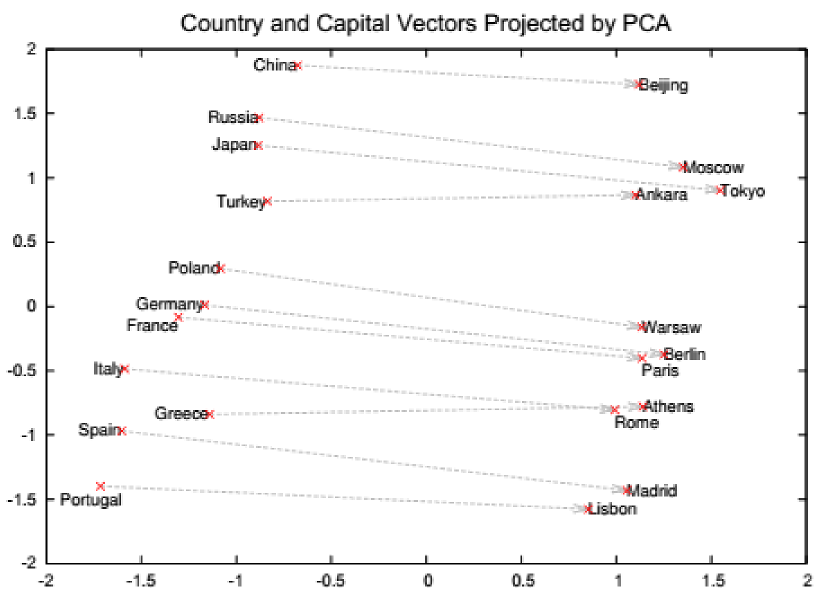
7. eleutherodactylus

https://nlp.stanford.edu/projects/glove/

### 词类比analogy

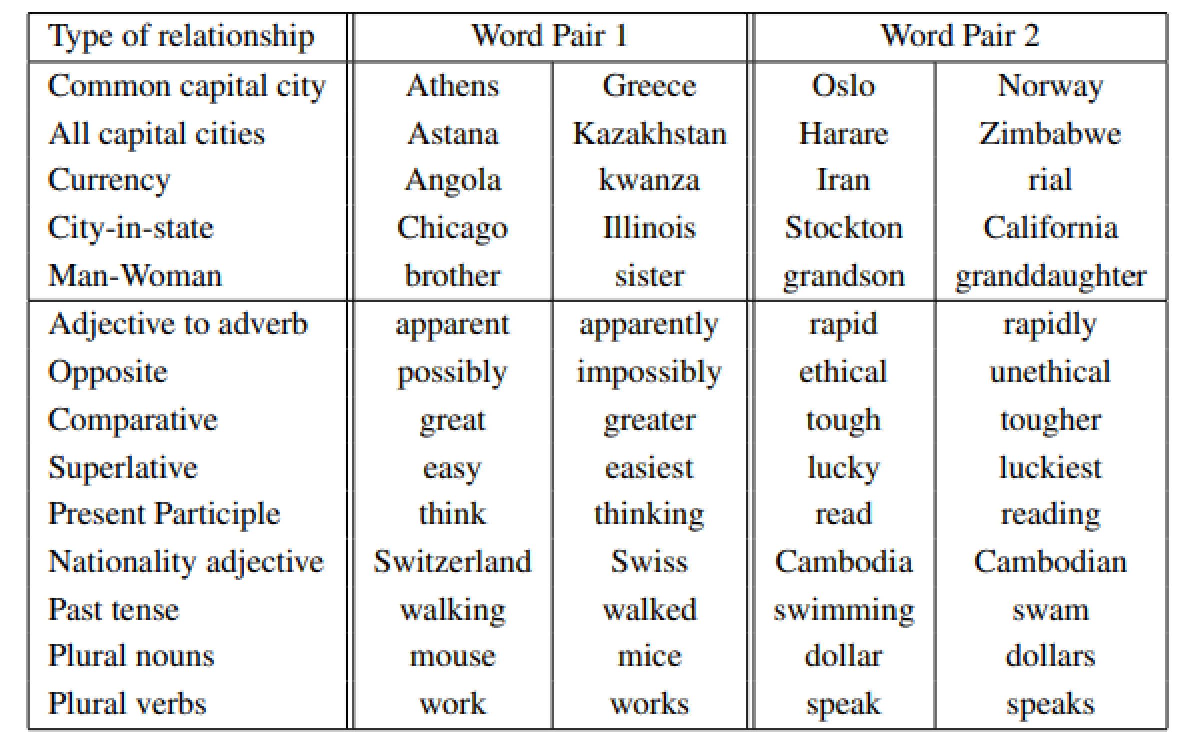
$$\cos(word1 - word2 + word3, vordvec4)$$

W(“woman")−W(“man") ≃ W(“aunt")−W(“uncle")
W(“woman")−W(“man") ≃ W(“queen")−W(“king")



# Word2vec 评价方法

## 内部任务评价



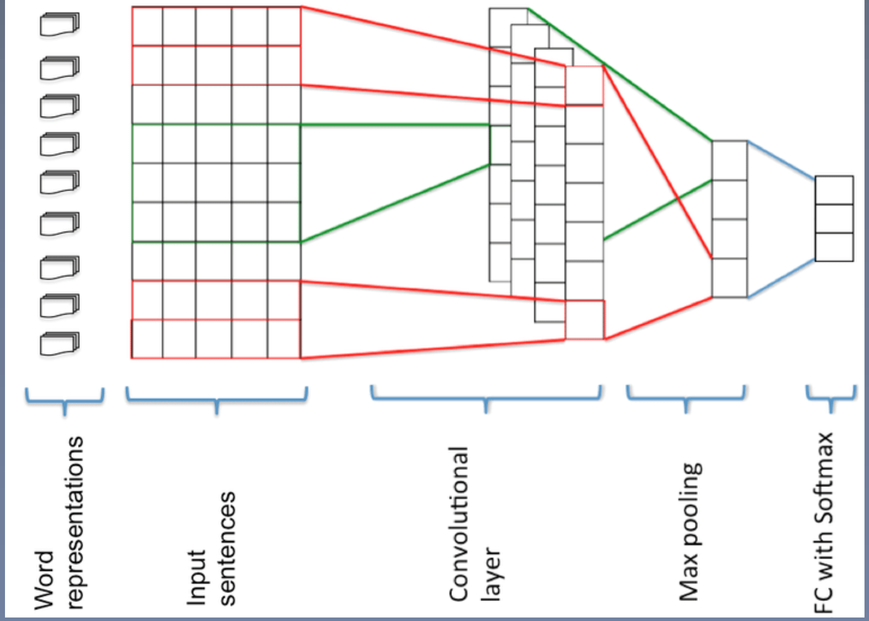
| Type of relationship | Word Pair 1 | | Word Pair 2 | |
|---|---|---|---|---|
| Common capital city | Athens | Greece | Oslo | Norway |
| All capital cities | Astana | Kazakhstan | Harare | Zimbabwe |
| Currency | Angola | kwanza | Iran | rial |
| City-in-state | Chicago | Illinois | Stockton | California |
| Man-Woman | brother | sister | grandson | granddaughter |
| Adjective to adverb | apparent | apparently | rapid | rapidly |
| Opposite | possibly | impossibly | ethical | unethical |
| Comparative | great | greater | tough | tougher |
| Superlative | easy | easiest | lucky | luckiest |
| Present Participle | think | thinking | read | reading |
| Nationality adjective | Switzerland | Swiss | Cambodia | Cambodian |
| Past tense | walking | walked | swimming | swam |
| Plural nouns | mouse | mice | dollar | dollars |
| Plural verbs | work | works | speak | speaks |

http://www.fit.vutbr.cz/~imikolov/rnnlm/word-test.v1.txt

# Word2vec评价方法

## 外部任务评价

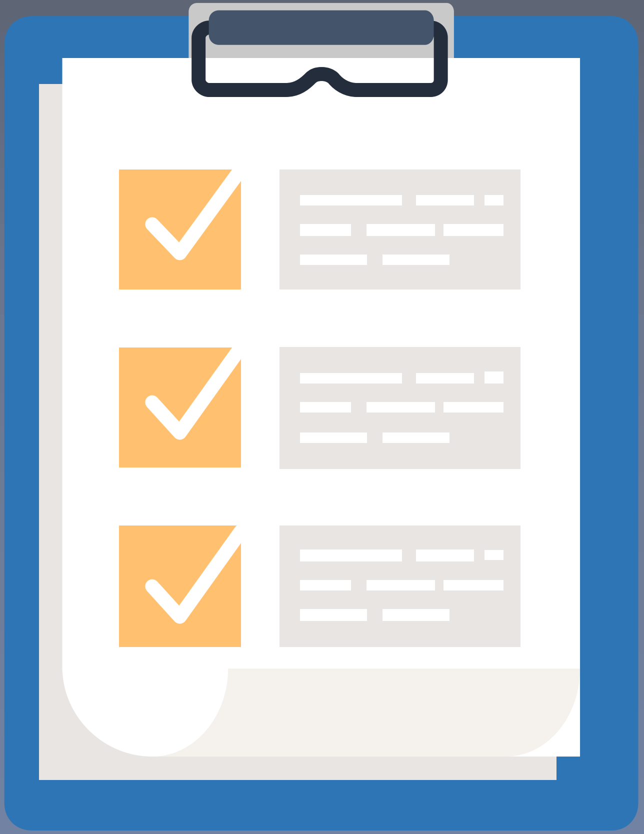
外部任务比如命名实体识别、文本分类



深度之眼
deepshare.net

# 本课回顾

Review in the lesson



## 01 论文背景知识

分布式表示概念，发展历史，研究意义

## 02 储备知识

数学知识、语言模型，机器学习等

## 03 论文泛读

论文泛读，阅读摘要和小标题

## 04 词向量评价

内部评价方法，外部任务评价

# 下期预告

Preview of next lesson

## 01 介绍word2vec之前的相关结构

对论文中提到的previous work的结构进行介绍

## 02 介绍模型并行计算概念

对论文中提到的并行计算概念进行详细阐述

## 03 介绍模型复杂度概念

介绍如何计算模型复杂度，并对论文中提到的所有模型计算模型复杂度

## 04 对各个模型进行精讲

对论文的2个模型的各个算法进行详细讲解，并推导概率公式、损失函数公式

# 下节课前准备

Preview of next lesson

- 下载论文
- 泛读论文
- 筛选出自己不懂的部分，带着问题进入下一课时

深度之眼
deepshare.net

联系我们:

电话：18001992849

邮箱：service@deepshare.net

Q Q： 2677693114

公众号

客服微信