Kaggle Quora Question Pairs

Can you identify question pairs that have the same intent?

Featured Prediction Competition

Quora Question Pairs

Can you identify question pairs that have the same intent?

$25,000
Prize Money

Quora · 3,307 teams · 9 days ago

Overview Data Kernels Discussion Leaderboard More My Submissions Submit Predictions

Your most recent submission

Name	Submitted	Wait time	Execution time	Score
all.fea.out.submit	9 days ago	30 seconds	19 seconds	0.52127

Complete

Jump to your position on the leaderboard

Public Leaderboard Private Leaderboard

The private leaderboard is calculated with approximately 65% of the test data.

This competition has completed. This leaderboard reflects the final standings.

Refresh

In the money Gold Silver Bronze

#	△pub	Team Name	Kernel	Team Members	Score	Entries	Last
1	—	DL guys			0.11580	263	9d
2	—	Depp Learning			0.11670	196	9d
3	—	Jared Turkewitz & sjv			0.11756	178	9d
4	—	YesOfCourse			0.11768	189	9d
5	—	Qingchen | KazAnova | Faron			0.11851	219	9d
6	—	LAMAA power			0.11887	406	9d
7	▲2	aphex34			0.12072	166	9d

任务路线

1. Features
2. Models
3. Rescaling
4. Stacking

关注的点

- Evaluation

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}\log(p_{i,j})$$

where N is the number of observations, M is the number of class labels, log is the natural logarithm, $y_{i,j}$ is 1 if observation i is in class j and 0 otherwise, and $p_{i,j}$ is the predicted probability that observation i is in class j.

- Siamese LSTM with pretrained Glove embedding

孪生网络(Siamese network)是一种网络结构，通过一个NN将样本的维度降低到某个较低的维度。 首先我们构造随机样本对，这样的样本对在这个较低的维度下，拥有如下特点：

* 如果样本对的标记是一致，那么这个低维度下的欧式距离很近

* 如果样本对的标记是不一致，那么这个低维度下的欧式距离保持在一个margin之上

- Decomposable attention (https://arxiv.org/abs/1606.01933) with pretrained FastText embedding. This model achieve ~0.3 on cv

A Decomposable Attention Model for Natural Language Inference

Ankur P. Parikh
Google
New York, NY

Oscar Täckström
Google
New York, NY

Dipanjan Das
Google
New York, NY

Jakob Uszkoreit
Google
Mountain View, CA

{aparikh,oscart,dipanjand,uszkoreit}@google.com

Abstract

We propose a simple neural architecture for natural language inference. Our approach uses attention to decompose the problem into subproblems that can be solved separately, thus making it trivially parallelizable. On the Stanford Natural Language Inference (SNLI) dataset, we obtain state-of-the-art results with almost an order of magnitude fewer parameters than previous work and without relying on any word-order information. Adding intra-sentence attention that takes a minimum amount of order into account yields further improvements.

1 Introduction

LSTMs henceforth) with the goal of deeper sentence comprehension. While these approaches have yielded impressive results, they are often computationally very expensive, and result in models having millions of parameters (excluding embeddings).

Here, we take a different approach, arguing that for natural language inference it can often suffice to simply align bits of local text substructure and then aggregate this information. For example, consider the following sentences:

- *Bob is in his room, but because of the thunder and lightning outside, he cannot sleep.*
- *Bob is awake.*
- *It is sunny outside.*

- ESIM (https://arxiv.org/abs/1609.06038) with pretrained FastText embedding. This is our best pure Deep Learning NLP model, it achieves ~0.27 on CV. However this model take too long to run, we only add it once in the first stacking layer

Enhanced LSTM for Natural Language Inference

Qian Chen
University of Science and Technology of China
cq1231@mail.ustc.edu.cn

Xiaodan Zhu
National Research Council Canada
xiaodan.zhu@nrc-cnrc.gc.ca

Zhenhua Ling
University of Science and Technology of China
zhling@ustc.edu.cn

Si Wei
iFLYTEK Research
siwei@iflytek.com

Hui Jiang
York University
hj@cse.yorku.ca

Diana Inkpen
University of Ottawa
diana@site.uottawa.ca

Abstract

Reasoning and inference are central to human and artificial intelligence. Modeling inference in human language is very challenging. With the availability of large annotated data (Bowman et al., 2015), it has recently become feasible to train neural network based inference models, which have shown to be very effective. In this paper, we present a new state-of-the-art result, achieving the accuracy of 88.6% on the Stanford Natural Language Inference Dataset. Unlike the previous top models that use very complicated network architectures, we first demonstrate that carefully designing sequential inference models based on chain LSTMs can outperform all previous models. Based on this, we further show that by explicitly considering recursive architectures in both local inference modeling and inference composition, we achieve additional improvement. Particularly, incorporating syntactic parsing information contributes to our best result—it further improves the performance even when added to the already very strong model.

condition for true natural language understanding is a mastery of open-domain natural language inference." The previous work has included extensive research on recognizing textual entailment.

Specifically, natural language inference (NLI) is concerned with determining whether a natural-language hypothesis h can be inferred from a premise p, as depicted in the following example from MacCartney (2009), where the hypothesis is regarded to be entailed from the premise.

p: *Several airlines polled saw costs grow more than expected, even after adjusting for inflation.*

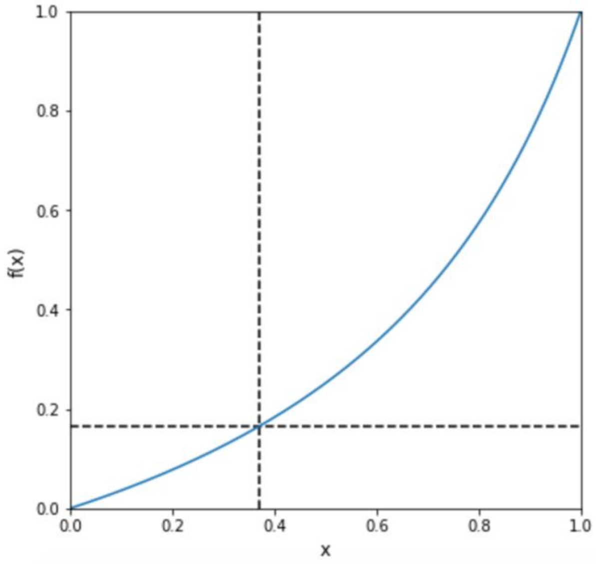
h: *Some of the companies in the poll reported cost increases.*

The most recent years have seen advances in modeling natural language inference. An important contribution is the creation of a much larger annotated dataset, the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015). The corpus has 570,000 human-written English sentence pairs manually labeled by multiple human subjects. This makes it feasible to train more complex inference models. Neural network models, which often need relatively large annotated data to estimate their parameters, have shown to achieve

- Cross entropy and training-test class imbalance

let a = 0.165 / 0.37, b = (1 - 0.165) / (1 - 0.37)

function to convert is f(x) = a * x / (a * x + b * (1 - x))



https://swarbrickjones.wordpress.com/2017/03/28/ … cla
ss-imbalance/