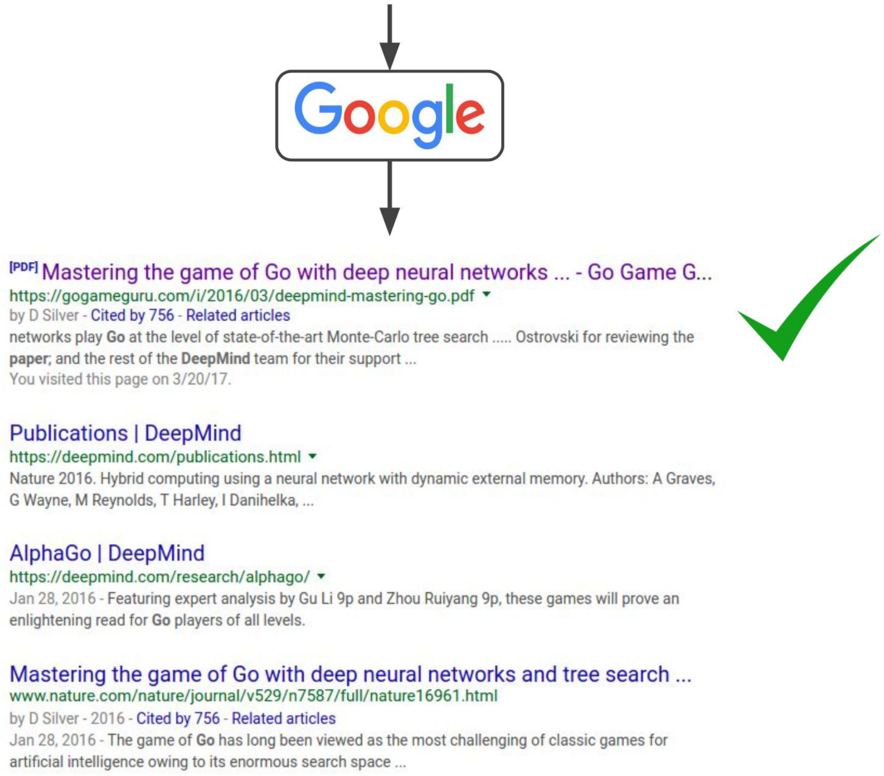
Task-Oriented Query Reformulation with Reinforcement Learning

Presenters:

Sreeja R Thoom
Jayavardhan Reddy Peddamail

Motivation

Query: "deepmind go paper"



Motivation

Query: "google artificial intelligence paper asian board game"

Google

[Master of Go Board Game Is Walloped by Google Computer Program ...](https://www.nytimes.com/2016/03/10/world/asia/google-alphago-lee-se-dol.html)
https://www.nytimes.com/2016/03/10/world/asia/google-alphago-lee-se-dol.html
Mar 9, 2016 - Lee Se-dol, the world's top player of the **boardgame** Go, lost the first of five matches to a computer ... Kim Sung-ryong, a South **Korean** Go master who provided commentary during ... wondered Rodney Brooks, a pioneering **artificial intelligence** researcher. ... Order Reprints| Today's **Paper**|Subscribe.

Google AI beats legendary player in Chinese board game - The Hindu
www.thehindu.com › Sci-Tech › Science
Mar 9, 2016 - South Korea's professional Go player Lee Sedol, right, playing the **game** with against **Google's artificial intelligence** program, AlphaGo. ... In a new feat, **Google**-run **artificial intelligence** (**AI**) programme "AlphaGo" has defeated legendary player Lee Se-dol in Go — a complex ...

Google AI algorithm masters ancient game of Go : Nature News ...
www.nature.com/news/google-ai-algorithm-masters-ancient-game-of-go-1.19234
Jan 27, 2016 - **Google AI** algorithm masters ancient game of Go ... A computer has beaten a human professional for the first time at Go — an ancient **board game** that ... in **Asia**, has frustrated the efforts of **artificial-intelligence** researchers for ...

Game over? Computer beats human champ in ancient Chinese game
https://phys.org › Technology › Computer Sciences
Jan 27, 2016 - In a milestone for **artificial intelligence**, a computer has beaten a ... human player at the 3,000-year-old **Chinese board game** known as Go, was ... of **Google** DeepMind, a British **artificial intelligence** (**AI**) company. ... added his colleague David Silver, who co-authored the **paper** in the science journal Nature.

?

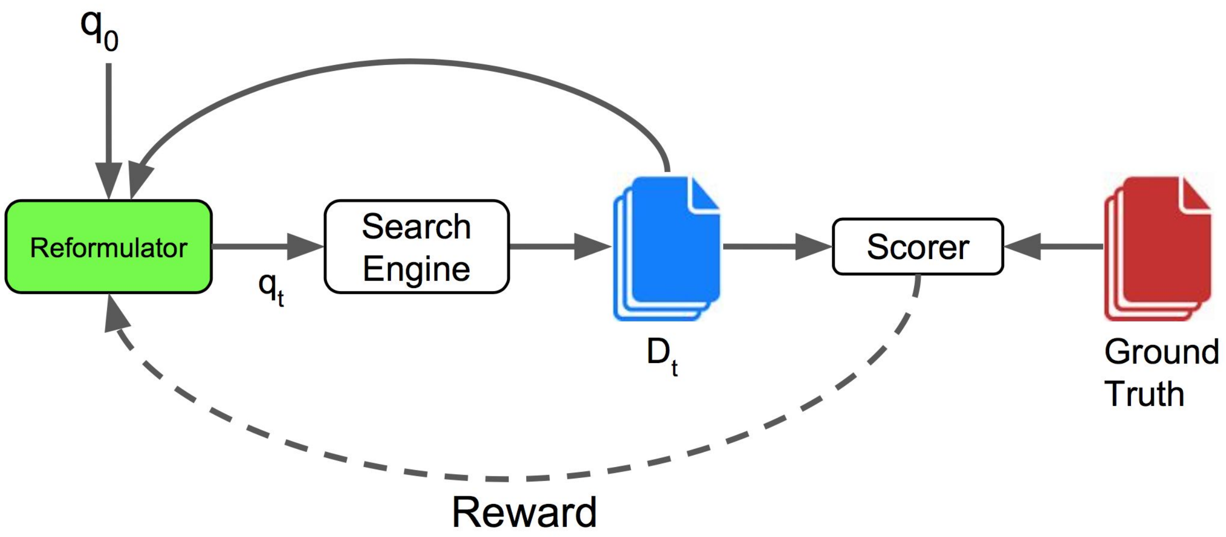
Motivation

The Vocabulary Mismatch Problem

The Idea

Query: "google artificial intelligence paper asian board game"
Google
Master of Go Board Game Is Walloped by Google Computer Program ...
https://www.nytimes.com/2016/03/10/world/asia/google-alphago-lee-se-dol.html
Mar 9, 2016 - Lee Se-dol, the world's top player of the **boardgame** Go, lost the first of five matches to a computer ... Kim Sung-ryong, a South **Korean** Go master who provided commentary during ... wondered Rodney Brooks, a pioneering **artificial intelligence** researcher. ... Order Reprints| Today's **Paper**|Subscribe.
Google AI beats legendary player in Chinese board game - The Hindu
www.thehindu.com › Sci-Tech › Science
Mar 9, 2016 - South Korea's professional Go player Lee Sedol, right, playing the **game** with against **Google's artificial intelligence** program, AlphaGo. ... In a new feat, **Google**-run **artificial intelligence** (**AI**) programme "AlphaGo" has defeated legendary player Lee Se-dol in Go — a complex ...
Google AI algorithm masters ancient game of Go : Nature News ...
www.nature.com/news/google-ai-algorithm-masters-ancient-game-of-go-1.19234
Jan 27, 2016 - **Google AI** algorithm masters ancient game of Go ... A computer has beaten a human professional for the first time at Go — an ancient **board game** that ... in **Asia**, has frustrated the efforts of **artificial-intelligence** researchers for ...
Game over? Computer beats human champ in ancient Chinese game
https://phys.org › Technology › Computer Sciences
Jan 27, 2016 - In a milestone for **artificial intelligence**, a computer has beaten a ... human player at the 3,000-year-old **Chinese board game** known as Go, was ... of **Google** DeepMind, a British **artificial intelligence** (**AI**) company. ... added his colleague David Silver, who co-authored the **paper** in the science journal Nature.
Reformulation
Query: "deepmind go paper"
Google
[PDF] Mastering the game of Go with deep neural networks ... - Go Game G...
https://gogameguru.com/i/2016/03/deepmind-mastering-go.pdf
by D Silver - Cited by 756 - Related articles
networks play **Go** at the level of state-of-the-art Monte-Carlo tree search ..... Ostrovski for reviewing the **paper**; and the rest of the **DeepMind** team for their support ...
You visited this page on 3/20/17.
Publications | DeepMind
https://deepmind.com/publications.html
Nature 2016. Hybrid computing using a neural network with dynamic external memory. Authors: A Graves, G Wayne, M Reynolds, T Harley, I Danihelka, ...
AlphaGo | DeepMind
https://deepmind.com/research/alphago/
Jan 28, 2016 - Featuring expert analysis by Gu Li 9p and Zhou Ruiyang 9p, these games will prove an enlightening read for **Go** players of all levels.

The Idea



q_0
Reformulator
q_t
Search Engine
D_t
Scorer
Ground Truth
Reward

Recall

$$\text{R@}K = \frac{|D_K \cap D^*|}{|D^*|}$$

Precision

$$\text{P@}K = \frac{|D_K \cap D^*|}{|D_K|}$$

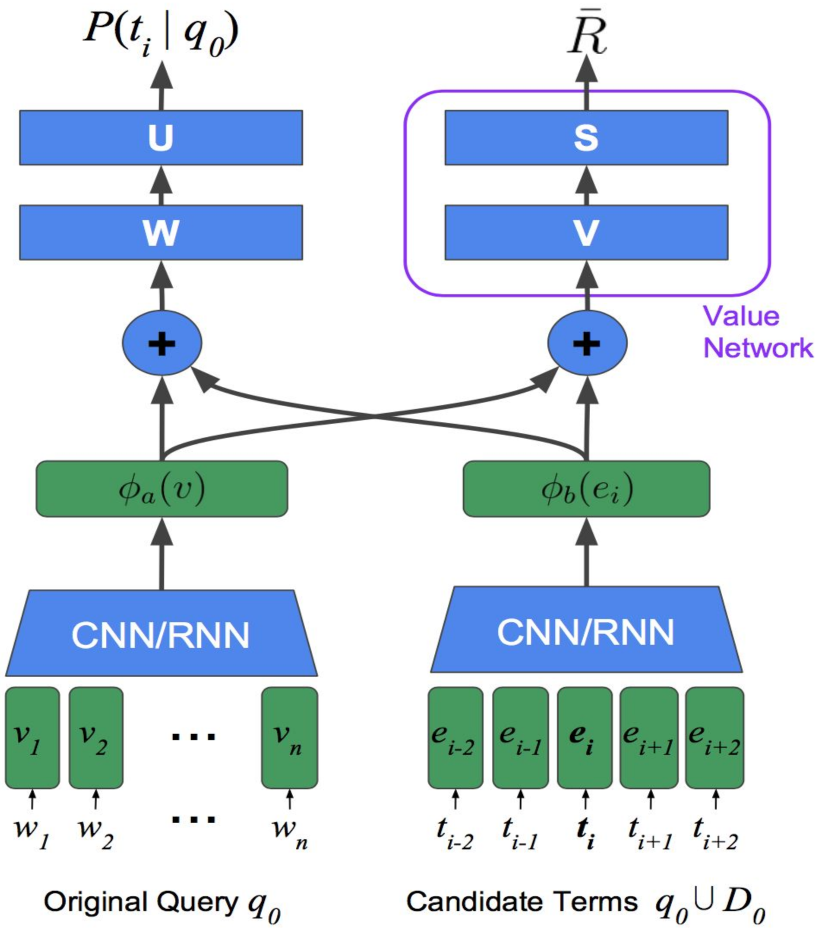
Mean Average Precision

$$\text{AP@}K = \frac{\sum_{k=1}^{K} \text{P@}k \times \text{rel}(k)}{|D^*|}$$

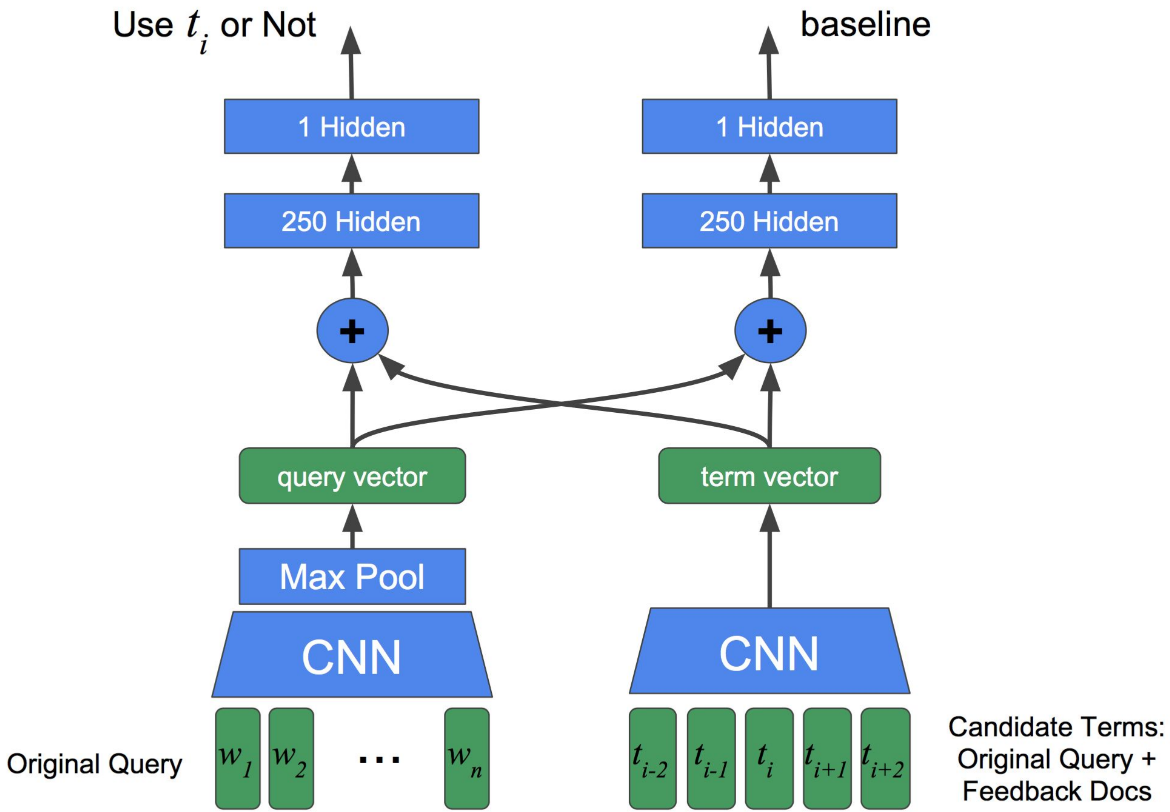
$$\text{rel}(k) = \begin{cases} 1, & \text{if the k-th document is relevant} \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{MAP@}K = \frac{1}{|Q|} \sum_{q \in Q} \text{AP@}K_q$$

Reformulator Network:



Reformulator



Probability of adding Candidate term to Query:

$$P(t_i|q_0) = \sigma(U^\intercal \tanh(W(\phi_a(v) \| \phi_b(e_i)) + b)), \tag{1}$$

where σ is the sigmoid function, $\|$ is the vector concatenation operation, $W \in \mathbb{R}^{d \times 2d}$ and $U \in \mathbb{R}^d$ are weights, and $b \in \mathbb{R}$ is a bias.

At test time, we define the set of terms used in the reformulated query as $T = \{t_i \mid P(t_i|q_0) > \epsilon\}$, where ϵ is a hyper-parameter. At training time, we sample the terms according to their probability distribution, $T = \{t_i \mid \alpha = 1 \wedge \alpha \sim P(t_i|q_0)\}$. We concatenate the terms in T to form a reformulated query q', which will then be used to retrieve a new set of documents D'.

RNN Sequence Generator

We define the probability of selecting t_i as the k-th term of a reformulated query as:

$$P(t_i^k|q_0) \propto \exp(\phi_b(e_i)^\intercal h_k), \tag{2}$$

where h_k is the hidden state vector at the k-th step, computed as:

$$h_k = \tanh(W_a \phi_a(v) + W_b \phi_b(t^{k-1}) + W_h h_{k-1}), \tag{3}$$

Training:

$$C_a = (R - \bar{R}) \sum_{t \in T} - \log P(t|q_0), \qquad (4)$$

where R is the reward and $\bar{R}$ is the baseline, computed by the value network as:

$$\bar{R} = \sigma(S^\mathsf{T} \tanh(V(\phi_a(v) \| \bar{e}) + b)), \qquad (5)$$

where $\bar{e} = \frac{1}{N} \sum_{i=1}^{N} \phi_b(e_i)$, $N = |q_0 \cup D_0|$, $V \in \mathbb{R}^{d \times 2d}$ and $S \in \mathbb{R}^d$ are weights and $b \in \mathbb{R}$ is a bias. We train the value network to minimize

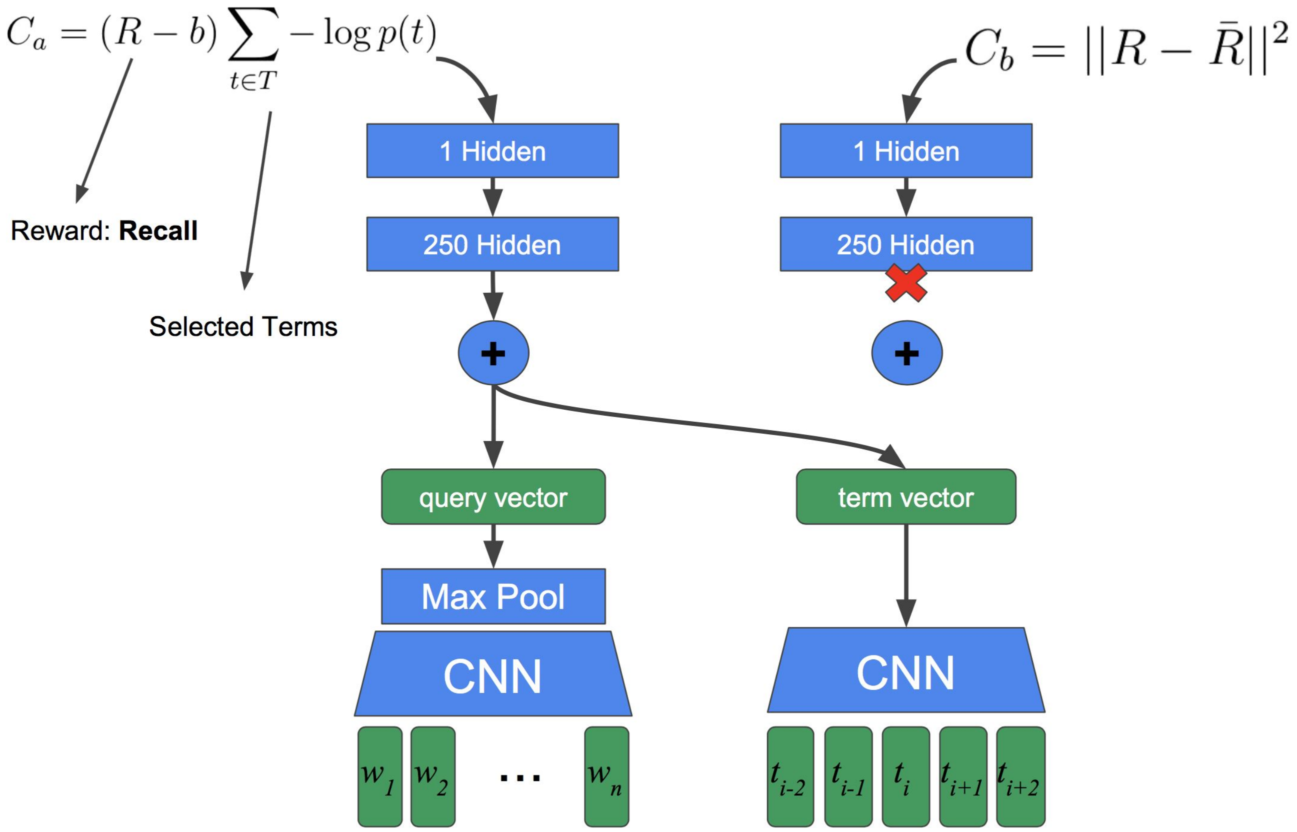
$$C_b = \alpha ||R - \bar{R}||^2, \qquad (6)$$

Entropy Regularization

$$C_H = -\lambda \sum_{t \in q_0 \cup D_0} P(t|q_0) \log P(t|q_0), \qquad (7)$$

where λ is a regularization coefficient.

Vanilla REINFORCE



Baseline Models:

1. Raw:
   Lucene, Google Search

2. PRF-TFIDF:
   The top-N TF-IDF terms from each of the top-K retrieved documents are added to the original query

3. PRF-RM:

$$P(t|q_0) = (1-\lambda)P'(t|q_0) + \lambda \sum_{d \in D_0} P(d)P(t|d)P(q_0|d)$$

$$P(t|d) = \frac{\mathrm{tf}(t,d) + uP(t|C)}{|d| + u}$$

4. Embedding Similarity:
   The top-N terms are selected based on the cosine similarity of their embeddings against the original query embedding.

Reinforcement Learning:

Variants: RL-CNN, RL-RNN, RL-FF, RL-RNN_SEQ

RL- ORACLE:

A conservative upper-bound performance of a RL model in a particular environment:

1- Split the **validation or test** data into smaller subsets (~2000 samples)

2- Overfit the RL model on each subset

3- Oracle performance = Average reward over all subsets

Supervised Learning(SL-CNN and SL-FF):

Step 1: Label each term as positive or negative based on its incremental reward.

Ex:

Query	Recall	Diff	Label
Google asian board game paper	0.40	-	-
Google asian board game paper **Deepmind**	0.45	0.05	**Pos**
Google asian board game paper **go**	0.43	0.03	**Pos**
Google asian board game paper **beats**	0.32	-0.08	**Neg**
Google asian board game paper **legendary**	0.35	-0.05	**Neg**

Step 2: Train a supervised classifier to predict if a term is positive or not.

Oracle: A classifier that perfectly selects relevant terms.

Datasets

- TREC-Complex Answer Retrieval
- Jeopardy
- Microsoft Academic

TREC-Complex Answer Retrieval Dataset

Input: Wikipedia Title + Section

"Sea Turtle Diet"

Output: Wikipedia Paragraphs under the Section

Sea turtle

Diet [edit]

The loggerhead, Kemp's ridley, olive ridley, hawksbill, flatback, and leatherback sea turtles are omnivorous for their entire life. Omnivorous turtles may eat a wide variety of plant and animal life including, decapods, seagrasses, seaweed, sponges, mollusks, cnidarians, echinoderms, worms and fish.[36][37][38][39] However some species specialize on certain prey.

The diet of green turtles changes with age.[40] Juveniles are omnivorous, but as they mature they become exclusively herbivorous.[37][40] This diet shift has an effect on the green turtle's morphology.[41][42] Green sea turtles have a serrated jaw that is used to eat sea grass and algae.[43]

Leatherback turtles feed almost exclusively on jellyfish and help control jellyfish populations.[44][45]

Hawksbills principally eat sponges, which constitute 70–95% of their diets in the Caribbean.[46]

Corpus: Paragraphs of all Wikipedia Articles

http://trec-car.cs.unh.edu/

Jeopardy Dataset

Input: Jeopardy question

"For the last 8 years of his life, Galileo was under house arrest for espousing this man's theory."

Output: Wikipedia article whose title is the Answer.

Nicolaus Copernicus

From Wikipedia, the free encyclopedia

"Copernicus" redirects here. For other uses, see Copernicus (disambiguation).

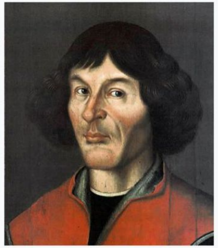
Nicolaus Copernicus (/koʊˈpɜːrnɪkəs, kə-/;[1][2][3] Polish: *Mikołaj Kopernik* [miˈkɔwaj kɔˈpɛrɲik] (listen); German: *Nikolaus Kopernikus*; 19 February 1473 – 24 May 1543) was a Renaissance mathematician and astronomer who formulated a model of the universe that placed the Sun rather than the Earth at the center of the universe, likely independently of Aristarchus of Samos, who had formulated such a model some eighteen centuries earlier.[a]

The publication of Copernicus' model in his book *De revolutionibus orbium coelestium* (*On the Revolutions of the Celestial Spheres*), just before his death in 1543, was a major event in the history of science, triggering the Copernican Revolution and making an important contribution to the Scientific Revolution.[7]

Copernicus was born and died in Royal Prussia, a region that had been part of the Kingdom of Poland since 1466. A polyglot and polymath, he obtained a doctorate in canon law and was also a mathematician, astronomer, physician, classics scholar, translator, governor, diplomat, and economist. In 1517 he derived a quantity theory of money – a key concept in economics – and in 1519 he formulated an economics principle that later came to be called Gresham's law.[8]

Contents [hide]

Nicolaus Copernicus



1580 portrait (artist unknown) in the Old Town City Hall, Toruń

(Nogueira and Cho, NIPS 2016)

Microsoft Academic Dataset

Input: Title/Abstract of a Paper

Output: References in that paper

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

References

[1] R.M. Bell and Y. Koren. Lessons from the netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, 9(2):75–79, 2007.

[2] A. Berg, J. Deng, and L. Fei-Fei. Large scale visual recognition challenge 2010. www.image-net.org/challenges. 2010.

[3] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] D. Cireşan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. *Arxiv preprint arXiv:1202.2745*, 2012.

[5] D.C. Cireşan, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. High-performance neural networks for visual object classification. *Arxiv preprint arXiv:1102.0183*, 2011.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[7] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei. *ILSVRC-2012*, 2012. URL http://www.image-net.org/challenges/LSVRC/2012/.

[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. URL http://authors.library.caltech.edu/7694.

[10] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R.R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[11] K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *International Conference on Computer Vision*, pages 2146–2153. IEEE, 2009.

[12] A. Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto, 2009.

(An in-house crawl of 500k papers)

Datasets - Summary

	TREC-CAR	Jeopardy	MS Academic
Corpus	Wikipedia Paragraphs	Wikipedia Articles	Academic Papers
Corpus Size	5.9M	3.5M	500k
Train / Valid / Test	580k / 195k / 195k	120k / 10k / 10k	270k / 20k / 20k

Implementation Details

Search Engine

Candidate Terms

Reformulation rounds

Experiment Details

Choice	Why?
Search Engine: Lucene	Free and Fast!
Feedback docs: Top-5 Wikipedia Articles Feedback terms: 300 first words of a Wikipedia doc	Maximum fit on a GPU
CNN for Docs: 1st layer: 9-word window, 250 filters 2nd layer: 3-word window, 500 filters CNN for Query: 2 layers, 3-word window, 250 filters	Painful Manual Trial-and-Error
Reward: Recall@40	Query Reformulation: Recall Ranking Functions: Precision

Results

Method	TREC-CAR			Jeopardy			MSA		
	R@40	P@10	MAP@40	R@40	P@10	MAP@40	R@40	P@10	MAP@40
Raw-Lucene	43.6	7.24	19.6	23.4	1.47	7.40	12.9	7.24	3.36
Raw-Google	-	-	-	30.1	1.92	7.71	-	-	-
PRF-TFIDF	44.3	7.31	19.9	29.9	1.91	7.65	13.2	7.27	3.50
PRF-RM	45.1	7.35	19.5	30.5	1.96	7.64	12.3	7.22	3.38
PRF-Emb	44.5	7.32	19.0	30.1	1.92	7.74	12.2	7.22	3.20
Vocab-Emb	44.2	7.30	19.1	29.4	1.87	7.80	12.0	7.21	3.21
SL-FF	44.1	7.29	19.7	30.8	1.95	7.70	13.2	7.28	3.88
SL-CNN	45.3	7.35	19.8	31.1	1.98	7.79	14.0	7.42	3.99
SL-Oracle	50.8	8.25	21.0	38.8	2.50	9.92	17.3	10.12	4.89
RL-FF	44.1	7.29	20.0	31.0	1.98	7.81	13.9	7.33	3.81
RL-CNN	47.3	7.45	20.3	33.4	**2.14**	8.02	14.9	7.63	4.30
RL-RNN	**47.9**	**7.52**	**20.6**	**33.7**	2.12	**8.07**	**15.1**	**7.68**	**4.35**
RL-RNN-SEQ	47.4	7.48	20.3	33.4	2.13	8.01	14.8	7.63	4.27
RL-Oracle	55.9	9.06	23.0	42.4	2.74	10.3	24.6	12.83	6.33

Table 2: Results on Test sets. We use R@40 as a reward to the RL-based models.

Examples

Original: *It can be a herdsman's little house in the Swiss Alps, or a ski lodge built in that style*

Reformulated: *house Swiss Alps ski lodge that style castle board chalet*

Original: Homelessness in Canada, Public Policy

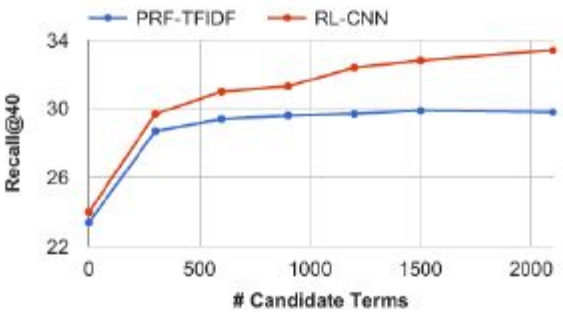
Reformulated: *homelessness in canada public policy* human service programs social policy california treatment of the homeless numerous

Observations

1. Framework performs better on the first dataset compared to other two datasets

2. RL-RNN-SEQ is faster in retrieving documents

3. With more computational resources framework performance improves.



Critic:

1. Only Term addition discussed. Term deletion can also improve retrieval performance.
2. Implemented using REINFORCE algorithm. Q-Learning and Deep Q-Learning can be implemented for the same model architecture.
3. Assumption, words in initial retrieved documents are right. If the initial results are wrong, the model might produce wrong results.
4. They mentioned they did not observe any improvement after 1 iteration. It might be because of the previous assumption
5. The reformulated query could have used Boolean operators.