SAND2015-10252C

Enhancement of Enterprise Search With Neural Network Language Model

Pengchu Zhang, Laritza Saenz and John Mareda
Sandia National Laboratories
November 30, 2015

Problems in Enterprise Search

- Search engines are basically matching the query terms and the terms in documents, this may result in a large number of false positives;
- Some terms may be overly represented in search:
  - E.g., "composite materials"
- Search engines may fail to deliver if the query terms not exist in the enterprise datasets but the datasets have the relevant information, e.g.,
  - Buckyballs vs. fullerene

Current Approaches to Enhance Information finding

- Boosting some terms;
  - E.g., Composite(+2) materials
- Synonym to expand query terms;
  - Java vs. coffee
- Others such as applying various similarity algorithms

In this presentation, we propose to improve information findability with Neural Network Language Models

NNLM in NATURE

www.nature.com/nature/journal/v521/n7553/pdf/nature14539.pdf

REVIEW

doi:10.1038/nature14539

Deep learning

Yann LeCun[1,2], Yoshua Bengio[3] & Geoffrey Hinton[4,5]

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning discovers intricate structure in large data sets by using the backpropagation algorithm to indicate how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Deep convolutional nets have brought about breakthroughs in processing images, video, speech and audio, whereas recurrent nets have shone light on sequential data such as text and speech.

Machine-learning technology powers many aspects of modern society: from web searches to content filtering on social networks to recommendations on e-commerce websites, and it is increasingly present in consumer products such as cameras and smartphones. Machine-learning systems are used to identify objects in images, transcribe speech into text, match news items, posts or products with users' interests, and select relevant results of search. Increasingly, these applications make use of a class of techniques called deep learning.

Conventional machine-learning techniques were limited in their ability to process natural data in their raw form. For decades, constructing a pattern-recognition or machine-learning system required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data (such as the pixel values of an image) into a suitable internal representation or feature vector from which the learning subsystem, often a classifier, could detect or classify patterns in the input.

Representation learning is a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification. Deep-learning methods are representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. With the composition of enough such transformations, very complex functions can be learned. For classification tasks, higher layers of representation amplify aspects of the input that are important for discrimination and suppress irrelevant variations. An image, for example, comes in the form of an array of pixel values, and the learned features in the first layer of representation typically represent the presence or absence of edges at particular orientations and locations in the image. The second layer typically detects motifs by spotting particular arrangements of

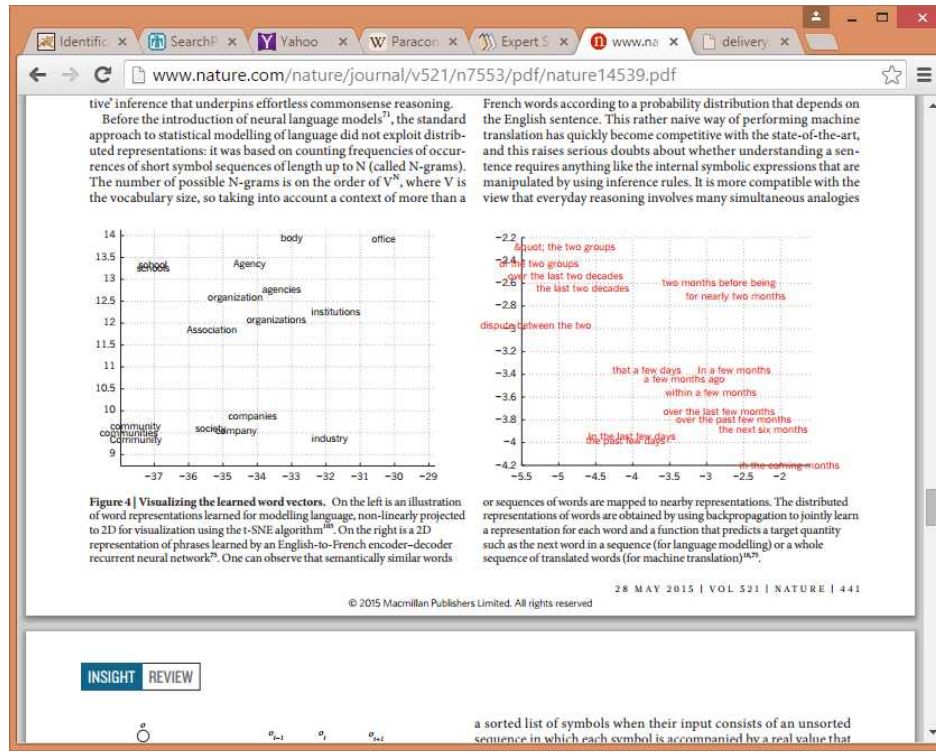
intricate structures in high-dimensional data and is therefore applicable to many domains of science, business and government. In addition to beating records in image recognition[1–4] and speech recognition[5–7], it has beaten other machine-learning techniques at predicting the activity of potential drug molecules[8], analysing particle accelerator data[9,10], reconstructing brain circuits[11], and predicting the effects of mutations in non-coding DNA on gene expression and disease[12,13]. Perhaps more surprisingly, deep learning has produced extremely promising results for various tasks in natural language understanding[14], particularly topic classification, sentiment analysis, question answering[15] and language translation[16,17].

We think that deep learning will have many more successes in the near future because it requires very little engineering by hand, so it can easily take advantage of increases in the amount of available computation and data. New learning algorithms and architectures that are currently being developed for deep neural networks will only accelerate this progress.

Supervised learning

The most common form of machine learning, deep or not, is supervised learning. Imagine that we want to build a system that can classify images as containing, say, a house, a car, a person or a pet. We first collect a large data set of images of houses, cars, people and pets, each labelled with its category. During training, the machine is shown an image and produces an output in the form of a vector of scores, one for each category. We want the desired category to have the highest score of all categories, but this is unlikely to happen before training. We compute an objective function that measures the error (or distance) between the output scores and the desired pattern of scores. The machine then modifies its internal adjustable parameters to reduce this error. These adjustable parameters, often called weights, are real numbers that can be seen as 'knobs' that define the input–output func-

From Nature 521, 4346-444 (28 May 2015)



www.nature.com/nature/journal/v521/n7553/pdf/nature14539.pdf

tive' inference that underpins effortless commonsense reasoning.

Before the introduction of neural language models[71], the standard approach to statistical modelling of language did not exploit distributed representations: it was based on counting frequencies of occurrences of short symbol sequences of length up to N (called N-grams). The number of possible N-grams is on the order of V^N, where V is the vocabulary size, so taking into account a context of more than a

French words according to a probability distribution that depends on the English sentence. This rather naive way of performing machine translation has quickly become competitive with the state-of-the-art, and this raises serious doubts about whether understanding a sentence requires anything like the internal symbolic expressions that are manipulated by using inference rules. It is more compatible with the view that everyday reasoning involves many simultaneous analogies

Figure 4 | Visualizing the learned word vectors. On the left is an illustration of word representations learned for modelling language, non-linearly projected to 2D for visualization using the t-SNE algorithm[103]. On the right is a 2D representation of phrases learned by an English-to-French encoder–decoder recurrent neural network[75]. One can observe that semantically similar words or sequences of words are mapped to nearby representations. The distributed representations of words are obtained by using backpropagation to jointly learn a representation for each word and a function that predicts a target quantity such as the next word in a sequence (for language modelling) or a whole sequence of translated words (for machine translation)[18,75].

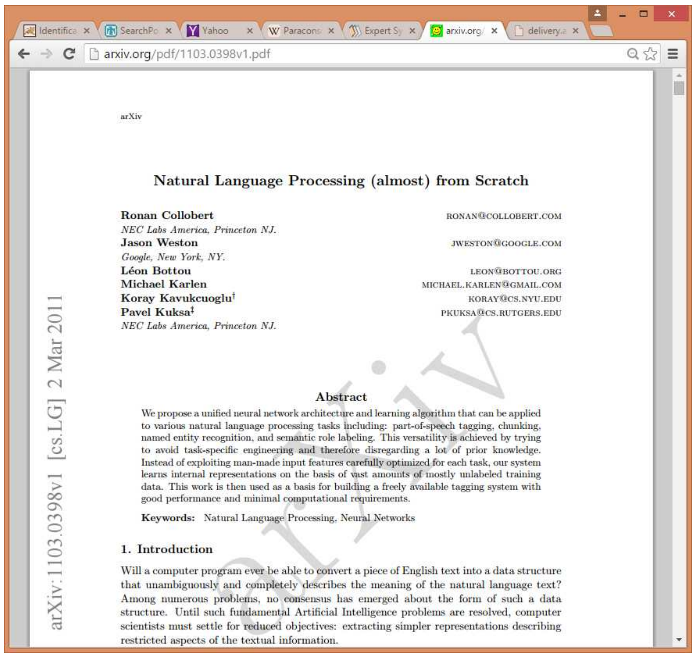
28 MAY 2015 | VOL 521 | NATURE | 441

© 2015 Macmillan Publishers Limited. All rights reserved

INSIGHT REVIEW

a sorted list of symbols when their input consists of an unsorted

From Theory to Application to Codes



arxiv.org/pdf/1103.0398v1.pdf

arXiv

Natural Language Processing (almost) from Scratch

Ronan Collobert RONAN@COLLOBERT.COM
NEC Labs America, Princeton NJ.
Jason Weston JWESTON@GOOGLE.COM
Google, New York, NY.
Léon Bottou LEON@BOTTOU.ORG
Michael Karlen MICHAEL.KARLEN@GMAIL.COM
Koray Kavukcuoglu[†] KORAY@CS.NYU.EDU
Pavel Kuksa[‡] PKUKSA@CS.RUTGERS.EDU
NEC Labs America, Princeton NJ.

arXiv:1103.0398v1 [cs.LG] 2 Mar 2011

Abstract

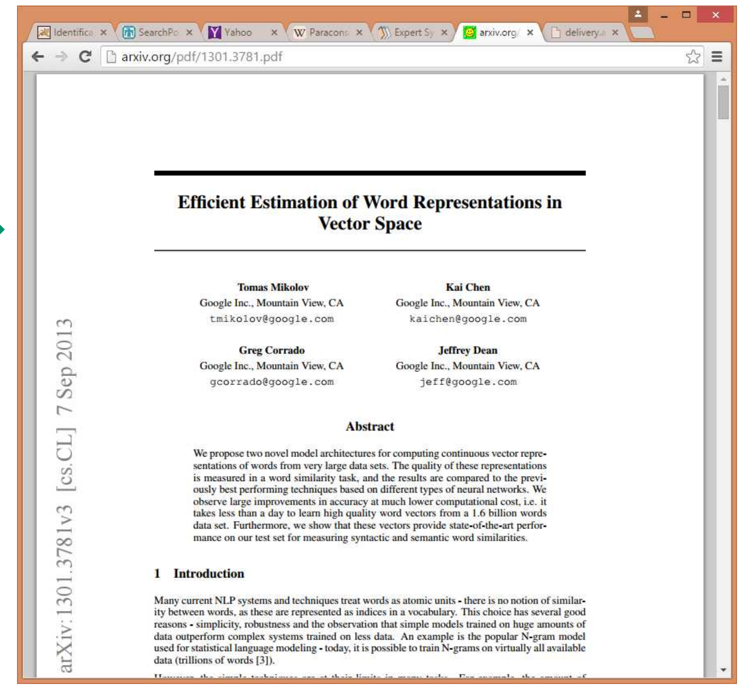
We propose a unified neural network architecture and learning algorithm that can be applied to various natural language processing tasks including: part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. This versatility is achieved by trying to avoid task-specific engineering and therefore disregarding a lot of prior knowledge. Instead of exploiting man-made input features carefully optimized for each task, our system learns internal representations on the basis of vast amounts of mostly unlabeled training data. This work is then used as a basis for building a freely available tagging system with good performance and minimal computational requirements.

Keywords: Natural Language Processing, Neural Networks

1. Introduction

Will a computer program ever be able to convert a piece of English text into a data structure that unambiguously and completely describes the meaning of the natural language text? Among numerous problems, no consensus has emerged about the form of such a data structure. Until such fundamental Artificial Intelligence problems are resolved, computer scientists must settle for reduced objectives: extracting simpler representations describing restricted aspects of the textual information.

arXiv:1301.3781v3



arxiv.org/pdf/1301.3781.pdf

Efficient Estimation of Word Representations in Vector Space

Tomas Mikolov
Google Inc., Mountain View, CA
tmikolov@google.com

Kai Chen
Google Inc., Mountain View, CA
kaichen@google.com

Greg Corrado
Google Inc., Mountain View, CA
gcorrado@google.com

Jeffrey Dean
Google Inc., Mountain View, CA
jeff@google.com

arXiv:1301.3781v3 [cs.CL] 7 Sep 2013

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.

1 Introduction

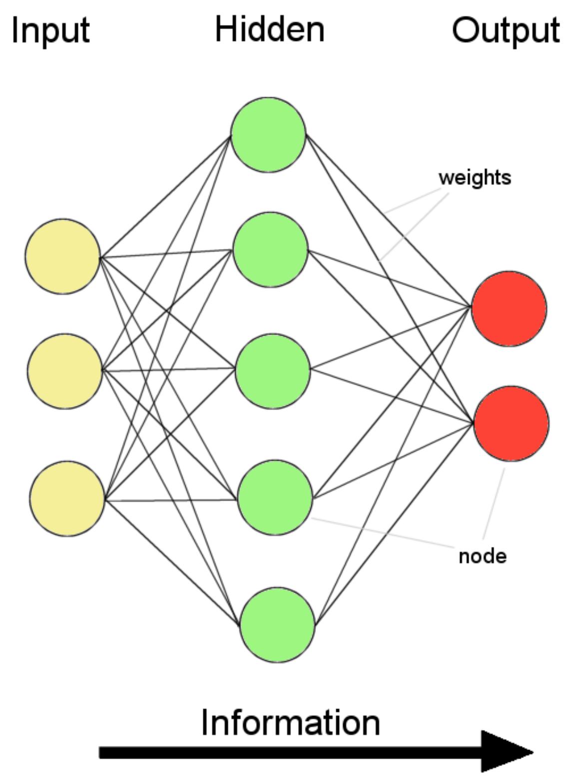
Many current NLP systems and techniques treat words as atomic units - there is no notion of similarity between words, as these are represented as indices in a vocabulary. This choice has several good reasons - simplicity, robustness and the observation that simple models trained on huge amounts of data 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]).

arXiv:1103.0398v1

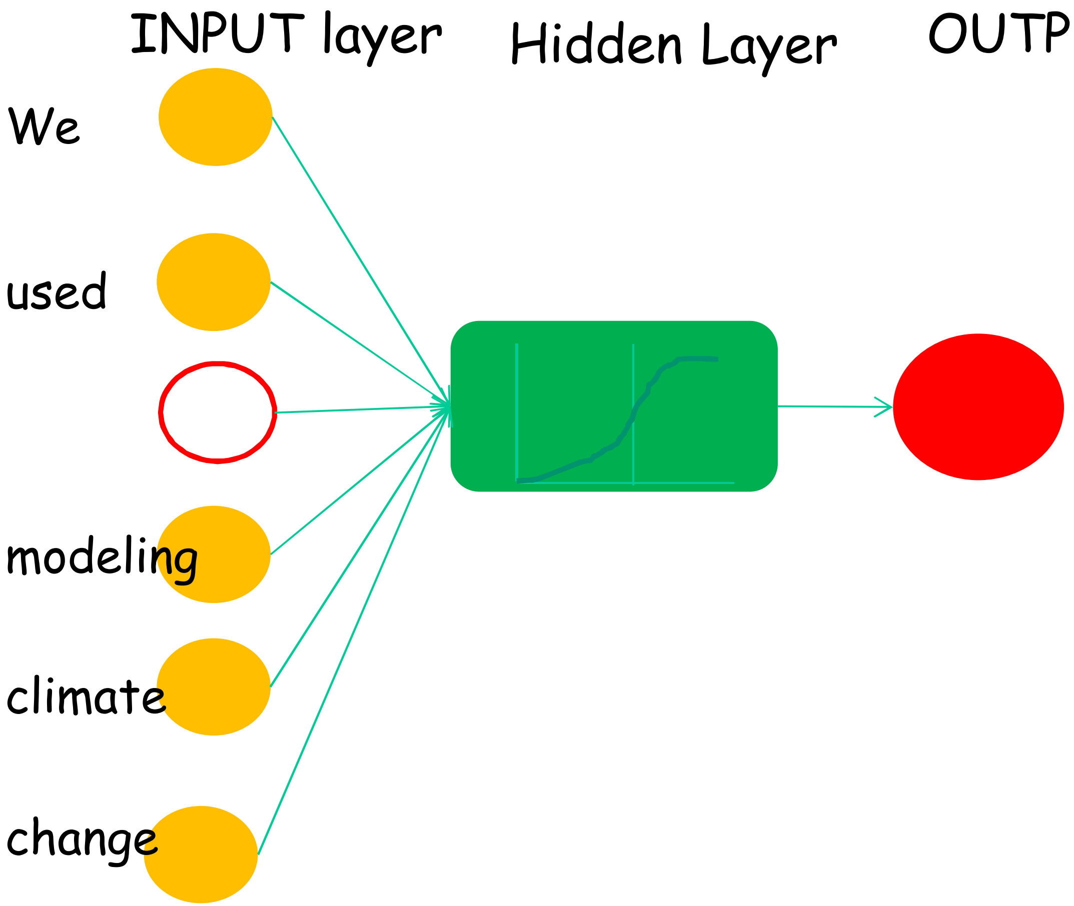
https://code.google.com/p/word2vec/

What is Neural Network Language Model?

Neural Network



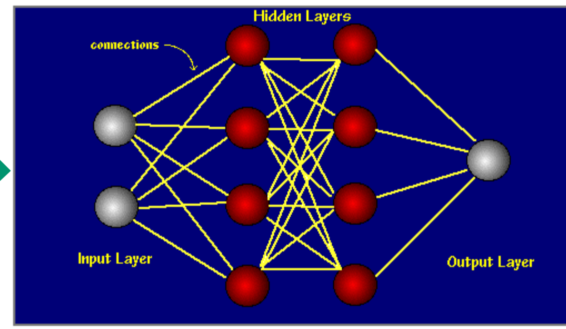
Concept of Neural Network Language Model

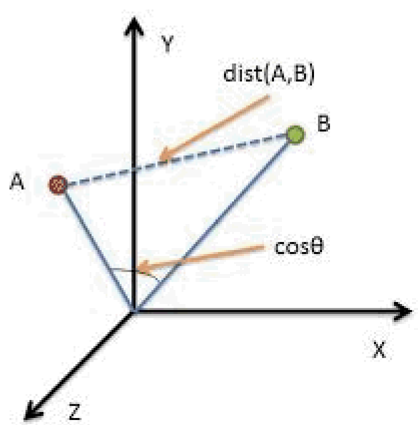


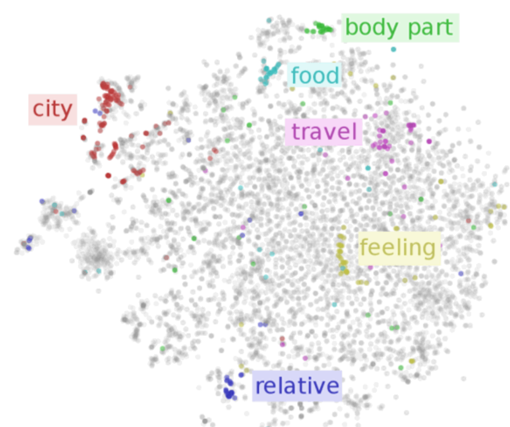
Term	P
Computer	0.9
Algorithm	0.8
Software	0.8
Brain	0.5
Food	0.2
Ghost	0.3
Dog	0.2

Build and optimize word vectors(Google 2013)

Word in sentence	vector		New vector
The	(0.12, 0.23, 0.22)		(0.12, 0.23, 0.22)
dog	(0.32, 0.27, 0.94)		(0.62, 0.99, 0.14)
is	(0.18, 0.88, 0.45)		(0.18, 0.88, 0.45)
walking	(0.23, 0.92, 0.23)		(0.23, 0.92, 0.23)
in	(0.77, 0.25, 0.11)		(0.77, 0.25, 0.11)
the	(0.12, 0.23, 0.22)		(0.12, 0.23, 0.22)
room	(0.41, 0.13, 0.29)		(0.41, 0.13, 0.29)





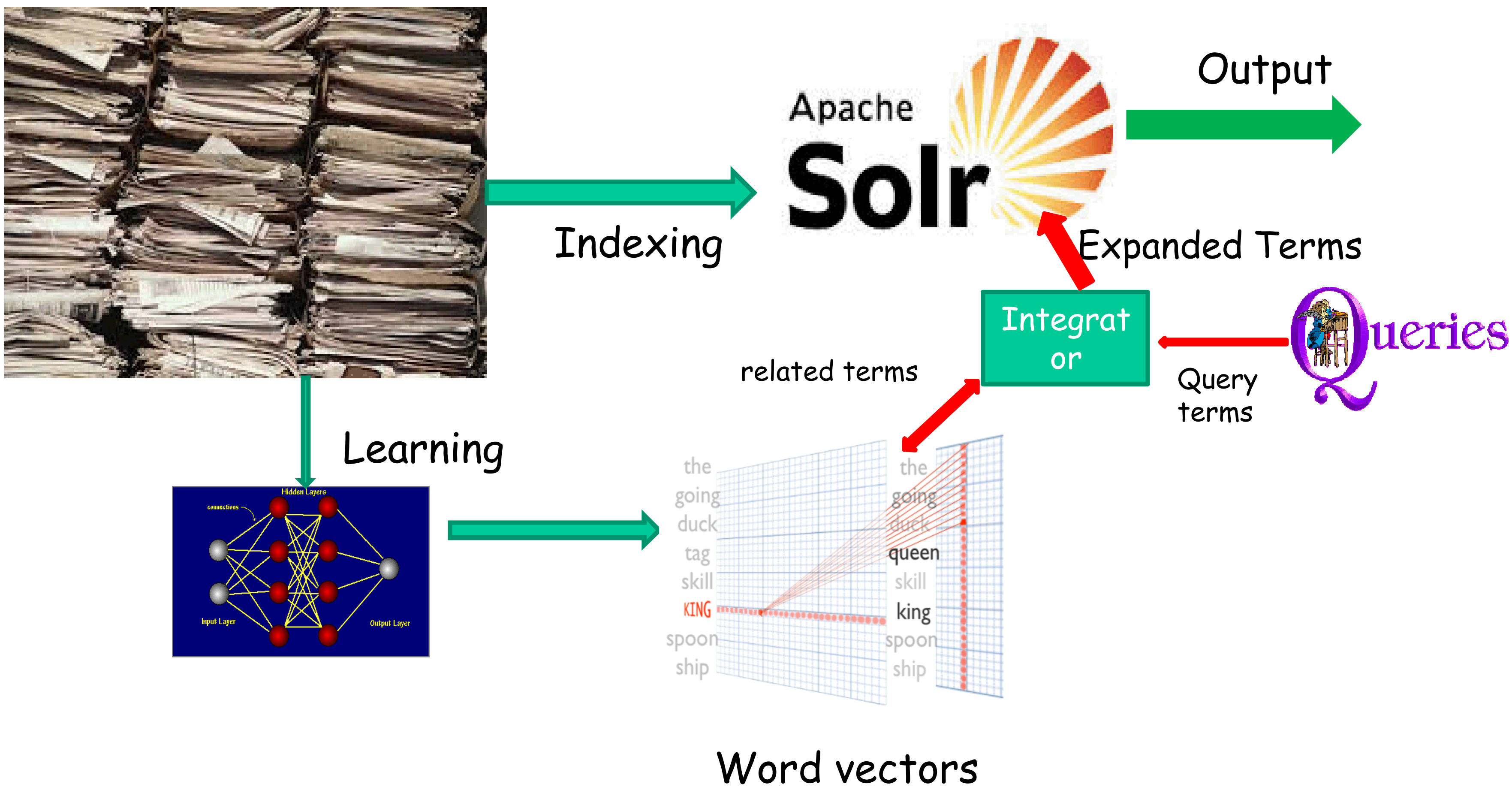


Sandia National Laboratories

Motivations in Applying Neural Network Language Model (NNLM)

- NNLM projects the terms into vector space, meaning the words that were used together will stay together at the vector space;
- NNLM vectorizes the terms that makes it possible to compute the distances between the terms;
- Most importantly, NNLM learns from natural language semantically and syntactically. With this NNLM meaningfully brings words together without parsing a speech or a document.

Concept for Enhanced Enterprise Search



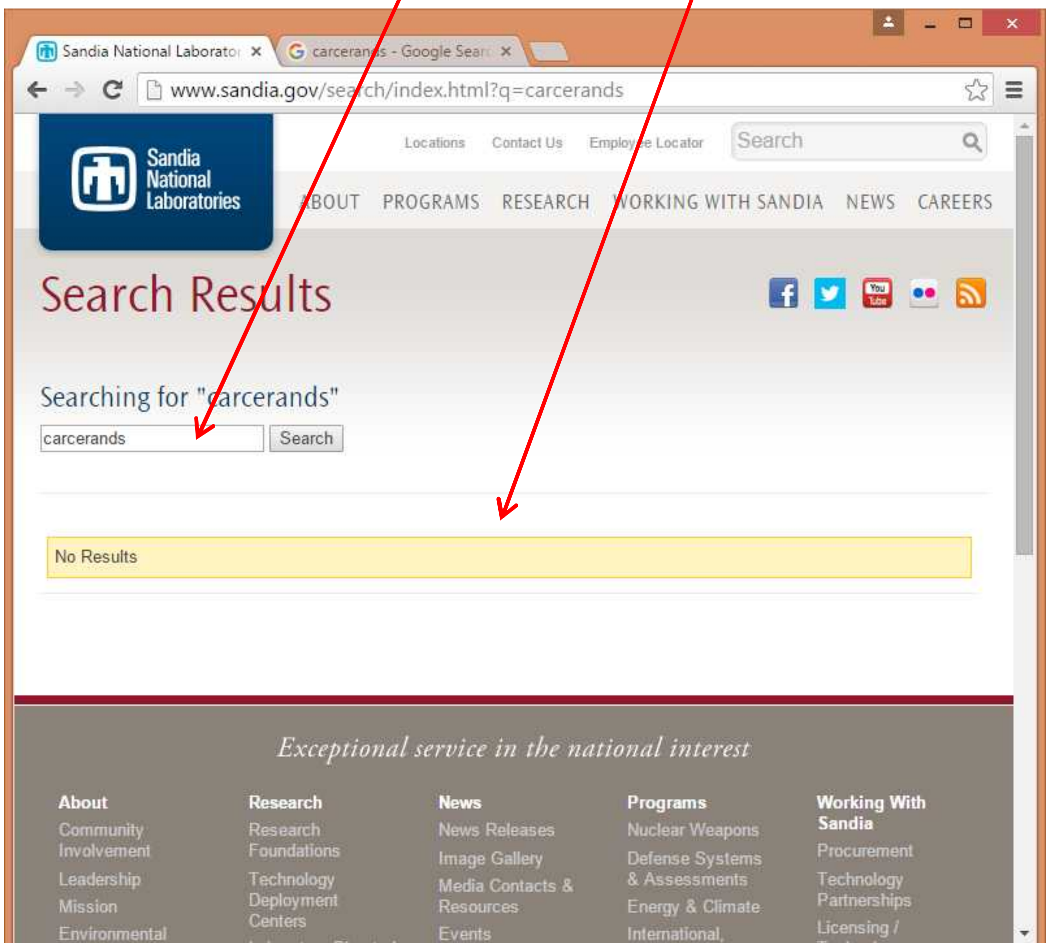
Note: All images are from online

What and How NNLM helps Enterprise Search

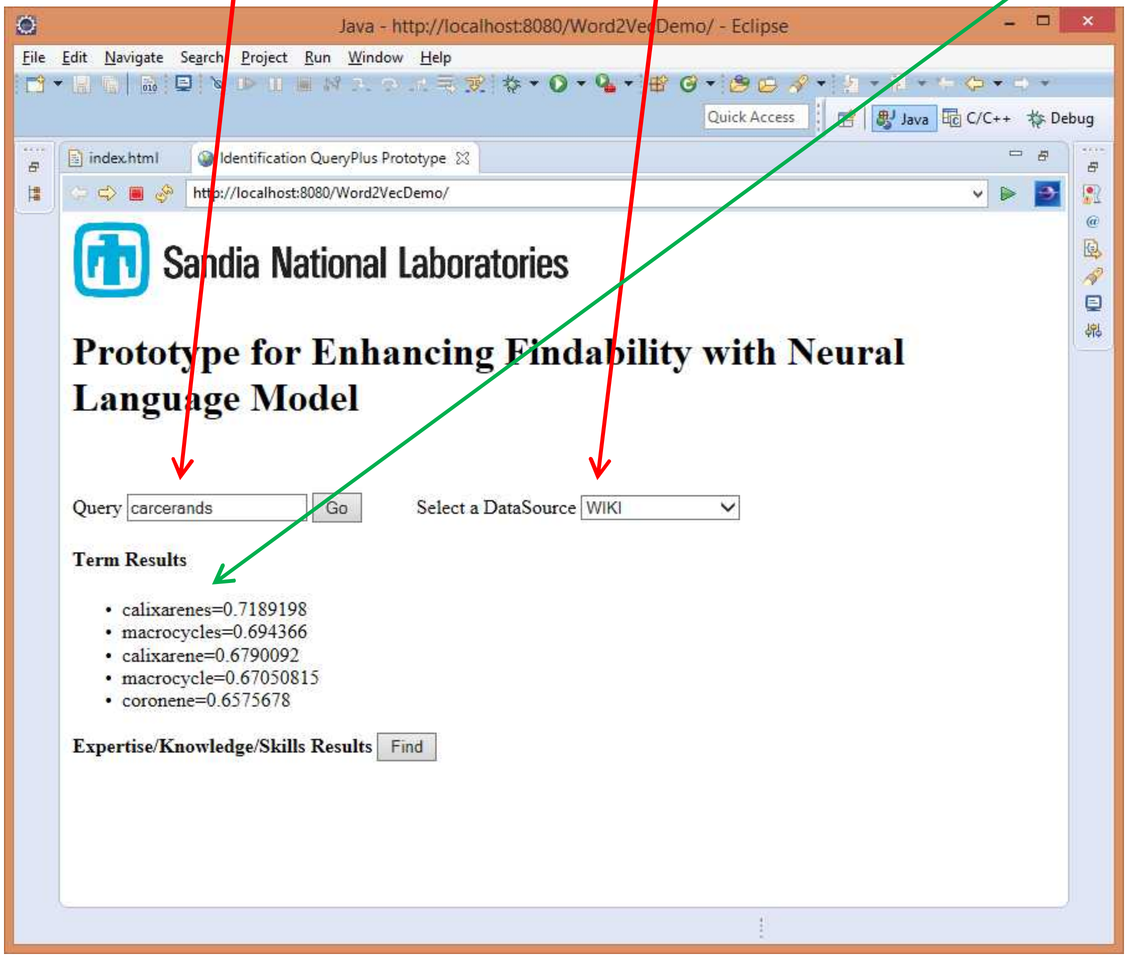
- Deliver the relevant information even the query terms not in enterprise dataset;
- Increase the quality of information delivered:
  - Relevance
  - Related

- Deliver the relevant information even the query terms not in enterprise dataset

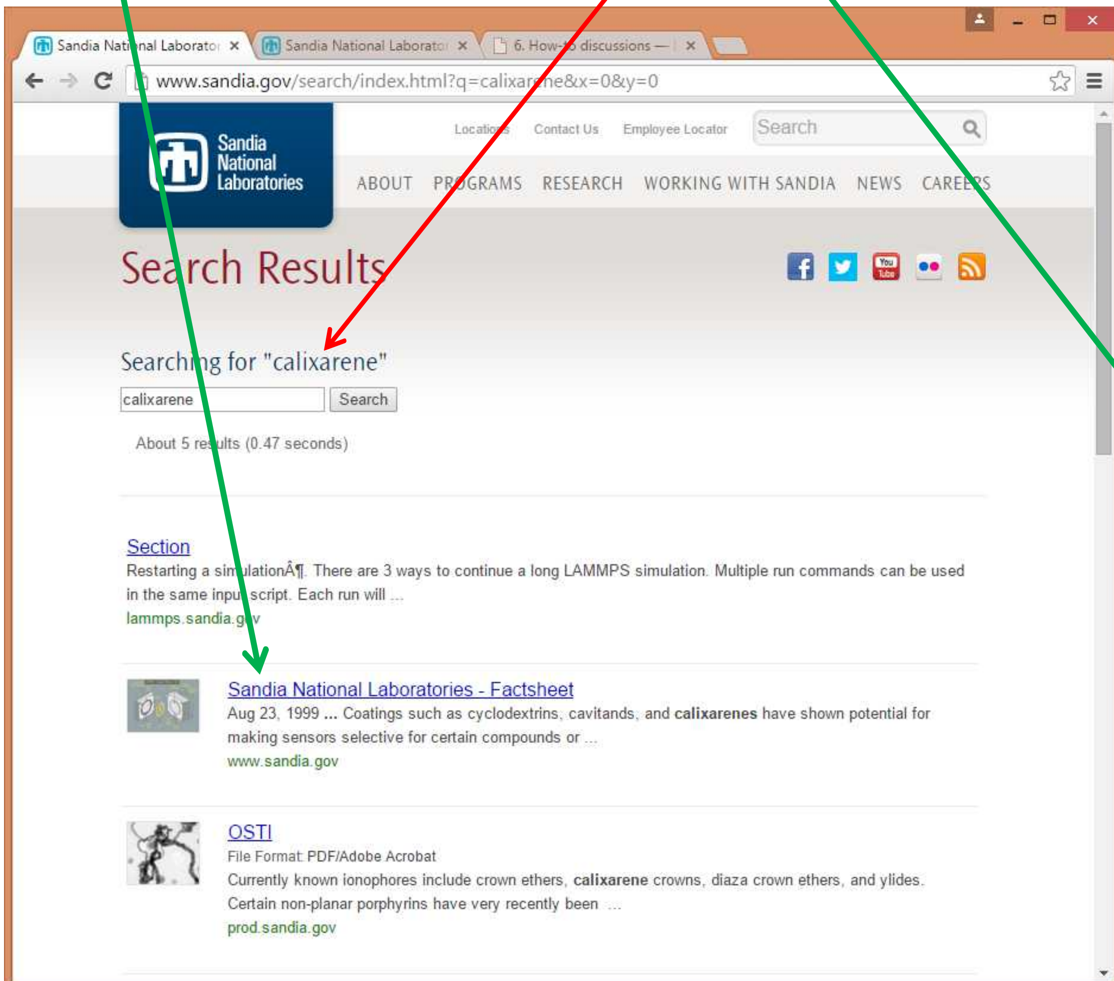
Query term "carcerands" not the in enterprise dataset

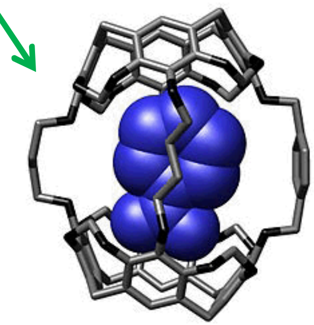


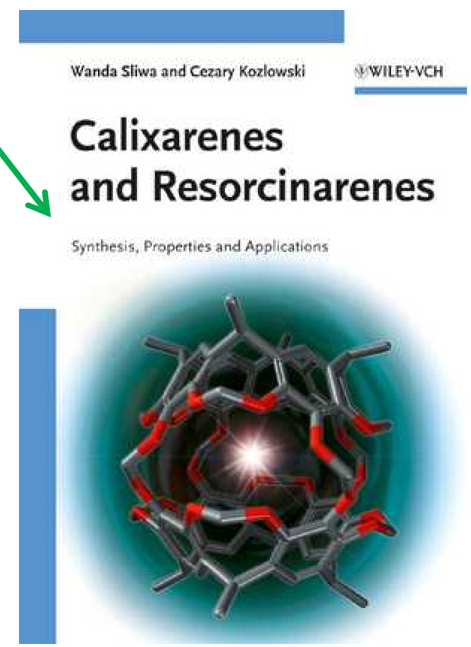
Term "carcerands" is found in Wikipedia English and a set of close terms are generated from NNLM



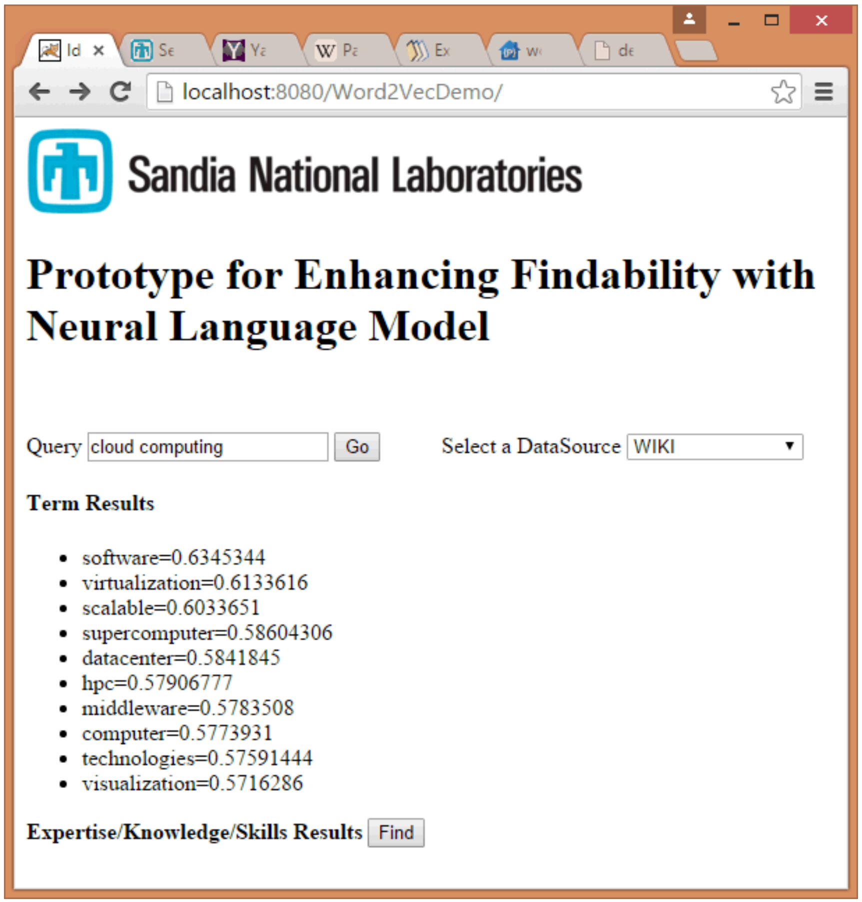
Sandia has a number of document associated with "carcerands" for which it is related to "calixarenes"

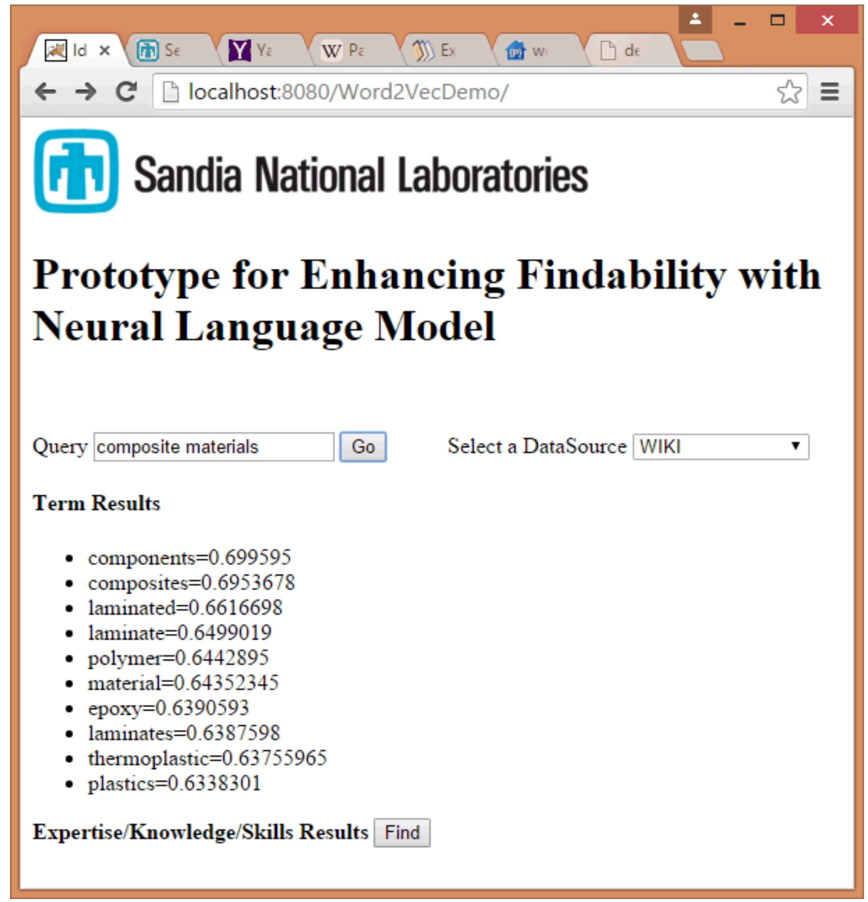




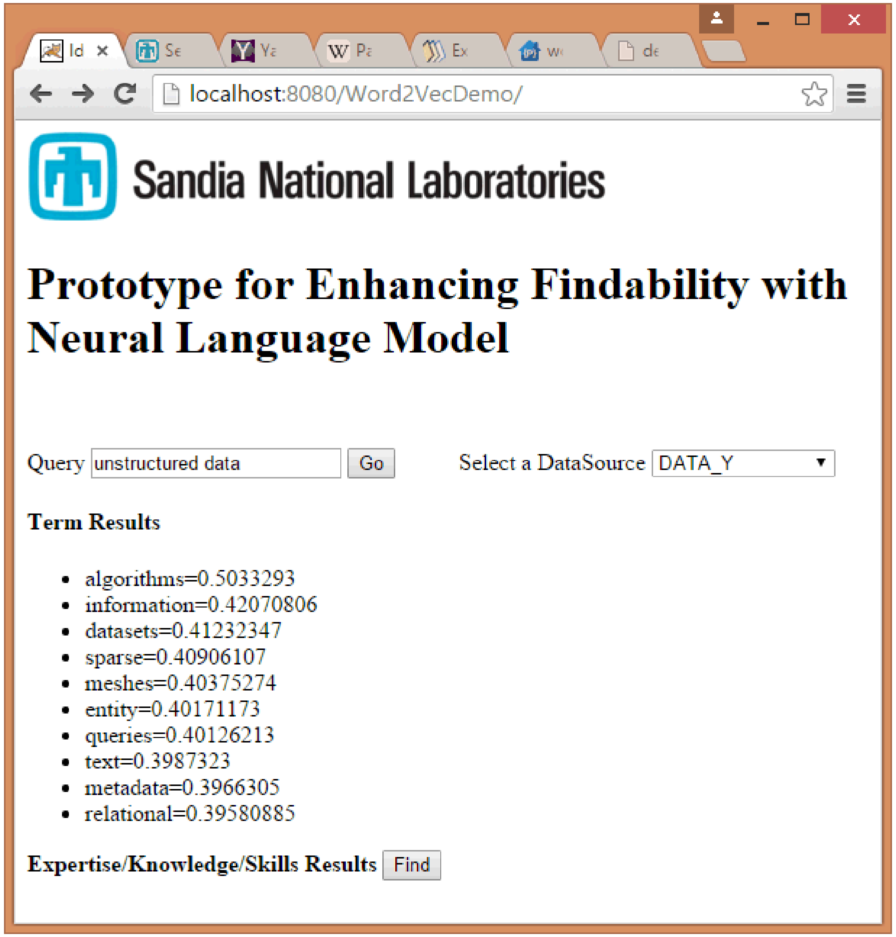


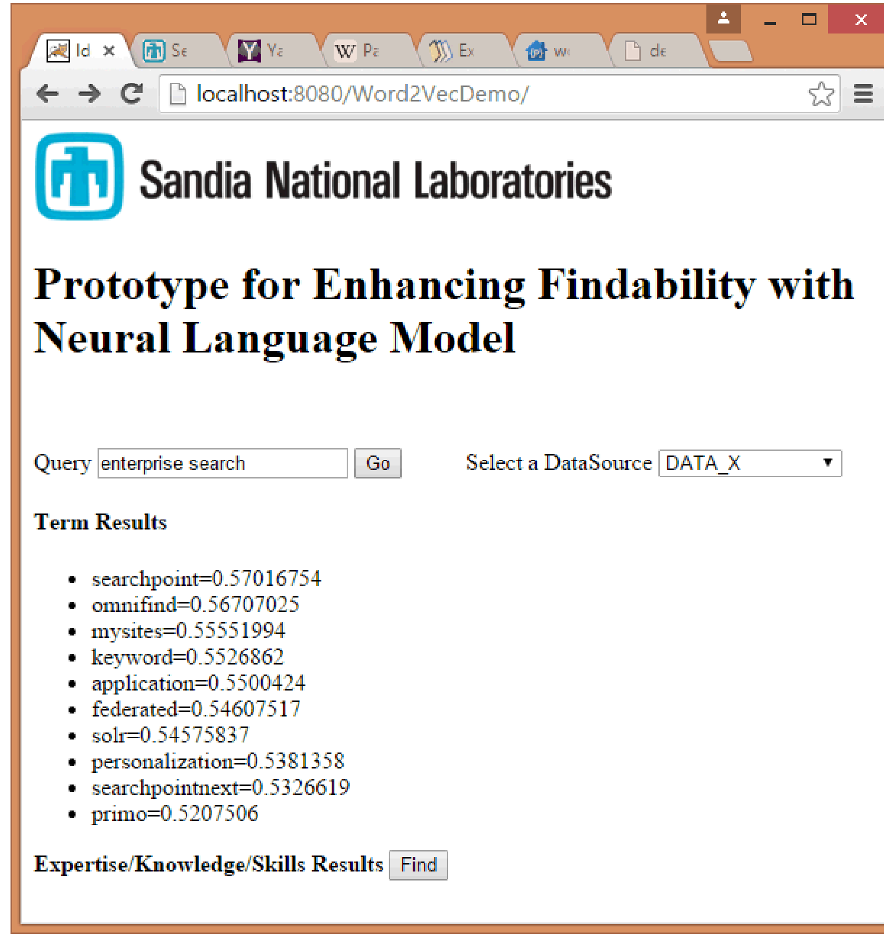
Increase the quality of retrieved information delivered: Relevant





Increase the quality of retrieved information delivered: Related

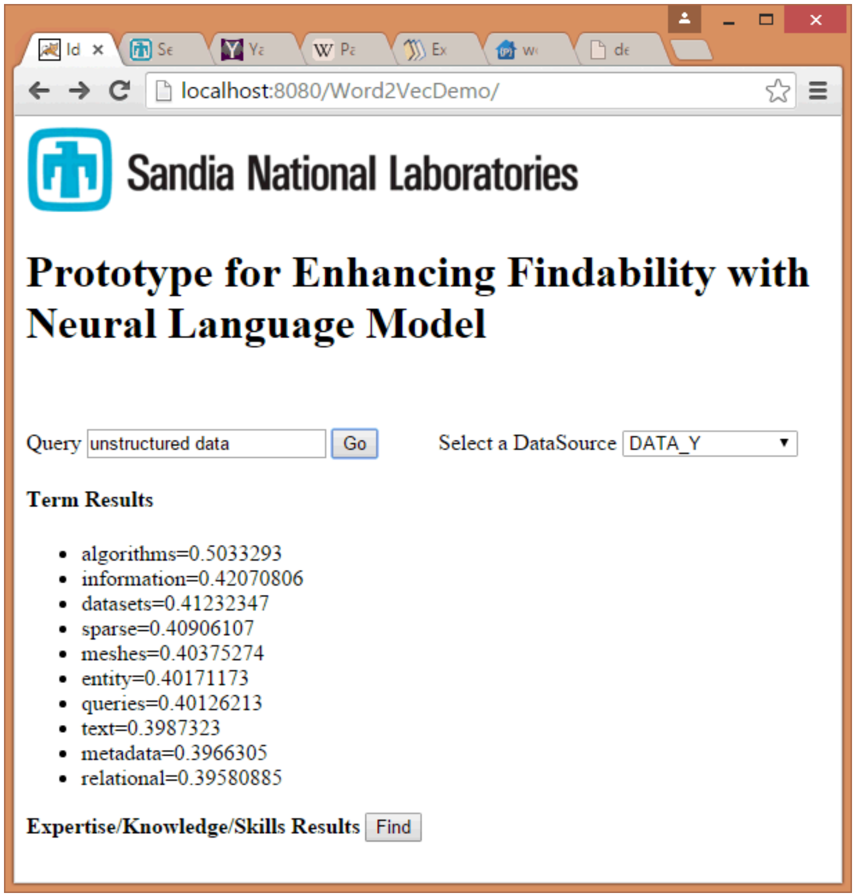


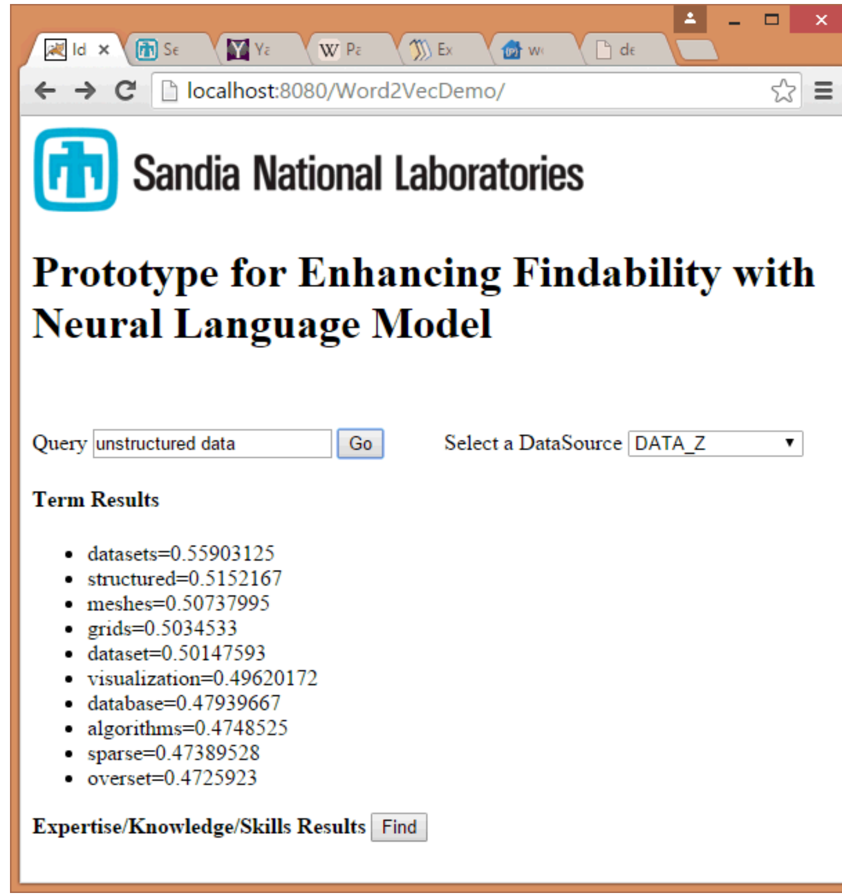


Challenges of Applying NNLM in Enterprise Search

- Domain Specific?
  - A NNLM trained from a domain may not be directly applied to other domains in a data repository with multiple domains
- Maintenance?
  - Data in an organization are updated dynamically, there is a need to update NNLM accordingly.

Differences among domains





Conclusions and Further Efforts

- Enterprise search can significantly improved in:
  - Deliver relevant information even the query terms not appear in the enterprise dataset
  - Increase the quality of retrieved information
  - Expand retrieved information to closely related documents
- Implementation of NNLM in Enterprise Search requires minor search engine reconfiguration
- Efforts are needed to improve training a NNLM to cover more domains in an enterprise data repository
- Further developments are needed to update NNLMs in real time