The 2016 International Conference on Computational Science and Computational Intelligence (CSCI'16), Dec 15-17, 2016, Las Vegas

Machine Learning Algorithms for Natural Language Semantics and Cognitive Computing

Dr. Emdad Khan
Dept. of Computer Science
MUM (Maharishi U. of Management)
Fairfield, Iowa, U.S.A
ekhan@mum.edu

Dr. Emdad Khan
InternetSpeech, Inc
San Jose, California, U.S.A
emdad@internetspeech.com

Abstract— **We present elegant machine learning algorithms to efficiently learn natural language semantics (MLANLP); thus enabling much better Natural Language Computing (NLC) and Cognitive Computing (CC). Our algorithms use human brain-like learning approach and achieve very good generalization on natural language (mainly text) data. Existing machine learning algorithms perform well on numerical data and cannot easily learn semantics of natural language. Such algorithms, however, can address well some specific problems of natural language, like Name Entity Recognition where data can be easily represented by numbers and semantics between words (name and entity) are simple. Besides, the generalization capabilities of existing machine learning algorithms are limited, especially for complex data. The generalization capability for learning semantics of natural language should be very good to ensure reliable NLC and CC. Our MLANLP has good generalization capability, and can also derive new semantics and knowledge, very much needed for NLC and CC.**

Keywords— *Machine Learning, Natural Language Computing, Cognitive Computing, Semantic Engine, Artificial Intelligence*

I. INTRODUCTION

Today's machine learning algorithms mainly address the learning of numerical data. These algorithms do not address learning of text data, where semantics between words, sentences and paragraphs are very crucial. All machine algorithms that exist today e.g. Neural Networks, Support Vector Machines, Probabilistic Learning, and Genetic Algorithms can be classified into 4 major categories – Supervised, Unsupervised, Reinforcement and Evolutionary Learning. Such algorithms use numerical data properties for learning key features helping mainly classification, regression and matching.

Such properties use common numerical features (e.g. diameter, weight or shape for coin recognition; or eyes, nose, mouth, eyebrows for face recognition [1]). Feature selection is very important in machine learning to ensure better results with good generalization. However, generalization can be inadequate for many such applications, especially when using complex data or large data or both.

On the other hand learning meaning of words, sentences, paragraphs and articles is the key for many natural language applications including information retrieval, question answering, summarization, reliable machine translation and drawing inference. All these problems use text i.e. non-numerical data. Classification, regression and matching - the major capabilities of existing numerical data driven learning algorithms - are not adequate for such semantics driven applications. The same is true for existing methods to define / learn semantics e.g. Predicate Logic, Ontology, or Frame. Such algorithms produce "mechanical" semantics where semantics are needed to be defined in a more crisp way and new semantics cannot be easily computed or derived from existing ones. Hence their use has been very limited - mainly works for small and medium size applications.

Thus, we need new machine learning algorithms to address the growing needs to handle text dominated data which contributes to 80% of data on the Internet (numerical data contributes the remaining 20%). In fact, we need a paradigm shift in machine learning approach for such semantics driven problems. Such a new paradigm should be able to use / learn semantics of words, compute the semantics of sentences using the semantics of the words, compute the semantics of a paragraph using the semantics of sentences, and compute the semantics of a document using the semantics of its paragraphs. The machine learning algorithms to efficiently learn natural language semantics (MLANLP) presented in this paper supports this new paradigm. In addition to learn and compute semantics, MLANLP can also derive new semantics and new knowledge from existing ones.

Section II discusses the new ML (Machine Learning) paradigm to address unstructured data, MLANLP and high level algorithms. Section III discusses SEBLA (Semantic Engine Using Brain-Like Approach) based semantics. Section IV describes ML algorithms for semantics. Section V describes some applications and Section VI provides conclusion and future works.

II. Machine Learning Paradigm to efficiently Learn Semantics of Unstructured Data

The key capabilities, namely, classification, regression and matching of existing numerical data driven machine learning algorithms are not adequate for natural language processing, natural language understanding, natural language computing and cognitive computing. An inherent property of any natural language computing and cognitive computing is the semantics. Machines need to compute semantics - properly use the word semantics to compute the meaning of a sentence; and use the meaning of sentences to compute the meaning of a paragraph; and use the meaning of paragraphs to compute the meaning of the whole document. Machines also need to learn (and refine learning) semantics, especially the semantics of words. One common important element for all these is learning how to derive (i.e. learn) new semantics and new knowledge. This is more needed for cognitive computing.

Moreover, proper **actions** need to be learned or computed based on the meaning of a sentence or query. Thus, ML for unstructured data would need to learn logic and determine appropriate actions. Hence, the new paradigm need to focus on semantics driven and logic driven learning as opposed to existing numerical data driven learning that usually minimizes error with respect to some objective function. MLANLP is designed to have all these new desired properties.

Semantics driven and logic driven learning paradigm mainly uses good explanation / logic rather than training using a large dataset. It is similar to the way human learn. We learn very easily when someone explains or teaches. A few examples may be used to enhance the teaching process. But our main part of the learning is based on explanation and logic. We do not use numerous data to learn something unless it is for regression. In contrast, numerical data driven ML systems learn from many examples of numerical data and do not use explanation or logic to learn. This is the main reason such systems can only do limited key functions like regression, classification and matching, and with limited generalization.

Another key point in the paradigm shift is that semantics driven learning process uses computing and learning at the same time for many cases. This makes good sense as semantics and semantics driven computation and learning / understanding are very closely related. In fact, for most cases computing is learning. Learning can be considered separated from computing to some extent when we derive new semantics and knowledge from existing semantics and existing knowledge.

The new paradigm also learns how to refine the meaning of words using WF (Word Feature) & WK (World Knowledge) tables and natural language corpora.

The generalization capability of Semantics driven and logic driven learning paradigm is much better as learning is dominated by semantics and logic. Since the semantics of words are the basic building block of learning and computation, there is also no need for a large natural language corpora to learn. In contrast, such large natural language corpora is very much needed when we use **probability based N-grams.** However, natural language corpora can still be used to further refine the semantics as appropriate.

MLANLP uses above mentioned new paradigms and thus uses human learning approach. The learning and computing capabilities of MLANLP enable it to determine the actions to be performed based on the input sentence(s). This means that the input is used to determine how to come up of the logic to use. In other words, programmer needs to write a short high level programs and the MLANLP determines and performs the lower level tasks automatically using the semantics of the input information. The following 3 examples show how MLANLP deals with a common request or a question:

[Example -1]

"Show me the pictures of last Saturday birthday party from my Facebook account",

Then MLANLP will perform the following:

a. Go to the Facebook website and log on (assuming that login / password information was already in the system).
b. Calculate the date for last Saturday considering today's date from the system. Then search the Facebook page for Birthday party pictures with the specified date.
c. Then identify the pictures that are more relevant using the title / subtitle / caption of the pictures.
d. Then present the requested pictures in a nice presentable form to the user.

[Example 2]
"How many people died in the major accident on highway 280 in San Jose today?"

In this case no specific actions are identified by the sentence. So, MLANLP will go to a common search engine and search for the above sentence. It will then evaluate all key results with highest "semantic match" with the query, and provide the most relevant filtered results to the user.

[Example 3]
"How many students graduated from Stanford University in Computer Science in 2015?"

MLANLP will determine the following actions:

a. Go to Stanford University website.

b. Find out the "search box" and enter "students graduated in Computer Science in 2015"
It will then try to evaluate the key content of this page using the semantics e.g. the on-site search for the above partial sentence will provide the following results:

Stanford University: Common Data Set **2015**-2016

CS Commencement | Stanford **Computer Science**

more……

c. Since no information on students graduated is found in the search results shown above, MLANLP will assume that the information is not in the public domain and

d. Advice the user to log on.

e. If user provides logon and password information, MLANLP will logon and then repeat above process. If it can find the information from the on-site results semantically related to the request, it will try to find the requested information from the search results using similar semantic matching process – otherwise it will say "Requested results no found".

Note that this is a very complex example.

As mentioned, MLANLP can derive new semantics and new knowledge. Please refer to Section 4 for more.

III. SEBLA (SEMANTIC ENGINE USING BRAIN-LIKE APPROACH) BASED SEMANTICS

While traditional approaches to NLU have been applied over the past 50 years and had some good successes mainly in a small domain, results show insignificant advancement, in general, and NLU remains a complex open problem. NLU complexity is mainly related to **semantics**: abstraction, representation, real meaning, and computational complexity. We argue that while existing approaches are great in solving some specific problems, they do not seem to address key Natural Language problems in a practical and natural way. In ([2],[3]) we proposed a Semantic Engine using **Brain-Like approach (SEBLA)** that uses Brain-Like algorithms to solve the key NLU problem (i.e. the semantic problem) as well as its sub-problems.

The main theme of our approach in SEBLA is to use each word as object with all important features, most importantly the semantics. In our human natural language based communication, we understand the meaning of every word even when it is standalone i.e. without any context. Sometimes a word may have multiple meanings which get resolved with the context in a sentence. The next main theme is to use the semantics of each word to develop the meaning of a sentence as we do in our natural language understanding as human. Similarly, the semantics of sentences are used to derive the semantics or meaning of a paragraph. The 3rd main theme is to use natural semantics as opposed to existing "mechanical semantics" of Predicate logic or Ontology or the like.

A SEBLA based NLU system is able to:

1. Paraphrase an input text.
2. Translate the text into another language.
3. Answer questions about the content of the text.
4. Draw inferences from the text.

At the word level, the key question we have addressed is how to represent the semantics for each word and how to associate appropriate world knowledge with each word. By using the representation and semantic feature of each word, along with the world knowledge associated with each word, the meaning of a sentence is derived by applying the grammar of the language and appropriate rules to combine words. Key features of the words and appropriate rules to combine them are learned / refined using large text corpora and existing machine learning algorithms (both existing ML and ML for learning semantics). The inference engine (i.e. the computing engine or Intelligent Agent) determines the meaning of a sentence by using the word semantics and appropriate rules to combine the words in a sentence.

Key features of a word are the features that basically defines it. For example, the key features of a ball are:

- - something that is round,
- - something that rolls,
- - something that bounces when it hits a wall

The color of the ball is a secondary feature as one can identify whether an object is a ball or not without using its color. So, a ball is represented in the Word Feature table as
{Move, roll, round, bounce back, play,..}
which is also its semantics.

IV. MACHINE LEARNING ALGORITHMS FOR SEMANTICS

ML algorithms for semantics are very different than ML algorithms for learning from numerical data. Semantics, computing using semantics and associated learning are sort of combined i.e. learning and computing are integrated. This is similar to when a teacher teaches something. Students compute the meaning of sentences and combine them (i.e. compute) to learn. Students may ask questions to clarify and teacher may use a few additional sentences (may be similar) to further explain. In this case combining the semantics of related sentences results some new understanding i.e. new meaning. In the process of such combining operation, human use World Knowledge, experience and more. Computing involves use of logic to derive new meaning and knowledge. There is still "training" (i.e. teaching) and "testing" (i.e. checking whether students have learned) phases but computation process is mainly driven by logic and semantics rather than, for example, using differential equations to minimize error with respect to an objective function which may take several cycles.

Looking at it from a different angle, we can say that ML algorithms to learn semantics and derive new meanings / semantics, directly use the semantic property from the input

sentences along with appropriate logic. Testing process is similar where new input sentences (usually in the form of questions) are used to see the understanding i.e. answering questions properly or deriving actions properly or both.
The semantics of the input sentences carry the real information to learn and the computing process using logic and semantics finally computes and learns.

When purely new semantics and new knowledge are derived (very important for Cognitive Computing), we can see how learning steps (and hence Learning Algorithms) and computing steps can be separated to some extent even though there still remains a good overlap.

4.1 Learning / Computing Algorithms for Semantics

We use SEBLA based semantics and semantic computation. Basic semantics for a word is defined using the features of the word as described in Section 3. The algorithms to calculate the meaning of a sentence and meaning of a paragraph are shown below:

Algorithm to Calculate Semantics:

A. Sentence Level

A1. Simple Sentences

We consider each parsed word in a sentence. For each of such words,

a) Get the Function words (WF)

b) Get the World Knowledge (WK) words

E.g. for the word "ball", the function words are {ball, move, roll, round, play..} as mentioned above.

Similarly, the function word for "go" is {go, move, not static,..} and the function word for "school" is {school, study, student, teacher, learn}. We can add other related words which are usually implied – e.g. for "school", "a place to" {study}, a place where {students} go etc. But in general a short list of function words suffices and makes it simpler. *Note that the word itself is included in its Function word. This can be done in the WK and thus, we may not include the word itself in its Function word.*

Now let's consider the sentence,

"I go to school". For semantic retrieval, we will use "I go school" as simplifying the verb phrase (VP) "go to school" to "go school" helps. This is a **"declarative"** type sentence.

From language standpoint, the word "I" (noun Phrase, NP) is the subject. "go" is the verb which is part of VP "go school" where "school" is a noun. Now, we need to apply the function words to calculate the semantics of the first two words i.e. "I go". In doing so, we first take the Function words for "I" and Function words for "go". Thus, we have,

I {person, he , she, living object,.. | eat, go, fly, all verbs} go {move, walk, run,..}. Then, we take only the subject words for "I" and verb words for "go", which yields the following **semantics**,

I {person} go {move, walk, run,..} ……… (1)

Similarly, the semantics of the sentence

"I go to school" is

I {person} go {move, walk, run,..} school {study, student, teacher, learn} ……………………………...….(2)

Note that the main words are there to visualize it better. The real semantics is represented by all words under the curly braces {}.

Consider another sentence,

"I open door", the semantics of which is

I {person} open {open, unlock, push, pull,..} door {a thing that blocks, close, open, move,…} …………… (3)

We can now ask a question like

" I am doing what". The semantics of this sentence is

I {person} doing {unlock, push, pull, open .. all verbs as included in WK} what {question} …………. (4)

A match operation between equations (3) and (4) will yield

I {person} doing-open {unlock, push, pull,..} what-door {question} ………………………………... (5)

The match between "doing" and "open" is true as "doing" includes all verbs as included in World Knowledge (WK).

Note: For other words (i.e. without an auxiliary verb) that imply open (e.g. ajar), the answer will be same as "ajar" would be included in the Function word of "open". In this case, we do not need to use the WK but it won't hurt even if it is in the WK.

Equation (5) can then be processed to yield the **answer**

"I am opening the door",

after some refinement using grammar.

To better explain this, let's use an invalid sentence, e.g.

"Door walks"

which is not valid as in the Function words for "door", "walk" is not there. Besides, "door" is not a living thing and hence it will not be supported by the WK. So, the semantics of this would be NULL or a question mark "?".

A similar approach is used for more complex sentences. Usually a long sentence is broken into smaller sub-sentences and "action words" are determined. Action words are mainly verbs that determine the key actions to be followed by computing, its relation with other words and the logic.

Our SEBLA based approach shown above for **"declarative"** sentences will work in a similar way for other types of English (and other languages) sentence including **"imperative", "yes/no", "wh-structure", and "wh-nonsubject-structure".**

A2. More Complex Sentences

SEBLA based approach also works with more complex sentences. Consider the sentence,

"I am trying [VP (Verb Phrase)] to find a flight that goes from Pittsburgh to Denver after 2 pm [VP]"…………………………………….………(6)

Note that the sentence starts with Noun [I], then VP [am trying] and then another VP [to find a flight that goes from Pittsburgh to Denver after 2 pm]. The WF in the basic sentence and the question will automatically match and so, we do not necessarily break the sentence as mentioned above. This is what traditional Semantic Role labeling and parsing try to do so that there is a clear structure of the sentence which makes it easier to formulate an answer to a question.

So, it is sort of S -> NP (VP VP) i.e. constituent of VP is another VP OR think it like S -> NP VP

But VP (sort of Big VP) has 2 VPs i.e. "am trying" and "to find a flight that goes from Pittsburgh to Denver after 2 pm".

Here, the basic idea is to use the sentence starting at top level and classify it as having a Noun Phrase (NP) and Verb Phrase (VP). Then, deal with the complexity of the VP using similar way as described above. So, the first level semantics is

I {person} trying {doing something, working on, } to find {looking, trying to look,…} ……………..(7)

as the main verb of the VP "find a flight that goes from Pittsburgh to Denver after 2 pm" is "find" or "to find". Now, we can focus on "a flight that goes from Pittsburgh to Denver after 2 pm" in a similar way.

This reduces to "flight goes" as the rest i.e. "from Pittsburgh to Denver after 2 pm" is from a city to another city after "time" 2 pm. The semantics for the words before the cities is

I {person} trying {doing something, working on, trying… } to find {looking, trying to look,…} flight {a plane going from one city to another,..} ………………… (8)

B. Paragraph Level

Similar algorithms can be used in calculating semantics for multiple sentences and paragraphs. However, some modifications are needed for the following reasons:

1. Within a sentence, words are used in a constrained way using grammar. But between sentences there is no such grammar.
2. Usually, a group of sentences carry a theme within a context and there are ***relations*** between sentences.

Thus, to calculate the semantics between sentences, we use word semantics as before BUT with some modifications. This is also true for a single long sentence segmented by "comma", "semicolon", "but", "as" and the like. We also need to take account **for "discourse" i.e. coherence or co-reference to words in previous sentences**. There are some good existing solutions mainly for a small domain problem. But, in general Computational Discourse (CD) in natural language is an unsolved problem. However, with our SEBLA based scheme, the CD problem can be solved to a good extent for large domains.

In calculating semantics in a long sentence, the previous, next and other words can further influence / refine the semantics. For convenience, we have included this aspect in calculating semantics of multiple sentences as described before.

4.2 Deriving New Semantics and New Knowledge

To derive new semantics and new knowledge, we use a similar approach with some enhancements, mainly causal relationships and logic. Consider the following 3 sentences:

I was tired. I fell asleep. Did work and made some good progress after I woke up. …………(9)

Here, we need to find causal relationships by using the semantics between sentences. The semantics of "tired" and "asleep" are related. Then from WK, we know that "tired" caused / may cause "asleep". Thus, the causal relationship between first 2 sentences are established. In the 3rd sentence "After I woke up" is opposite of "asleep" and hence a relation is established. And "made some good progress" is related with not "tired" in the first sentence. The basic information in

WK will help compute the causal relationship between first sentence and 3rd sentence. Thus, SEBLA based semantics approach will compute the new derived fact

"I made good progress" because my "tiredness" was gone due to "sleep".

This is an important feature for question answering, drawing inference, summarization where sentences can be shortened / shrunk (i.e. for all NLC and CC that absolutely need new fact or new knowledge).

V. KEY APPLICATIONS

MLANLP algorithms for semantics described above can be used in numerous applications. A few key application areas are Intelligent Search (that can produce under 50 search results as opposed over million hits for simple search like "Low price Thai Restaurant in Silicon Valley" ([4],[5]), Intelligent Information Retrieval, Question Answering, Summarization, Drawing Inference of an article, and more accurate Machine Translation ([6],[7],[8]).

VI. CONCLUSIONS AND FUTURE WORKS

It is important to address the issue of learning semantics in an effective and efficient way to address many problems related to unstructured data and mixed data. Unstructured data (mainly text data) dominates the big data world including the Internet. We have presented MLANLP (machine learning algorithms to efficiently learn natural language semantics), an elegant algorithm that uses a new paradigm to learn semantics in an effective way. MLANLP uses SEBLA (Semantic Engine using Brain-Like Approach) to learn semantics of sentences, paragraphs and documents by using the basic word semantics - similar to what we do as human. MLANLP also uses semantics of words & sentences, and logic to derive new semantics and knowledge.

MLANLP is critical for Natural Language Computing (NLC) and Cognitive Computing (CC) – the two key aspects essential for many Intelligent Systems that need derivation of new semantics and knowledge. Today's machine learning algorithms are numerical data driven, and mainly capable of regression, classification and matching. In contrast NLC & CC are mainly text data driven and need to use semantics for learning and all computations. The learning in MLANLP is fast as it uses semantics and logic directly. It is similar to learning by a teacher via explanation.

There are many applications of NLC and CC including intelligent information retrieval, serving basic requests, question answering, summarization, drawing inference and more accurate machine translation. Our main future works are:

(a) Further improving the process of deriving new semantics, logic and knowledge. The high generalization capability of MLANLP will help this process.
(b) Apply MLANLP in more applications.

References

[1] Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471#.mf9o89dwa

[2] E. Khan,: Natural Language Understanding Using Brain-Like Approach: Word Objects and Word Semantics Based Approaches help Sentence Level Understanding, US Patent files 2011–US Patent Publication 2014/0032574A1

[3] E. Khan, "*N*atural Language Processing: Key for Next Generation Big Data and Data Science" Plenary talk *at 15th International Conference on Applied Computer and Applied Computational Science, March 18-20, 2016* http://www.wseas.org/wseas/cms.action?id=12693

[4] E. Khan, "Next generation web - intelligent search, question answering, summarization and more", INTERNATIONAL JOURNAL of COMPUTERS AND COMMUNICATIONS, (NAUN & UNIVERSITY PRESS), Vol. 9, June 2015.

[5] E. Khan, " Intelligent Internet: Natural Language and Question & Answer based Interaction", INTERNATIONAL JOURNAL of COMPUTERS AND COMMUNICATIONS, (NAUN & UNIVERSITY PRESS) Oct. 2013.

[6] C. Eaton et al, "**Understanding Big Data: Analytics for enterprise class Hadoop and Streaming Data**", http://public.dhe.ibm.com/common/ssi/ecm/en/iml14296usen/IML14296USEN.PDF

[7] P. Ryan et al, "**The Problem of Analyzing Unstructured Data**", Grant Thoronton, 2009,

[8] E. Khan, "Processing Big Data with Natural Semantics and Natural Language Understanding using Brain-Like Approach", INTERNATIONAL JOURNAL of COMPUTERS AND COMMUNICATIONS, (NAUN & UNIVERSITY PRESS) January 2014.