Module 6: Natural Language

This is a single, concatenated file, suitable for printing or saving as a PDF for offline viewing. Please note that some animations or images may not work.

Learning Objectives

This module explores the relationship of AI to natural (i.e., ordinary) language. Natural language (NL) has always been part of AI but two things have recently made an NL a key technology—if not part of our lives. The first is the recognition that machine learning is extraordinarily helpful for this field; the second (not entirely independent) is the fact that people have grown accustomed to speaking to devices.

After successfully completing this module, you will be able to do the following:

1. Frame the natural language enterprise.
2. Enable role of grammars.
3. Use tools.
4. Explore logic approaches.
5. Leverage neural net approaches.

Module 6 Study Guide and Deliverables

Module Theme:	Natural Language; Integration of AI techniques
Readings:	• Module 6 online content • Russell & Norvig Chapter 23 (Natural Language Processing), concentrate on Section 23.1
Assignments:	• Optional Lab 6 due Sunday, October 17 at 6:00 AM ET • Assignment 6, due Wednesday, October 20, at 6:00 AM ET
Live Classrooms:	• Wednesday, October 13 from 8:00 PM to 9:00 PM ET • Thursday, October 14, from 8:00 PM to 9:00 PM ET • Live Office: Wednesday and Thursday after Live Classroom, for as long as there are questions

Introduction

We begin with discussions of analytical approaches to NL and cap this with the application of neural Nets.

Speech Act

It can be useful to think of natural language in terms of "***speech acts***"—something that involves the source, the content, and the destination.

Speaker → Utterance → Hearer

There is no preconceived limit to the content size of a speech act.

Example:

```
From an AI course, you can learn how to create intelligent applications.
"Intelligent" means that the application's output is informed by human
understanding.
```

Some Types of Speech Act

To achieve a speaker's goals, there are several kinds of speech acts as shown below. Although they all contain words, they are very different.

Inform:	`There's a pit in front of you`
Query:	`Can you see the gold?`
Command:	`Pick it up`
Promise:	`I'll share the gold with you`
Acknowledge:	`OK`

Requires Knowledge of …

1. Situation
2. ***Semantic*** and ***syntactic conventions*** (make sure that you know the difference)
3. Hearer's …
   - goals
   - knowledge base
   - rationality

Source: Adapted from Russell & Norvig.

Every speech act exists in some context—the same speech act can mean something entirely different in a different context. A sequence of words obeys syntactic conventions like programming languages: in other words, it must have a recognized format.

At the same time, it must convey meaning—it's semantics.

In addition to the context, semantics, and syntax, the better we understand the goals, knowledge, and reasoning of the hearer and of the speaker, the better we can interpret and process an utterance.

Stages of Informing: Speaker and Hearer

The figures show the roles of the speaker (S) and the hearer (H).

Stages of Informing: Speaker

Intention `S wants to inform H that P`

Generation `S selects words W to express P in context C`

Synthesis `S utters words W`

Source: Adapted from Russell & Norvig.

Stages of Informing: Hearer

Perception `H perceives W' in context C'`

Analysis H infers possible meanings $P_1 \ldots P_n$

Disambiguation H infers intended meaning P_i

Incorporation H incorporates P_i into KB

Source: Adapted from Russell & Norvig.

How Relations Are Expressed in English

Continuing this analytic approach, we can measure how often a verb, for example, occurs, or a noun/preposition sequence such as "here of."

Table: Eight General Templates that Cover about 95% of the Ways that Relations are Expressed in English

Type	Template	Example	Frequency
Verb	$NP_1 \;\; Verb \;\; NP_2$	X established Y	38%
Noun-Prep	$NP_1 \;\; NP \;\; Prep \;\; NP_2$	X settlement with Y	23%
Verb-Prep	$NP_1 \;\; Verb \;\; Prep \;\; NP_2$	X moved to Y	16%
Infinitive	$NP_1 \;\; \mathbf{to} \;\; Verb \;\; NP_2$	X plans to acquire Y	9%
Modifier	$NP_1 \;\; Verb \; NP_2 \;\; Noun$	X is Y winner	5%

Source: Adapted from Russell & Norvig.

Type	Template	Example	Frequency
Noun-Coordinate	NP_1 (,| **and** | - | :) NP_2 NP	X-Y deal	2%
Verb-Coordinate	NP_1 (,| **and**) NP_2 $Verb$	X, Y merge	1%
Appositive	NP_1 NP (:| ,)? NP_2	X hometown: Y	1%
Source: Adapted from Russell & Norvig.			

Grammars

The content in this module is adapted from Russell & Norvig (2020).

This section reviews grammar in natural and formal language—its structure.

A ***grammar*** is defined by a vocabulary—e.g., {S, p, q}—and a set of ***rewrite rules***, as shown below.

Example: grammar G={S, p, q}, with productions

S → pSp

S → qSq

S → ε.

Typical derivation:

S → pSp → ppSpp → ppqSqpp → ppqqpp

The next shows a common rewrite rule. It says that every sentence consists of a noun phrase followed by a verb phrase. The symbols used for the latter are sufficient at this level (they are expressive, but you could use any symbols you want, actually.

The grammar is a set of rewrite rules, e.g.,

$$S \rightarrow NP \;\; VP$$

Here S is the sentence symbol, and NP and VP are nonterminals.

(Backus Nauer Form)

The example shows the set of rewrite rules that specify expressions of the form $a^n b^n c^n$.

Example: Grammar for $\{a^n b^n c^n : n \geq 1\}$: e.g., $a^3 b^3 c^3$

```
1.    S → a B C

2.    S → a S B C

3.  C B → C Z

4.  C Z → W Z

5.  W Z → W C

6.  W C → B C

7.  a B → a b

8.  b B → b b

9.  b C → b c

10. c C → c c
```

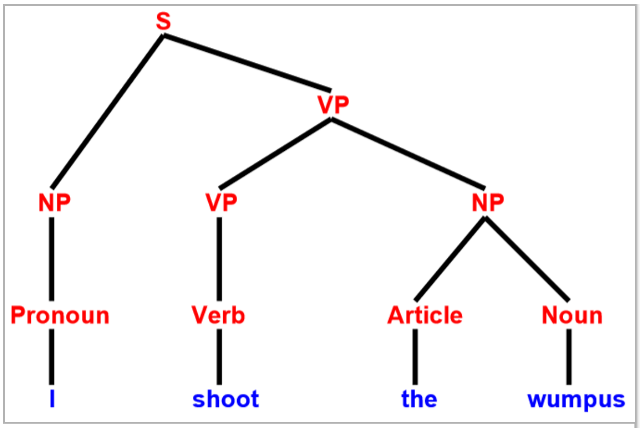
S
→2 **aSBC**
→2 **aaSBC**BC
→1 aaa**BC**BCBC
→3 aaaB**CZ**CBC
→4 aaaB**WZ**CBC
→5 aaaB**WC**CBC
→6 aaaB**BC**CBC
→3 aaaBBC**CZ**C
→4 aaaBBC**WZ**C
→5 aaaBBC**WC**C
→6 aaaBBC**BC**C
→3 aaaBB**CZ**CC
→4 aaaBB**WZ**CC
→5 aaaBB**WC**CC
→6 aaaBB**BC**CC
→7 aa**ab**BBCCC
→8 aaa**bb**BCCC
→8 aaab**bb**CCC
→9 aaab**bb**cCC
→10 aaabbb**cc**C
→10 aaabbbc**cc**

Source: https://en.wikipedia.org/wiki/Context-sensitive_grammar#Examples

Wumpus NL Parse Tree Example

Russell and Norvig developed a simple language for Wumpus World, defined by the figure.

Example: Wampus World Lexicon



The following examples show the complete grammar for Wumpus World. The first shows the terminals and the second the nonterminals.

Example: Wampus World Lexicon

Noun → *stench* | *breeze* | *glitter* | *nothing*
| *wumpus* | *pit* | *pits* | *gold* | *east* | ...
Verb → *is* | *see* | *smell* | *shoot* | *feel* | *stinks*
| *go* | *grab* | *carry* | *kill* | *turn* | ...
Adjective → *right* | *left* | *east* | *south* | *back* | *smelly* | ...
Adverb → *here* | *there* | *nearby* | *ahead*
| *right* | *left* | *east* | *south* | *back* | ...
Pronoun → *me* | *you* | *I* | *it* | *S/HE* | *Y'ALL* ...
Name → *John* | *Mary* | *Boston* | *UCB* | *PAJC* | ...
Article → *the* | *a* | *an* | ...
Preposition → *to* | *in* | *on* | *near* | ...
Conjunction → *and* | *or* | *but* | ...
Digit → *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9*

Wumpus World Grammar

Grammar (continued)	Example
S → *NP VP*	I + feel a breeze
| *S Conjunction S*	I feel a breeze + and + I smell a wumpus
NP → *Pronoun*	I
| *Noun*	pits
| *Atricle Noun*	the + wumpus
| *Digit Digit*	3 4
| *NP PP*	the wumpus + to the east
| *NP RelClause*	the wumpus + that is smelly
VP → *Verb*	stinks
| *VP NP*	feel + a breeze

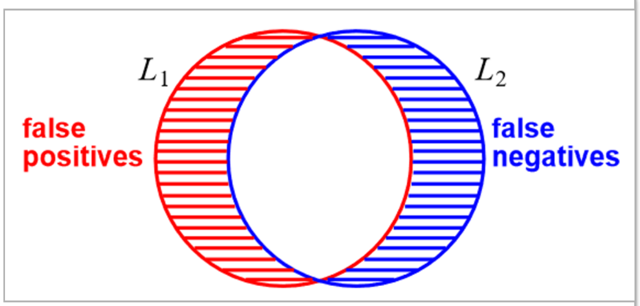
```
    | VP Adjective          is + smelly
    | VP PP                 turn + to the east
    | VP Adverb             go + ahead

PP → Preposition NP         to + the east

RelCluse → that VP          that + is smelly
```

Comparing Formal (L_1) & Natural (L_2) Language

The following figure classifies problems with using a grammar for natural language. There are natural language utterances that make good sense to humans but that the grammar can't parse (false negatives); conversely, there are expressions that the grammar can parse but that make no sense in the real world (false positives).



Adjusting $L-1$ to agree with L_2 is a learning problem!

- the gold grab the wumpus
- I smell the wumpus the gold
  I give the wumpus the gold
- I donate the wumpus the gold

Syntax → Meaning in NL?

Recognizing the legality of an utterance according to a grammar is just the beginning: we still need to assign meaning to the utterance. In other words, the utterance must belong within the real world. That typically means connecting it with concepts already known to users (readers, listeners etc.).

Example: "Mary hit John" $\neq$ "John hit Mary"

"And since I was not informed—as a matter of fact, since I did not know that there were excess funds until we, ourselves, in that checkup after the whole thing blew up, and that was, if you'll remember, that was the incident in which the attorney general came to me and told me that he had seen a memo that indicated that there were no more funds."

Recall the Wide Variety of Contexts

The following figures remind us that there can be several—perhaps many—contexts for an utterance. Parsing does not distinguish them well—if at all.

Intention S wants to inform H that P
Generation S selects words W to express P in context C
Synthesis S utters words W

Perception H perceives W' in context C'
Analysis H infers possible meanings $P_1 \ldots P_n$
Disambiguation H infers intended meaning P_i
Incorporation H incorporates P_i into KB

Source: Adapted from Russell & Norvig.

Tools

Tools for dealing with natural language have multiplied on the Web. A good example of a tool is from Google (https://cloud.google.com/natural-language), where the figures show a typical analysis session. It goes well beyond syntax, and provides semantic options.

Figure: Natural Language AI from Google

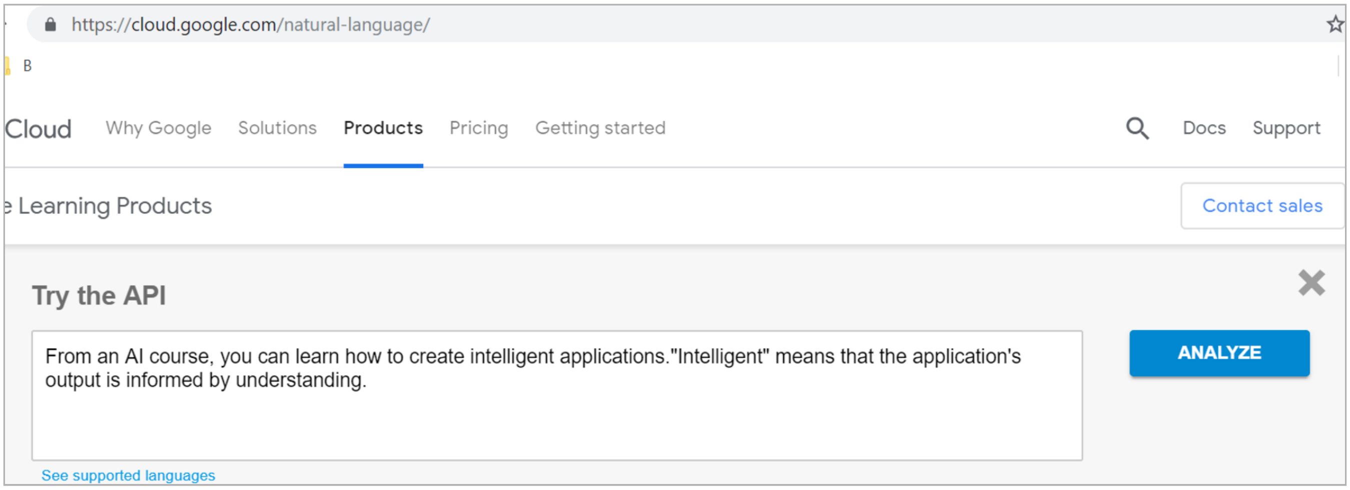


Figure: Natural Language AI from Google, *Entities* in an Utterance

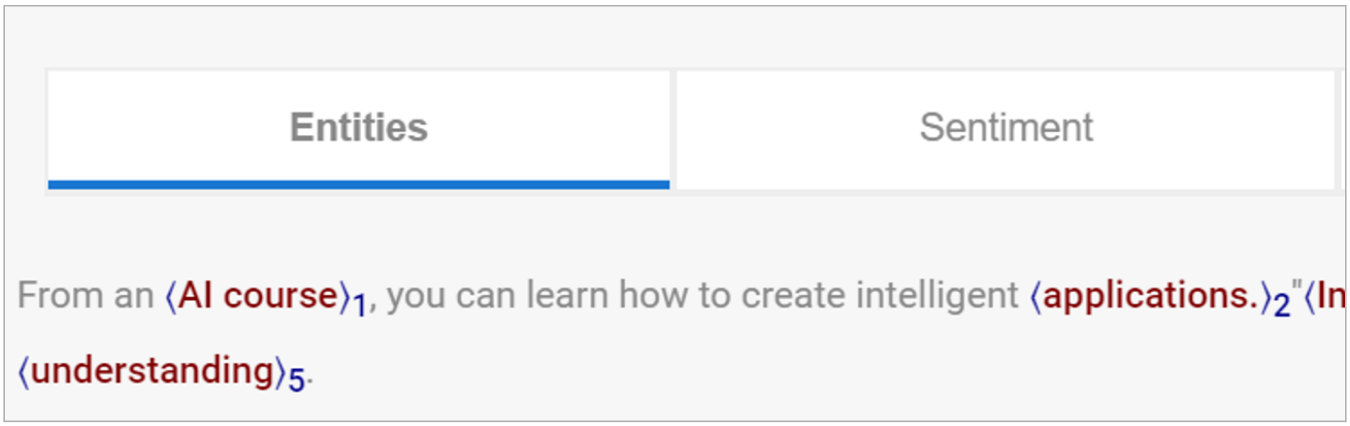
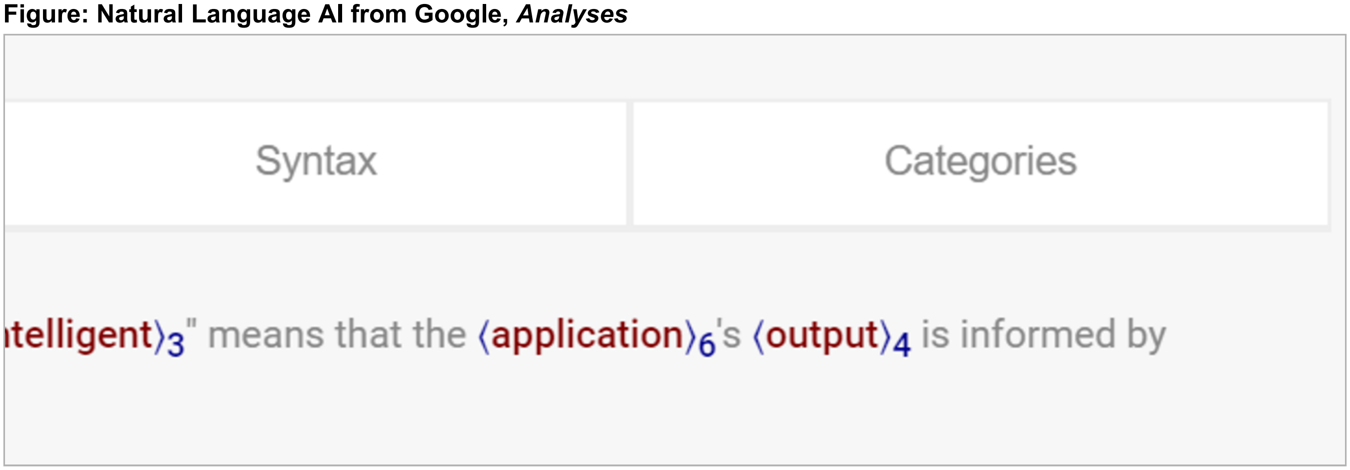


Figure: Natural Language AI from Google, *Analyses*



This figure shows its enhanced parsing.

Figure: Natural Language AI from Google, *Parsing* Example—First Part

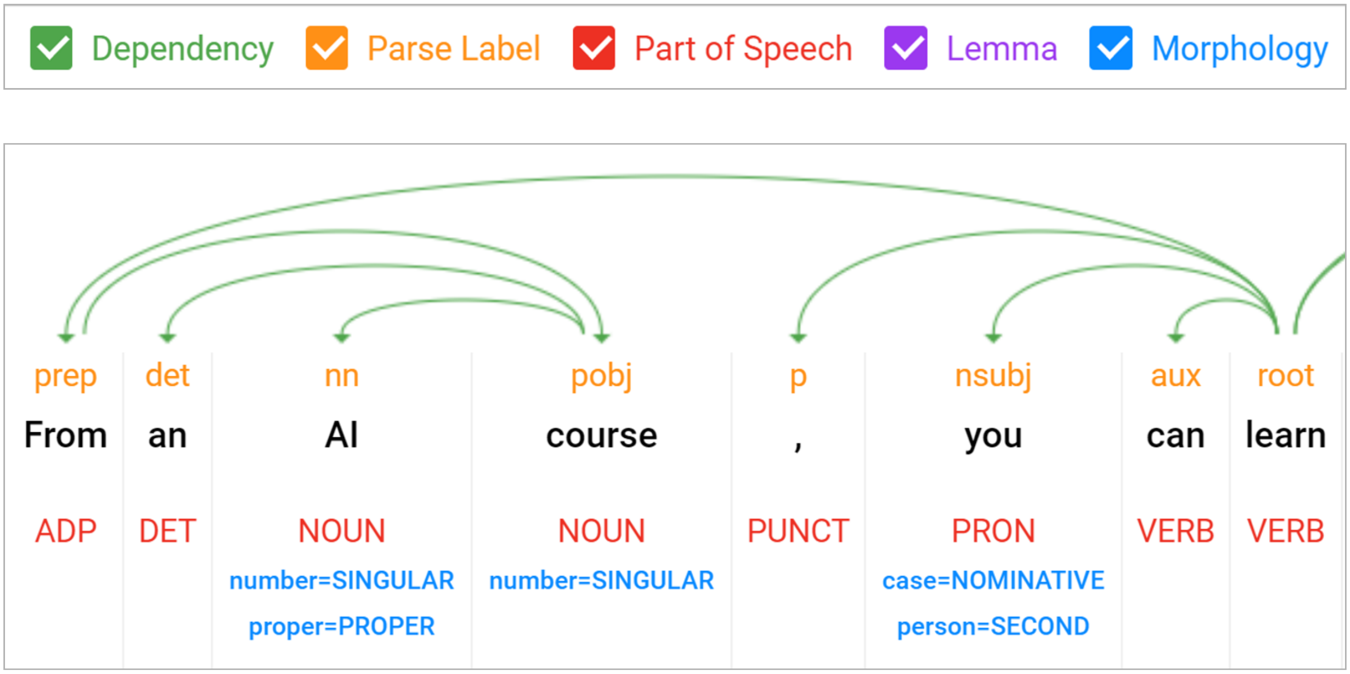
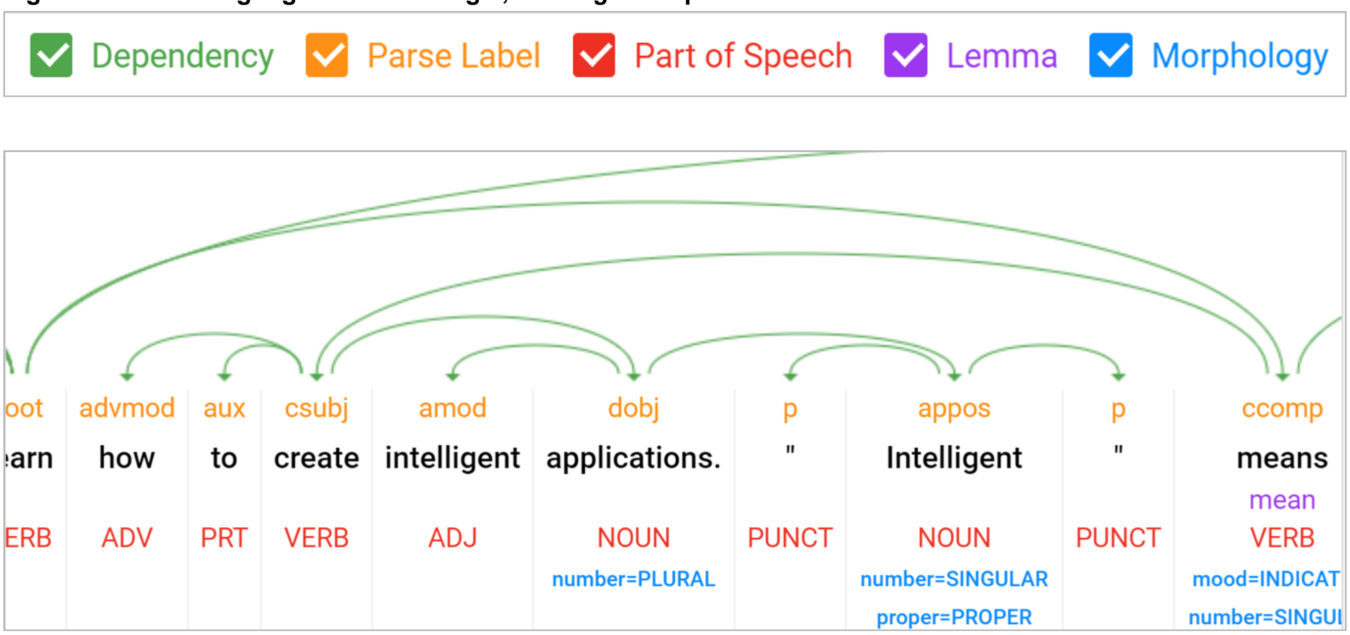


Figure: Natural Language AI from Google, *Parsing* Example—Second Part



Sentiment Analysis for a Document

To handle semantics in the real world, it is often useful to understand how the user is feeling. This is partially captured by sentiment analysis.

Sentiment Analysis for a Document

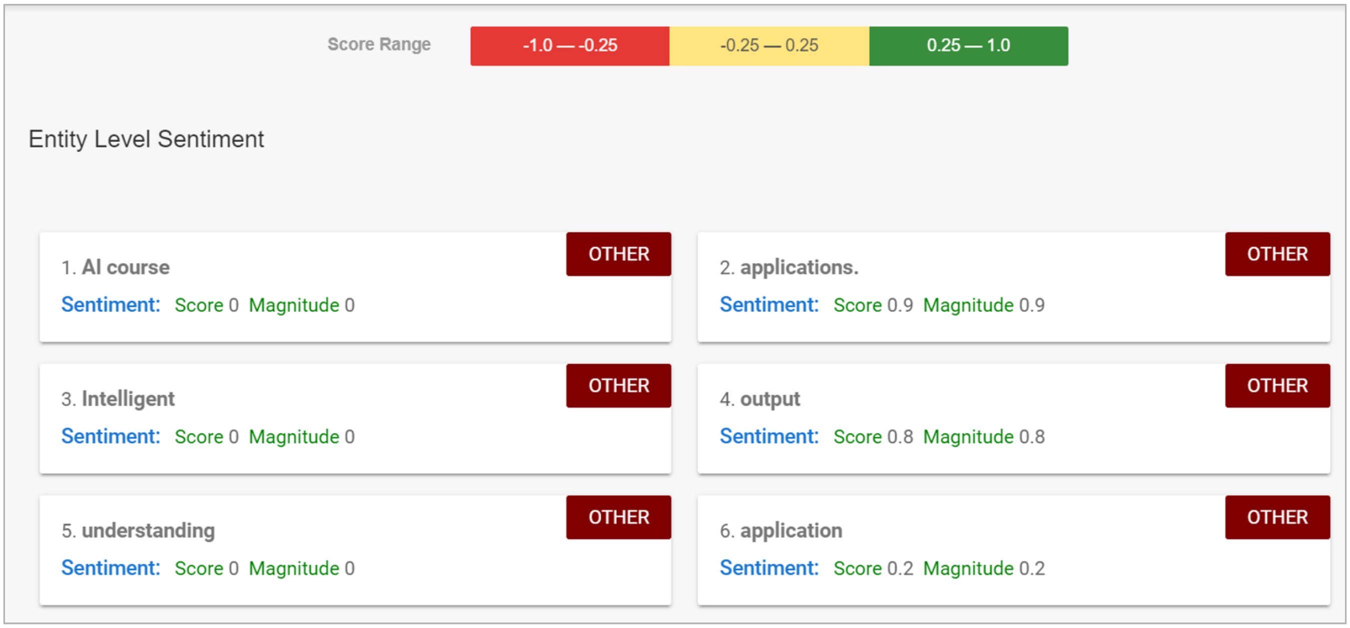
score → *overall emotion*

magnitude → how much emotional content

Source: Adapted from [Google Natural Language AI]

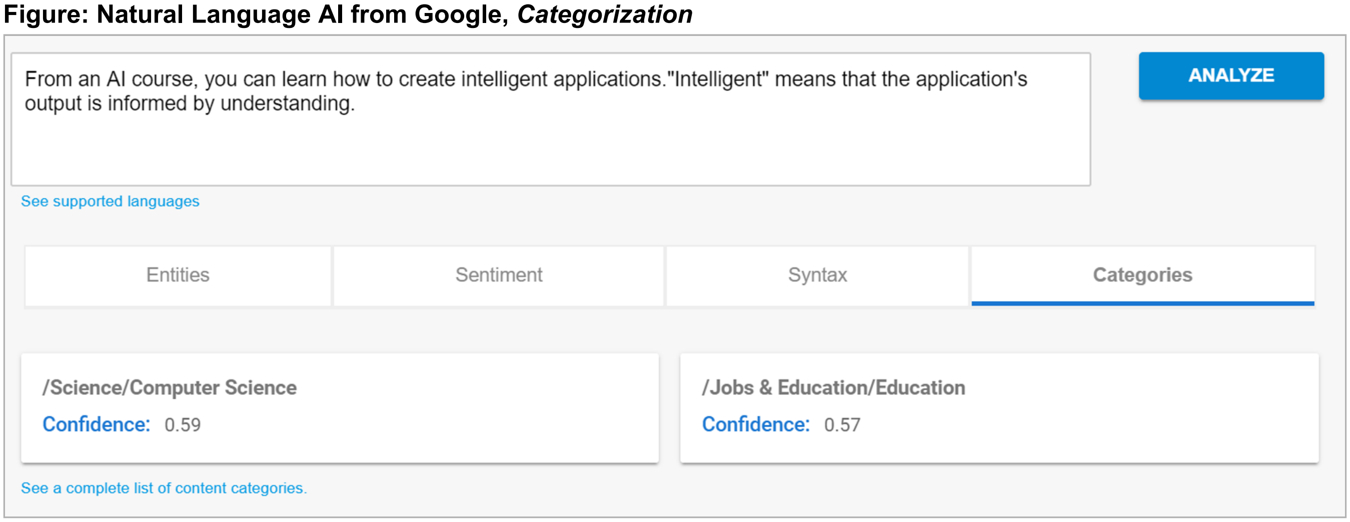
The color coding etc. can be translated into useful API's, enabling a degree of natural language understanding to be integrated into applications.

Figure: Google Natural Language, *Sentiment Analysis*



Tools like this are a reminder that there are many aspects to semantics. For example, simply categorizing an utterance can be much of what is needed for a given application. Contrary to early expectioans in AI research, word-by-word semantic analysis may be of limited importance—even unnecessary.

Figure: Natural Language AI from Google, *Categorization*



Application: TEXTRUNNER

There are many natural language applications, textrunner being one. The utility of a natural language analyzer is highly dependent on what will be done with the results. For example, whether we can deal with an error like “water kills bacteria.”

```
Achieves precision of 88% and recall of 45% (F1 of 60%) on a large Web corpus.

Has extracted hundreds of millions of facts from a corpus of a half-billion Web pages.

E.g., even though it has no predefined medical knowledge, it has extracted over 2000 answers to what kills bacteria. Correct answers include antibiotics, ozone, chlorine, Cipro, and broccoli sprouts. Questionable answers include “water,” which came from the sentence “Boiling water for at least 10 minutes will kill bacteria.”

Source: https://openie.allenai.org/
```

How This Can Go Wrong?

As mentioned above, there are many contexts for utterances. Natural language systems may not recognize them. Humans can be good at detecting sincerity or ambiguity whereas it is often believed that natural language systems have a harder time with this.

Intention S wants to inform H that P
Generation S selects words W to express P in context C
Synthesis S utters words W

Perception H perceives W' in context C'
Analysis H infers possible meanings $P_1 \ldots P_n$
Disambiguation H infers intended meaning P_i
Incorporation H incorporates P_i into KB

How could this go wrong?

- Insincerity (S doesn't believe P)
- Ambiguous utterance
- Differing understanding of current context ($C \neq C'$)

Example API: TextRazor

TextRazor is an example of a natural language API. An example follows.

TextRazor Example: Input

```
Barclays misled shareholders and the public about one of the
biggest investments in the bank's history, a BBC Panorama
investigation has found.

The bank announced in 2008 that Manchester City owner Sheikh
Mansour had agreed to invest more than £3bn.

But

…

Neither Sheikh Mansour nor IPIC responded to questions raised
by Panorama.

In August last year, the UK's Serious Fraud Office said it had
started an investigation into commercial arrangements between
the bank and Qatar Holding LLC, part of sovereign wealth fund
Qatar Investment Authority.
```

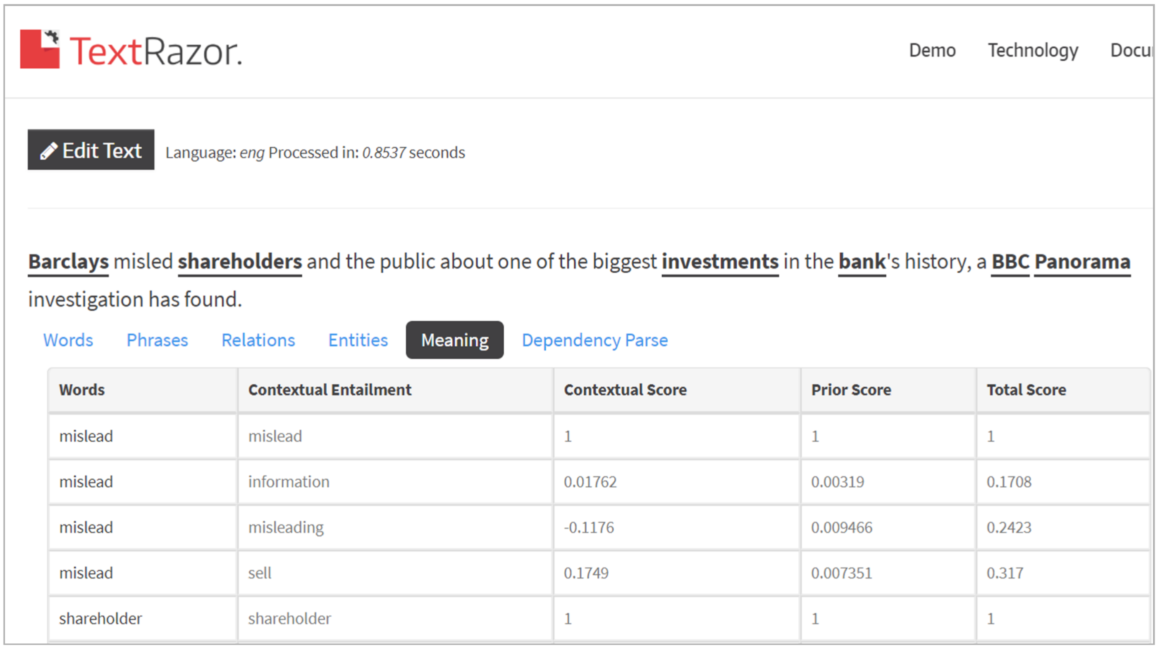
TextRazor Example Output: *Categories*

```
0.93
economy, business and finance>economy>macro
economics>investments
0.72
economy, business and finance>business
information>business finance>shareholder
0.70
economy, business and finance>economy
0.64
economy, business and finance>market
and exchange>securities
0.49
crime, law and justice>law
```

TextRazor Example Output: *Topics*

```
1.00
Barclays
1.00
Mansour bin Zayed Al Nahyan
1.00
Qatar Investment Authority
1.00
Finance
1.00
Economy
1.00
…
```

TextRazor Example Output: *Meaning*



TextRazor. Demo Technology Docu

Edit Text Language: *eng* Processed in: *0.8537* seconds

Barclays misled **shareholders** and the public about one of the biggest **investments** in the **bank**'s history, a **BBC** **Panorama** investigation has found.

Words Phrases Relations Entities **Meaning** Dependency Parse

Words	Contextual Entailment	Contextual Score	Prior Score	Total Score
mislead	mislead	1	1	1
mislead	information	0.01762	0.00319	0.1708
mislead	misleading	-0.1176	0.009466	0.2423
mislead	sell	0.1749	0.007351	0.317
shareholder	shareholder	1	1	1

Python NLTK for NL Text Processing

The natural language toolkit http://text-processing.com/demo/ can be tried out online. It poroduces sentiment analysis but also “Tokenizing”, “Stemming”, and “Tagging,” described at the site.

Logic Approaches

Before we get to machine learning approaches to NLP, we discuss one more family of approaches based on formal grammars.

We've discussed some of the characteristics of natural language that are not found in pure logic. The following list shows more of them.

Natural Language, Realistically

- ambiguity
- ***anaphora***: the use of a word referring to or replacing a word used earlier in a sentence.
- ***indexicality***: words, such as "I" or "here", that can have different meanings depending on who is saying them
- vagueness
- discourse structure
- ***metonymy***: substitution of the name of an attribute or adjunct for that of the thing meant, e.g., suit for business executive, or the track for horse racing.
- metaphor

Need *More* Than Backus Nauer Form

The rewrite rules that we've been using are known as Backus-Nauer form. But we need to go beyond this.

Backus-Nauer form notation for grammars is too restrictive:

- difficult to add "side conditions" (number agreement, etc.)
- difficult to connect syntax and semantics.

One approach is to use classical logic instead of production rules. In this case we can be more flexible. For example, instead of saying `produces X` we can say `can be interpreted as an X`. In other words, we're using predicates (of which we will say more later).

Idea: express grammar rules as logic.

$X \rightarrow Y Z$ becomes $Y(s_1) \wedge Z(s_2) \Rightarrow X(Append(s_1, s_2))$
$X \rightarrow word$ becomes $X([\text{"}\mathbf{word}\text{"}])$
$X \rightarrow Y \mid Z$ becomes $Y(s) \Rightarrow X(s) \quad Z(s) \Rightarrow X(s)$

Here, $X(s)$ means that string s **can be interpreted** as an X.

Augmented Rules using FOL

The following example says `if` s_1 `is interpreted as a noun phrase` ... But notice that we can add another implication `if` s_1 `is interpreted as <something else>`, and thus build alternative ways to process NL.

$$NP(s_1) \wedge EatsBreakfast(Ref(s_1)) \wedge VP(s_2)$$
$$\Rightarrow NP(Append(s_1, ["\ who\ "], s_2))$$

If s_1 is interpreted as a noun phrase and reference e.g., Raymond
and the result is interpreted as eating breakfast

and s_2 is interpreted as a verb phrase e.g., was eating

then s_1 who s_2 can be interpreted as a noun phrase. e.g., Raymond who was eating

Querying a Knowledge Base (KB)

Leverage automated logic processing has been quite successful in recent years.

Now it's easy to augment the rules:

$$NP(s_1) \wedge EatsBreakfast(Ref(s_1)) \wedge VP(s_2)$$
$$\Rightarrow NP(Append(s_1, ["\ who\ "], s_2))$$
$$NP(s_1) \wedge Number((s_1, n)) \wedge VP(s_2) \wedge Number(s_2, n)$$
$$\Rightarrow S(Append(s_1, s_2))$$

- Parsing is reduced to logical inference:

  Ask(*KB, S([**"I" "am" "a" "wumpus"**])*)

  (Can add extra arguments to return the parse structure, semantics)

- Generation simple requires a query with uninstantiated variables:

  Ask(*KB, S(x)*)

- If we add arguments to nonterminals to construct sentence semantics, NLP generation can be done from a given logical sentence:

  Ask(*KB, S(x, At(Robot,[1,1])*)

Neural Net Approaches

NLP via Neural Nets

It is ironic that after years of natural language analysis, a simpler but massive alternative has turned out to be remarkably successful: using machine learning and a corpus of utterances in various contexts.

"…neural network models started to be applied also to textual natural language signals, again with very promising results. … input encoding for natural language tasks, feed-forward networks,…" -- by Yoav Goldberg

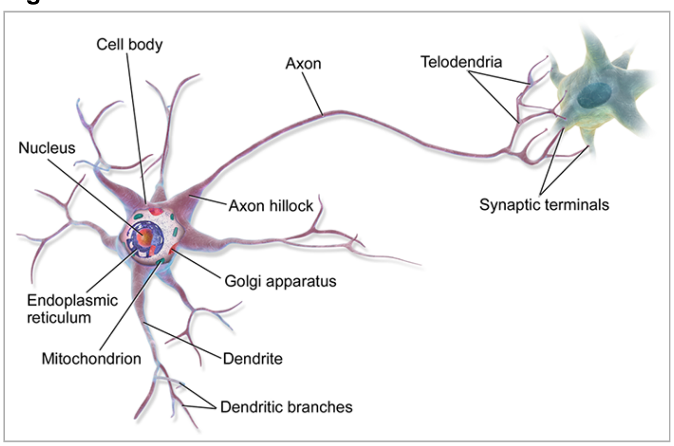
Source: https://www.jair.org/index.php/jair/article/view/11030/26198

Neural Nets: What?

Neural nets are based on aspects of the brain. A Neural net consists of *neurons*—cells that take input from and provide output to other neurons. Importantly, a single neuron does not seem to encode knowledge as we understand it. Knowledge is encoded by the set of connections *between* neurons.

A neural net approach is a problem-solving technique that simulates neurons and their interaction.

Figure: Neurons



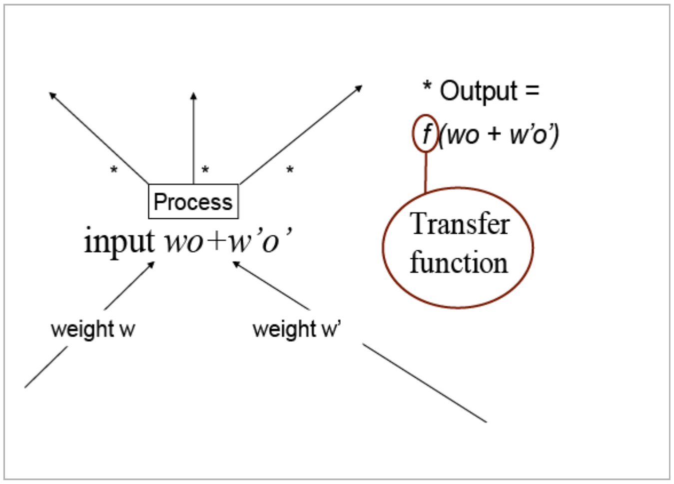
Source: https://en.wikipedia.org/wiki/Neuron

Modeling Neuronal I/O

In software, we model each connection with a number (the *weight*) that reflects its relative strength.

Each neuron in a neural net takes as input the sum of outputs of other neurons, weighted by the connection strength. It then applies a function (a *transfer* function) to this quantity. The output of this function becomes an input to other neurons (after weighting), or else it is the output to the whole neural net.

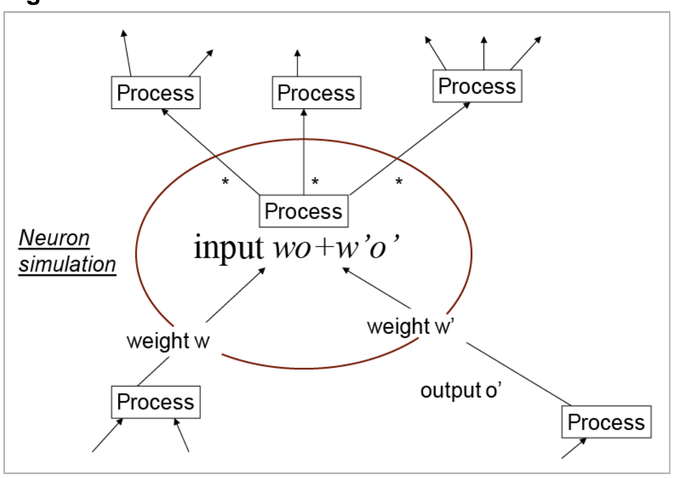
Figure: Modeling Neuronal I/O



This figure below shows how a (simulated) neuron interacts with other neurons.

As with (what we know about) biological neural networks, learning consists mainly of modifying weights.

Figure: Neuron Simulation



Google Translate Technical Architecture

One success, for example, has been **Google Translate**. This was a natural first target because there is a large corpus of input output data.

> "Neural machine translation (NMT) … machine translation that uses a large artificial neural network to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model.
>
> Deep neural machine … multiple neural network layers …"
>
> Source: https://en.m.wikipedia.org/wiki/Neural_machine_translation

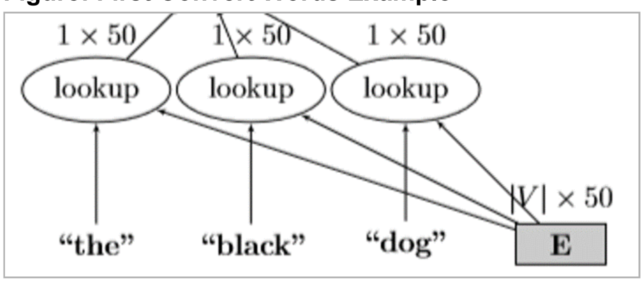
The original paper “Neural Machine Translation” by Bahdanau, Cho, and Bengio (2015) describes the use of fixed length vectors.

> “…encode a source sentence into a fixed-length vector from which a decoder generates a translation…
>
> …automatically search for parts of a source sentence that are relevant to predicting a target word…
>
> … achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. …”
>
> Source: Bahdanau, Cho, and Bengio (2015). Neural Machine Translation. https://arxiv.org/pdf/1409.0473.pdf

A Neural Net Architecture for Natural Language: First Convert Words

Neural nets deal with numbers (within vectors) so the first order of business is to convert words into numerical code.

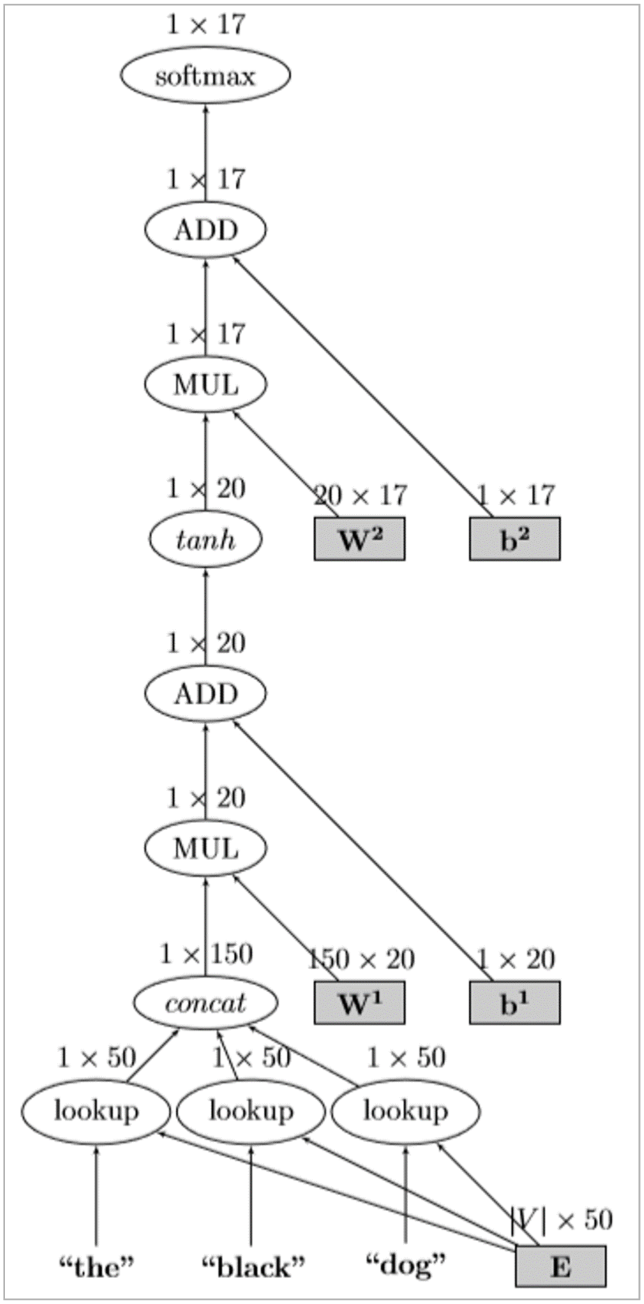
Figure: First Convert Words Example



Source: * (partial) http://www.jair.org/index.php/jair/article/view/11030

Neural Nets typically have multiple levels. The example shown has 7 hidden (neither input nor output) layers. It shows the type of activation at each level.

Figure: First Convert Words Example (Multiple-Levels)



Source: * (partial) http://www.jair.org/index.php/jair/article/view/11030

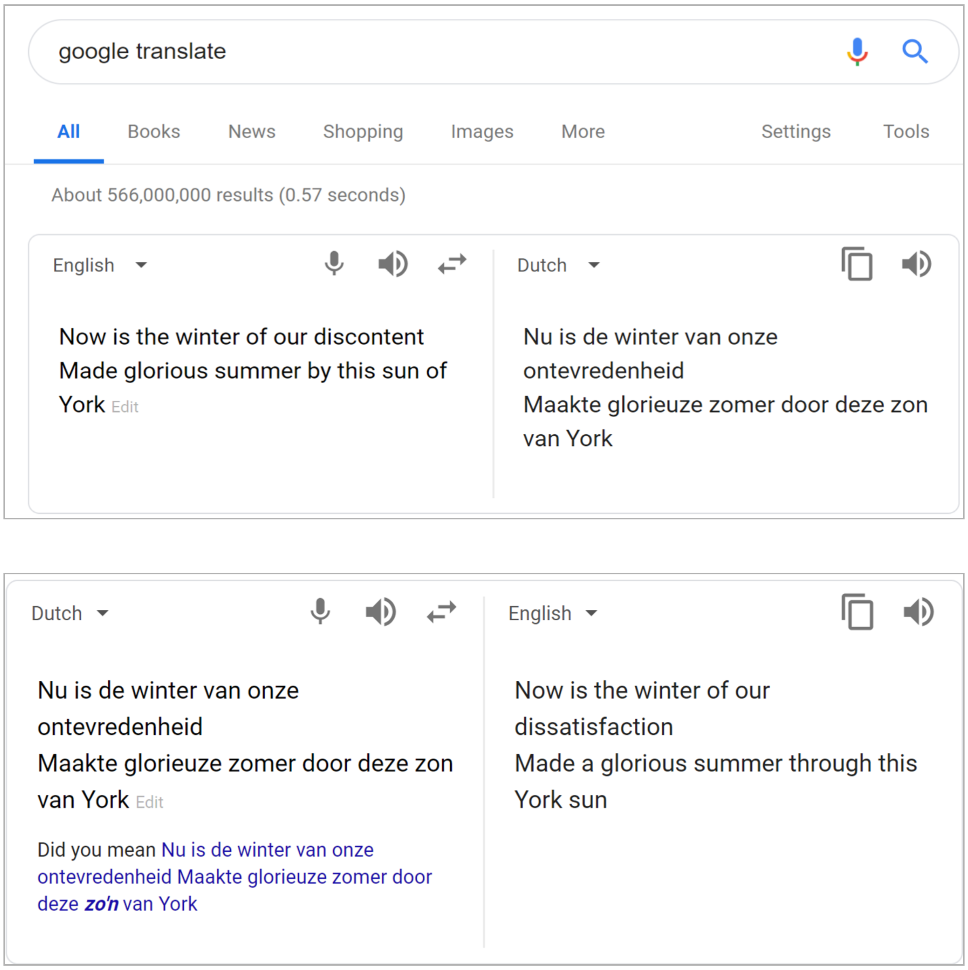
A breakthrough occurred when Google Translate was used on a passage from language A to language B and the result back to language A. Even for a nontrivial passage, the two versions were surprisingly comparable. An example is shown in the following figures.

Now is the winter of our discontent.

Made glorious summer by this sun of York.

`English → Dutch (for example) → English`

Example: Google Translate



NLP Summary

To summarize: natural language has become a major application area of AI in the real world. Although grammars continue to have a role, API's have become prevalent, and the neural net approach, leveraging existing text, is the dominant approach for the foreseeable future (e.g., see announcements from OpenAI).

- Substantial enterprise
- Grammars have role
- Tools prevalent
- Logic approaches may help
- Neural net approaches ascendant

Boston University Metropolitan College