Fundamental Concepts in Data Insight:

Natural Language Processing & Ethics

Fundamentals for a General Audience

Natural Language Processing & Ethics

- How are words represented to machines?
  - WB. How do you compare words?
- How do machines process Dostoevsky?
  - Let's get Crime and Punishment
  - Let's model a vocabulary
  - Let's find out how Dostoevsky uses words
  - Does the distributional hypothesis work?
- Natural Languages
  - What do words mean?
  - Can we learn word meanings by parsing text?
  - What can we learn from text?
  - What is association?
  - Are Human Implicit Associations expressed in Text?
  - Is association part of semantics?
- Bias
  - What is Bias?
  - WB. How do associations relate to prejudice?
  - Review: How do associations relate to prejudice?
  - Can machines detect stereotypes?
  - Where does AI bias come from?
  - Is AI racist just because we are?

This module will introduce Natural Language Processing to illustrate several conceptual issues within Machine Learning.

We are interested in the following questions,

1. How is meaningful information encoded for digital processing?
2. How is this encoding processed by inferential algorithms?
3. On what basis do these algorithms plan actions?
4. What information is omitted, or poorly represented, in this process?
5. What ethical issues arise from the use of inferential algorithms?

How are words represented to machines?

Consider the sentence, "the cat sat on the mat".

In order to draw automated inferences from this sentence (eg., "the mat has a cat sat on it"), we need to represent it for a digital computer.

That is, our goal is to *encode* this setence as a sequence of numbers for digital processing.

Let's assign a random number to each term,

```python
phrase = "the cat sat on the mat".split()

words = { word: random_number() for word in phrase }

words
```

```
{'the': 1, 'cat': 0, 'sat': 4, 'on': 3, 'mat': 5}
```

These numbers arent unique, so we wouldn't be able to use them as-is. A computer could not distinguish, say, 'cat' from 'sat'.

How should we update them? What is the *best* choice of number for each word?

Is there a way we can capture the *meaning* of a word numerically?

Consider trying to make the number choice represent how *related* the words are.

Eg., 'the' is two words *before* 'sat'.

To do this we will record a set of numbers for each word, their *relative position* to other words,

```
words
```

```
{'the': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'cat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'sat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'on': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'mat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0}}
```

Since `sat` is two before `the`, we can update manually,

```
words['the']['sat'] = 2
```

```
words
```

```
{'the': {'the': 0, 'cat': 0, 'sat': 2, 'on': 0, 'mat': 0},
 'cat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'sat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'on': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0},
 'mat': {'the': 0, 'cat': 0, 'sat': 0, 'on': 0, 'mat': 0}}
```

If we fill this table out, we know how each word is related (*by position*) to the other terms,

```
relations
```

	the	cat	sat	on	mat
the	0	1	2	3	5
cat	1	0	1	2	4
sat	2	1	0	1	3
on	3	2	1	0	2
mat	5	4	3	2	0

This table provides a means of encoding each word.

However the encoding is *relative* to a body of text (ie., the numbers will be different, given a different phrase).

The "Distributional Hypothesis" says the *relative position* of terms is a *guide to their meaning*.

> *linguistic items with similar distributions have similar meanings*
> *-- wikipedia, Distributional Semantics*

Read literally, this seems false: *that* we happen to write words together does not imply they mean the same thing.

This hypothesis has lead to the view that the best way of capturing the *meaning* of a word is to analyse its frequency and position as it occurs *in a vast library of documents*.

WB. How do you compare words?

WB. How do you compare words?

How do machines process Dostoevsky?

Let’s get Crime and Punishment

Downloading the "Project Gutenberg EBook of Crime and Punishment, by Fyodor Dostoevsky",

```
book = request.urlopen("http://www.gutenberg.org/files/2554/2554-0.
txt").read().decode('utf8')
```

And for later processing, we need the book formatted into its sentences,

```
sentences = nltk.sent_tokenize(book)
```

```
print("Sentences in book:", len(sentences))
```

```
Sentences in book: 12059
```

The 101st sentence is:

```
print(*sentences[101:102])
```

```
The bell gave a faint tinkle as though it were made of
tin and not of copper.
```

Let’s model a vocabulary

The process we performed manually above, *encoding* text, can be done automatically. There are many approaches here.

The `Word2Vec` algorithm scans the sentences of the book looking at how often words occur togther.

It *reexpresses* a word as a series of (eg.,) 100 numbers.

If we expressed a word in terms of how it related to *all possible other words*, each would be $400,000$ numbers or more. Consider the question, "how often is brick near wall?" but for every possible pair of words.

The 100-number version is an attempt at a compressed verison of this much longer set.

We can build a "vocabulary" of these 100-number "words", automatically, from "Crime and Punishment",

```
vocab = Word2Vec(sentence_words).wv
```

The word `man` corresponds to the following numbers,

```
vocab['man'].round(2)
```

```
array([ 0.05,  0.54,  0.34,  0.21,  0.23, -0.66,  0.4 ,  0.62, -0.21,
       -0.39, -0.06, -0.31,  0.23,  0.24,  0.06, -0.17,  0.37, -0.48,
       -0.1 , -1.1 ,  0.29, -0.05,  0.65, -0.04, -0.3 ,  0.16,  0.06,
       -0.15, -0.28,  0.18,  0.46, -0.42,  0.58, -0.48, -0.09,  0.31,
       -0.05, -0.14, -0.17, -0.61,  0.39, -0.42, -0.21,  0.21,  0.07,
        0.22, -0.36, -0.13,  0.08,  0.39,  0.03, -0.34, -0.03, -0.34,
       -0.24,  0.23, -0.15, -0.13, -0.49,  0.09, -0.04, -0.18,  0.02,
        0.18, -0.03,  0.57,  0.1 ,  0.48, -0.39,  0.22, -0.12,  0.18,
        0.32,  0.16,  0.42,  0.06,  0.1 , -0.07, -0.06, -0.17, -0.63,
        0.05, -0.47,  0.26, -0.4 , -0.5 ,  0.5 ,  0.29,  0.13, -0.29,
        0.39, -0.26,  0.06,  0.11,  0.2 ,  0.  ,  0.22, -0.26, -0.1 ,
       -0.11], dtype=float32)
```

Let's find out how Dostoevsky uses words

A simple mathematical operation (multiplying all numbers and totalling the result) is here taken as a guide to how similar two words are.

Consider the 100-numbers for `child` and 100-numbers for `man`, `child @ man` means multipling each number in-turn and summing the result,

```
child   = vocab['child']  / length(vocab['child'])
man     = vocab['man']    / length(vocab['man'])
weak    = vocab['weak']   / length(vocab['weak'])
strong  = vocab['strong']   / length(vocab['strong'])

child @ man, man @ weak, man @ strong, child @ weak
```

```
(0.9714216, 0.88394564, 0.9478421, 0.95429015)
```

Each number above is a guide to how similar the words are, essentially derived from *how often they occur near each other*.

Aside: the `vocab['man'] / length(vocab['man'])` operation is needed to place all the numbers on the same scale, so they can be compared.

```
vocab.most_similar('man')
```

```
[('by', 0.9796208143234253),
 ('little', 0.9786400198936462),
 ('who', 0.9785995483398438),
 ('wood', 0.9781473278999329),
 ('expression', 0.9781315922737122),
 ('fever', 0.9776231646537781),
 ('time', 0.9774249196052551),
 ('full', 0.977258026599884),
 ('woman', 0.977125346660614),
 ('other', 0.9770156145095825)]
```

```
vocab.most_similar('girl')
```

```
[('new', 0.9991231560707092),
 ('A', 0.9988678693771362),
 ('drunken', 0.9988556504249573),
 ('later', 0.9986962676048279),
 ('between', 0.998674750328064),
 ('young', 0.9986497759819031),
 ('letter', 0.9986132979393005),
 ('cold', 0.9985955953598022),
 ('second', 0.9985937476158142),
 ('idea', 0.9985917806625366)]
```

Does the distributional hypothesis work?

The claim of adhearents is that *with enough text* we are sure to capture what words mean.

There are major problems with this idea, not least that (1) the scale of text needed runs into trillions of examples; and (2) no matter how texts *contain* words their meaning is determined by *use*.

The latter is a *fatal* problem: the co-occurance of words in some text **isn't** what they *mean*.

Natural Languages

What do words mean?

Meaning is use -- Wittigenstein

Words are like *tools* (eg., a hammer). Their meaning is how we *use them like tools.*

Eg., two people are talking, one says "pass me the salt" and the other hands them the salt.

The *tool* "salt" comes to *refer to* **the salt in our shared environment** in the context of this communication.

The communicative actions between the two speakers is *primary*.

We are *first* acting within shared enviroments, and second, we use words to *help us do this.*

Aside: it is not clear if "salt" has any meaning outside of a communicative context; or if it does, this very general sort of meaning is merely *the many potential specific **uses** it could acquire.*

Can we learn word meanings by parsing text?

Since machines do not use words (as above, like tools), they do not *mean* anything when they generate text.

Consider a machine generating the text "pass me the salt" here it cannot mean what *I* say when *I* say, "pass me the salt": I want the salt. The machine isn't even aware of salt, nor has ever encountered any (etc.).

Since the machine isn't *with me*, it isnt *talking to me*.

Suppose I input, "I like salt on chips" and it replies, "I like vinegar, try vinegar too!". Approximately, it replies *because* vinegar *is associated with salt*, and that *try .. too!* is associated with *like*.

What's missing here is that the machine has never tried vinegar, has no knoweldge *of vinegar*, and only knowledge of *the term vinegar*: ie., what other terms co-occur in historical texts.

So it cannot possibly *use* the phrase "I like vinegar". And if it not *using* it, it is not *meaning* it.

Aside: consider also a non-english speaker being instructed to say "Eye LieKuh Vee Nee Guar", do they? Do those *sounds* **mean** anything? Consider a cave through which a wind blows similar sounds: is it speaking?

What can we learn from text?

You can determine *statistical term associations* from text, eg., that "King" is associated with "Queen".

Since statistical AI can only parse text, it is *asserted* that meaning *is* asssocation (distributional hypothesis).

However we need to be extremely warey of this claim.

What is association?

Association is how likely it is two *terms* are *related*. A statistical association is evidencing that association by co-occurance in some body of text.

For example, I might *associate* "candy" with "horror" if I only eat sweets when watching horror films. A statistical anaysis of my chat history *might* capture this; it might not.

We can define statistical association as a probability.

Two terms are associated if, in any sequence of words,

- $P(CurrentTerm = A|Previous = B) > P(CurrentTerm = A)$
- $P(Term = King|Previous = Queen) > P(King|Previous = Anything)$

- Generically,
- for a set of options $O_1, O_2, \ldots,$
  - $P(A|O_1) > P(A|O_1)$ says O_1 is more associated with A than O_2

Human Implicit Associations are expressed in Text Associations

An *implicit association* is any association we make which isn't explicitly expressed in our understanding of what we believe about a topic or ourselves.

The implicit-association test (IAT) is a controversial assessment in the field of social psychology intended to detect the strength of a person's subconscious association between mental representations of objects (concepts) in memory. It is commonly applied to assess implicit stereotypes held by test subjects, such as unconsciously associating stereotypically black names with words consistent with black stereotypes.[1] The test's format is highly versatile, and has been used to investigate biases in racial groups, gender, sexuality, age, and religion, as well as assessing self-esteem.

Eg., $P(\text{Person is Powerful}|Male) > P(\text{Person is Powerful}|Female)$

The IAT test is, roughly, a system which measures how quickly we respond to words being shown on screen with each other. The central claim of this test is that systematic timing differences reveal associations.

Whether this is true or not, it is not in doubt that people associate terms.

Is association part of semantics?

Some[1] natural language processing adhearents believe the distributional hypothesis, and *in addition*, a hypothesis about association,

1. The meaning of a term is the terms it's associated with
2. *Association* is statistical association
3. Statistical association can be determined by scanning a representative amount of text

([1] It tends not to be the most senior practicioners in the area, but junior practicioners or people *selling* the technology.)

There are major problems with this.

One : that *how* we associate ideas is not expressed in the way the terms are distributed in text. We may believe "X" and "Y" are connected by *similarity*; and "A" and "B" by *cultural contingency* but the cooccurance of X-Y and A-B may be identical.

Two: If we claim that *association is meaning* then we build incidental stereotypes into the meaning of words themselves.

People associate terms of all kinds of private incidental reasons and are aware that these associations do not bare on the meaning of the term ("candy" does not mean "horror"). Worse statistical term assocaitions do not capture even *our* associations.

If association is meaning then it is *literally* the case that "Programmer" means "a man who...". As many associate "Programmer" with "Man", and even if these associations change **much historical text** continues to do so!

Programmer is a well-defined term which can be used to *communicate successfully* in many contexts where both women and men are being considered.

Note that it is because *meaning isn't association*, we are able to correct and identify misunderstanding.

Bias

Since NLP systems work on statistical associations (as, essentially does every ML technique), we are in great danger of encoding non-semantic sterotypical assumptions into algorithmic decision-making.

What is Bias?

- statistical bias
    - systematic inaccuracy
- biased decision/treatment
    - unfair treatment
- subjective bias
    - stake in the outcome
- biased process
    - process with unfair concequences

WB. How do associations relate to prejudice?

Review: How do associations relate to prejudice?

- prior associations,
  - expectations derived from experience *of* regularities
  - eg., $P(\text{Programmer is Male}) > P(\text{Programmer is Female})$
  - thef. "Programmer" is associated with "Male"
- stereotype
  - we believe something **normative** *about* an association (which are circumstantial/cultural)
    - eg., "Popular = Good"
    - eg., "Common = Safe, Normal, ..."
  - so we take the more associated option as being *good*
    - a sterotype is a immoral conclusion of this kind
    - an archetype is a morally neutral example
  - eg., "Since programmers are mostly male, being male is characteristic of being a good programmer"
- prejudical action
  - making decisions based on sterotypes
  - eg., "Since programmers are mostly male, being male is characteristic of being a good programmer"
    - **therefore** hiring only *male* programmers

A *prior association* is evidence for beliefs about the types of things being associated. It is an ethical problem when we form *normative* beliefs which lead to prejudicial actions.

Can machines detect stereotypes?

No.

- Machnines have no access to natural word semantics, ie., meaning
  - they can only observe association
- $P(Prg = Male|Text) > P(Prg = Female|Text)$ **is true**
  - the machine cannot distinguish this from $P(Bee = Insect|Text) > P(Bee = Fish|Text)$
- NLP systems only have access to this information, so they learn both:
  - Bees are insects
  - Programmers are male

Where does AI bias come from?

- inherent to algorithm
    - misattributing association for meaning
- human labelling (via culture)
- system design (ignorance of developer)
- delibate (intention of developer)
- and more..

Is AI racist just because we are?

Consider processing a book of policy which advocates for improving the circumstances of a minority. The machine will likely *statistically associate* ethnic terms with *negative words* precisley because the book is *about* that association. However the book is a *refutation* of this association, not an endorsement.

The issue here isn't the racism of the authors, or the malice or racism of the culture. The issue is that the distributional hypothesis is false. The machine has no access to the meaning of what's said, so it reporduces racism *where none existed*.

This isn't "machines reporducing the bias and racism of culture". It is machines creating *novel racism* where none existed prior.

It is often claimed by proponents of this technology that the problem is "in people", and the machines are merely passive observes of the truth of *what we mean when we talk to one another*. However this must be wholey rejected if we are to anticipate all the ethical concequences of these systems.