QLD AI Foundations - NLP Fundamentals

otso.ai

Who is otso

otso is a machine learning company that specialises in the analysis of unstructured text data using state of the art natural language processing and artificial intelligence technology.

otso supports a range of use cases including:

- Voice of the Customer
- Media Monitoring
- Event Management
- QA / QC
- Survey Coding
- Claim Automation

otso.ai

Overview

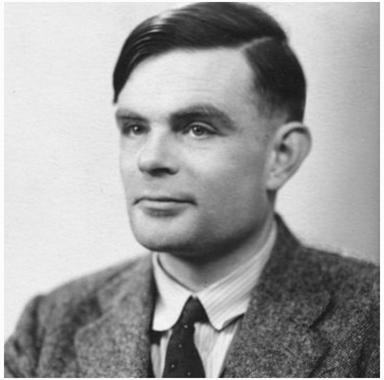
otso.ai

1.0 NLP Overview

1.1 Historical NLP

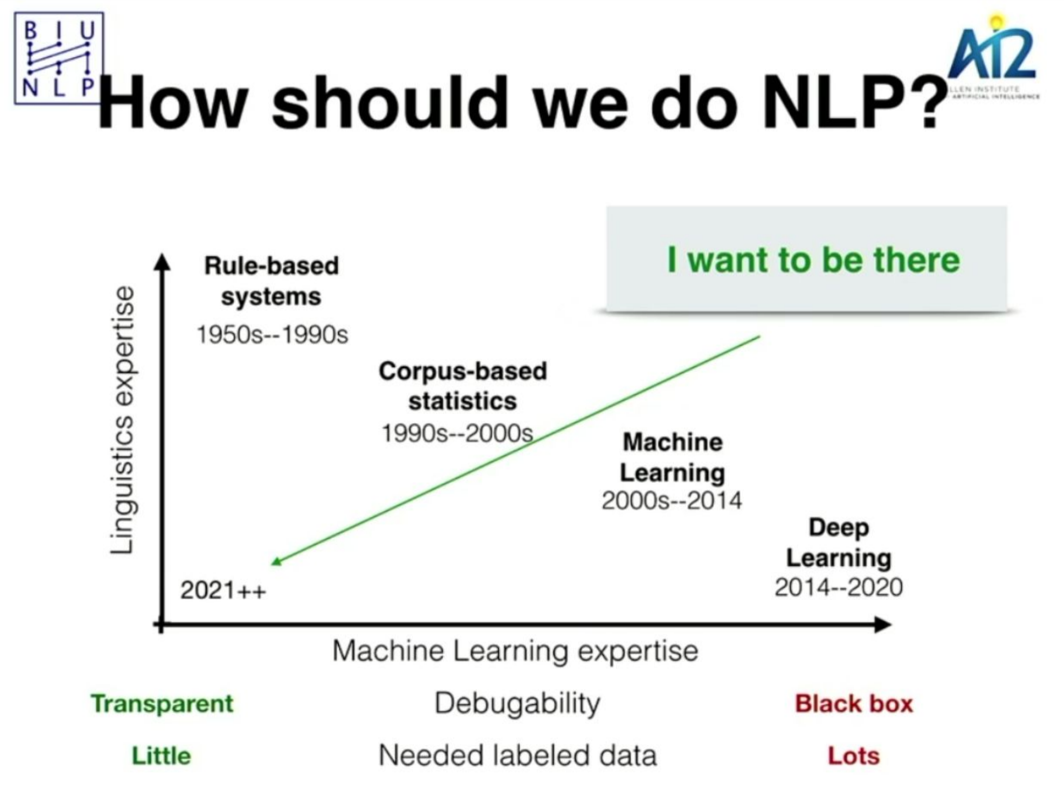
- **Abraham Anulafia (13th century).** Pioneered the "science of combining letters", using divine rules derived from scripture [1]
- **Leibniz (17th century).** German polymath Gottfried Leibniz outlined a theory for automating knowledge production using "thoughts" as an atomic unit manipulated by rules [1]
- **Markov and Shannon (1913).** Markov's work around applying probability to text; previous utterances influence future utterances. Shannon via Markov demonstrated that increasing complexity of probability models improved the comprehensibility of output, an echo of contemporary language models [1]
- **Turing Test (1950).** "A computer would deserve to be called intelligent if it could deceive a human into believing that it was human", via text-based exchanges.

Letter combinatorics was an area of study in the 13th century. Abraham Abulafia pictured.



The turing test, a somewhat dated approach to gauging machine intelligence, uses text-based interaction

1.2 Contemporary NLP Techniques



Courtesy of Yoav Goldberg’s *The missing Elements in NLP* talk, Spacy IRL 2019 [2]

1.3 NLP and Linguistics

- Both fields make use of formal training in CS, linguistics and machine learning. Terms are often interchangeable

- **NLP** is Generally more engineering focused, emphasis upon helping people navigate and digest large quantities of information that already exist in text form. Something more than just “commercial text processing” though; NLP finds use within political science, economics, biology, medicine and (digital) humanities [3]

- **Linguistics** is the study of languages and how they function, typically using corpora and computers (in the case of computational linguistics) [3]

1.4 Current Areas of Work

Technical Areas [4]

- **Parsing problems.** Constituency, dependency etc.
- **Signal-orientated problems.** Speech recognition, machine translation.
- **Information extraction problems.** Named entity recognition, coreference resolution, entity linking, POS tagging.
- **Document classification.** Sentiment, arbitrary classification.
- **Information retrieval.** Search engines, recommendation systems.
- **“BERTology”.** Work based upon transformer models.

Qualitative Issues [5] [6]

- **Bias.** Especially important with the widespread use of embeddings/language models; which have the effect of “injecting” source bias into downstream models
- **Transparency.** RE: rules-based systems v Blackbox algorithms.
- **Non-English NLP.** Related to issues of Bias. Algorithmic compatibility (RE: English algorithms “breaking” when applied to other languages). Low resource languages etc.

1.5 Interesting Applications of NLP

- **Legal NLP.** Blackstone spacy variation, for processing long-form, unstructured legal text [7]
- **Biological sciences.** Using tools like SciSpacy, a custom tokenizer/NER/abbreviation resoltuion models designed for use on biomedical text [8]
- **Semantic Scholar.** Advanced search engine for academic papers, by AI2 [9]
- **Redaction and PII removal.** Removing references to people, places etc. via NER

2.0 NLP Concepts

otso.ai

2.1 Tokenisation

- Given a sequence of text, segment the text into smaller pieces (tokens), in preparation for later processing
- A **token** is an instance of a sequence of characters in some particular document that are grouped together as a useful semantic unit for processing (single words, punctuation etc.) [10]
- Can ignore certain tokens, depending upon the tokenizer (punctuation? non-utf8 characters?)
- **Sentence boundary detection** performed here as well

'otso is a machine learning company that specialises in the analysis of unstructured text data using state of the art natural language processing and artificial intelligence technology.'

	text	start_char	end_char	is_digit	is_punct
0	otso	0	4	False	False
1	is	5	7	False	False
2	a	8	9	False	False
3	machine	10	17	False	False
4	learning	18	26	False	False
5	company	27	34	False	False
6	that	35	39	False	False
7	specialises	40	51	False	False
8	in	52	54	False	False
9	the	55	58	False	False
10	analysis	59	67	False	False
11	of	68	70	False	False
12	unstructured	71	83	False	False
13	text	84	88	False	False
14	data	89	93	False	False
15	using	94	99	False	False
16	state	100	105	False	False
17	of	106	108	False	False
18	the	109	112	False	False
19	art	113	116	False	False
20	natural	117	124	False	False
21	language	125	133	False	False
22	processing	134	144	False	False
23	and	145	148	False	False
24	artificial	149	159	False	False
25	intelligence	160	172	False	False
26	technology	173	183	False	False
27	.	183	184	False	True

2.2 Part of Speech Tags (POS)

- Assign part of speech tag to each token, based upon its relationship with adjacent and related words in a larger sequence (phrase, sentence, paragraph) [11]
- Rule-based and statistical variants exist
- **Morphology:** inflectional morphology is the process by which a root form of a word is modified by adding prefixes or suffixes that specify its grammatical function but do not change its part-of-speech [12]
- A **lemma** (eg. the word "run") is **inflected** with some **morphological features** (eg. present tense, past tense) to create some **surface** variation (eg. the word "running")

POS	DESCRIPTION	EXAMPLES
ADJ	adjective	big, old, green, incomprehensible, first
ADP	adposition	in, to, during
ADV	adverb	very, tomorrow, down, where, there
AUX	auxiliary	is, has (done), will (do), should (do)
CONJ	conjunction	and, or, but
CCONJ	coordinating conjunction	and, or, but

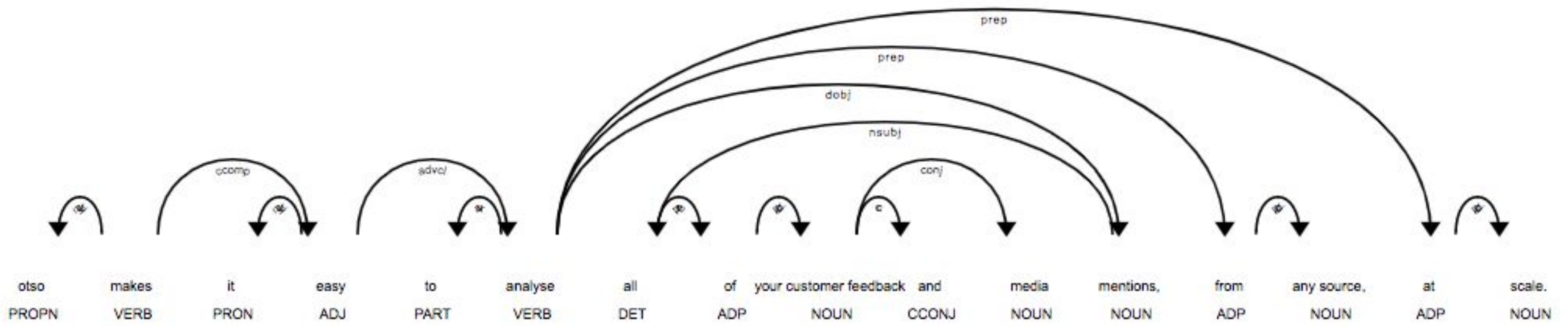
	text	pos	lemma	embedding_sentence
0	otso	PROPN	otso	(otso, is, a, machine, learning, company, that...
1	is	AUX	be	(otso, is, a, machine, learning, company, that...
2	a	DET	a	(otso, is, a, machine, learning, company, that...
3	machine	NOUN	machine	(otso, is, a, machine, learning, company, that...
4	learning	VERB	learn	(otso, is, a, machine, learning, company, that...
5	company	NOUN	company	(otso, is, a, machine, learning, company, that...
6	that	DET	that	(otso, is, a, machine, learning, company, that...
7	specialises	VERB	specialise	(otso, is, a, machine, learning, company, that...
8	in	ADP	in	(otso, is, a, machine, learning, company, that...
9	the	DET	the	(otso, is, a, machine, learning, company, that...

2.3 Dependency Parsing

- Recognizing a linguistic sequence, and assigning syntactic structure to each component within it [13]
- Resolves the structural ambiguity between words within a sequence, in a formal way [13]
- Typically results in a dependency **parse tree**, reflective of the hierarchical tree structure found in most texts
- Allows us to uncover subjects, objects, their attributes, root aspects, the children of these **root aspects** etc.
- Can be used to derive **noun chunks**; flat phrases featuring nouns as their head (eg. some noun, plus the words describing it)

	noun_chunk	root	root_lemma
0	otso	otso	otso
1	a machine learning company	company	company
2	the analysis	analysis	analysis
3	unstructured text data	data	datum
4	state	state	state
5	the art natural language processing	processing	processing
6	artificial intelligence technology	technology	technology

2.3 Dependency Parsing



Using spacys' *displacy* visualiser we can inspect the results of the dependency parser. [14]

2.4 Named Entity Recognition

- Utilizes the token attributes assigned during the POS and Dependency parsing phase (eg. get all Noun's which are the syntactic root of a tree, assign some label x?)

- Seeks to assign candidate entities into predefined categories (PERSON, GPE, ORG etc.) [15]

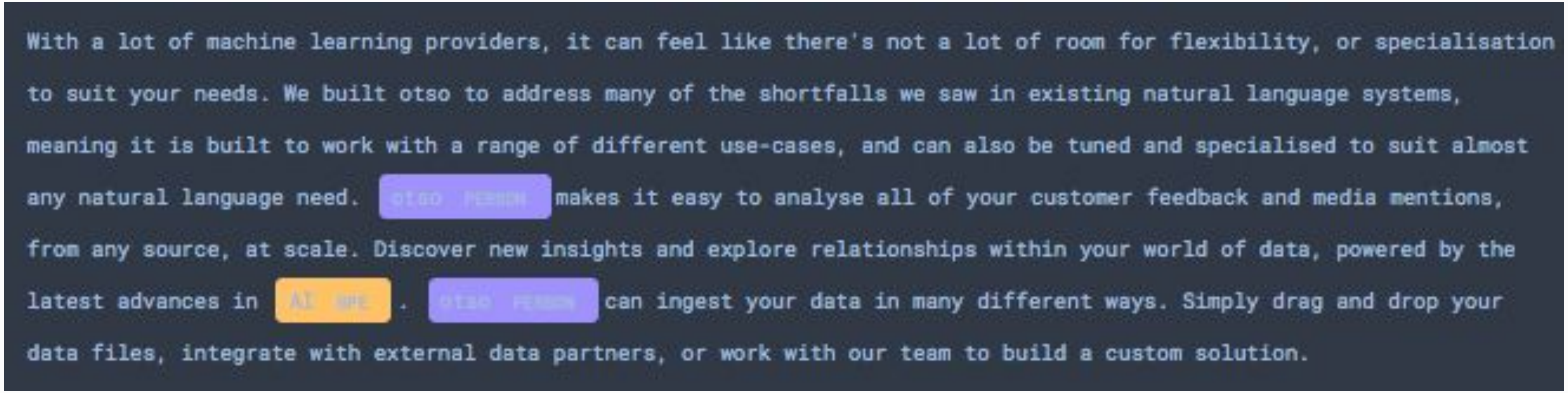
- For our purposes, entities consist of **spans**, multiple tokens [15]

- Can recombine in powerful ways using other spacy attributes; extract the embedding sentence for some specific entity types? Filter noun-chunks based upon the root being some specific entity type? Tabulate and aggregate entire entity categories?

	embedding_sentence	entity	entity_label	entity_lemma	entity_pos	start	end
0	otso makes it easy to analyse all of your cust...	otso	PERSON	otso	PROPN	0	4
1	Discover new insights and explore relationship...	AI	GPE	ai	PROPN	109	111
2	otso can ingest your data in many different ways.	otso	PERSON	otso	PROPN	0	4

otso.ai

2.4 Named Entity Recognition

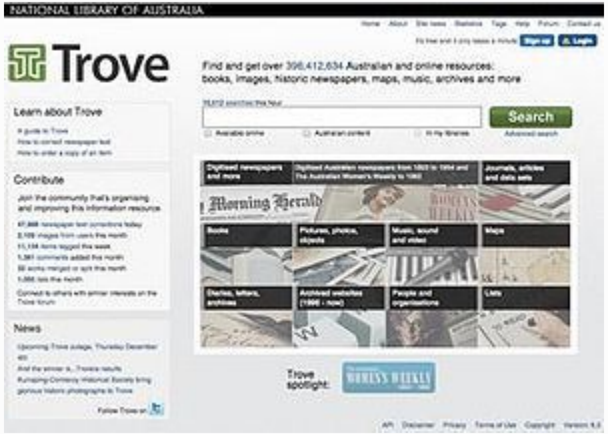


Using spacys' *displacy* visualiser we can inspect the results of the NER model. [14]

3.0 Building a Dataset

3.1 Trove

- An Australian library database aggregator, hosted by the National Library of Australia [16]
- Focusing upon Australian content
- Covering digitised newspapers, government gazettes, journal articles, books, pictures, maps, diaries/letters etc.
- Incredible depth and breadth of resources, dating back to colonial times
- Decent API, readily provides API keys



3.2 Alternatives

Notable mentions include:

- **Auslii.** Australiasian Legal Information Institute, caselaw and legislation repository. Public policy initiative operated by UTS and UNSW to improve access to justice via legal information [17]
- **Project Gutenberg.** 60,000 public domain (generally older) ebooks. Intended to digitize and archive cultural works [18]
- **Pandora Archive.** Australian Web content archive [19]
- **Seinfeld scripts.** Scripts for all 180 episodes.
- **Simpsons scripts.** Scripts for episodes 1-31.
- **Guardian Open Platform.** Generous and well designed API allowing access to Guardian content [20]

4.0 Analysing NLP Outputs

4.1 From Unstructured to Structured

- Once we've parsed text as a spacy document, we can "tabulate" the objects within it, goal being to use common relational DB techniques

- **Entity tables.** For each entity in a document, extract the entity text, entity lemma, token offsets, embedding sentence and other useful attributes as records that can be formatted as a table

- **Noun-chunk tables.** Equally, for each noun-chunk within a spacy document, extract the noun chunk text, root lemma and other useful attributes as records that can be formatted as a table

- Other variants like subject-verb-object triples also exist

4.2 Future Improvements

- **OCR Accuracy.** To its credit, the “first pass” with the OCR software is entirely automated (scales well), and is pretty decent in terms of accuracy. Trove claims to aim for 98% accuracy, though also acknowledges the variability in translation.
- **Model Alignment/inaccuracies.** Spacy model “pre-trained” with GLOVE embeddings and fine-tuned on academic tasks like the CONNL NER dataset or Universal Dependencies. This is quite different to the language and structure of the trove texts. Entity model needs some work in particular.
- **Domain knowledge.** Would be nice to give these types of tools to historians instead.
- **Corpus-specific processing.** Would be be nice to apply corpus-level instead of document-level analysis, perhaps Latent Dirichlet Allocation (LDA) or Singular Value Decomposition (SVD)
- **Distributed processing.** Definitely hitting the upper limit of what a well-resourced, single instance can process. Good news is that document-based NLP is embarrassingly parallel and the processing we’ve applied lends itself well to parallel methods.

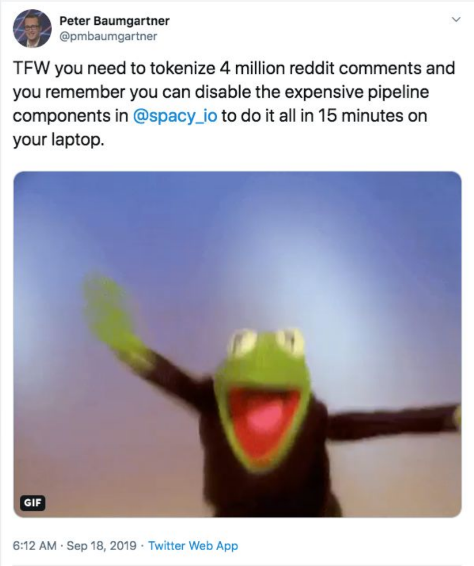
5.0 Scaling NLP Analysis

5.1 Defining Scale

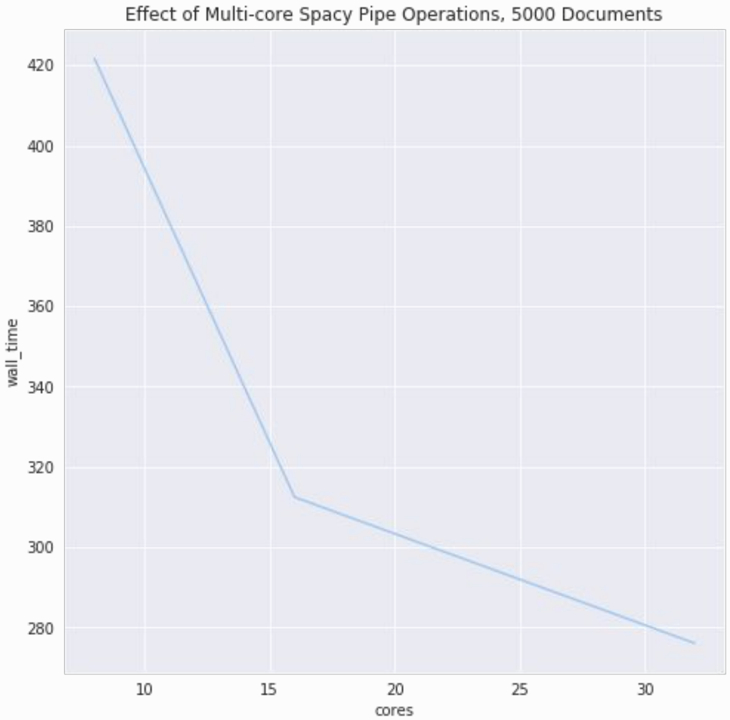
- "Scale" means different things to different organisations and people
- Thousands of documents? Tens of thousands? Hundreds of thousands? Of what size? A few sentences? A few paragraphs?
- Difficult to quantify the size of "documents" in this sense, unit of measurement is usually at the token level instead (see spacy's benchmarks for details)
- As an aside, what can be parallelized can usually (always?) be done serially with more time. Consider the time needed to configure a parallel solution as opposed to "just running" a serial one

5.2 Specific Tips

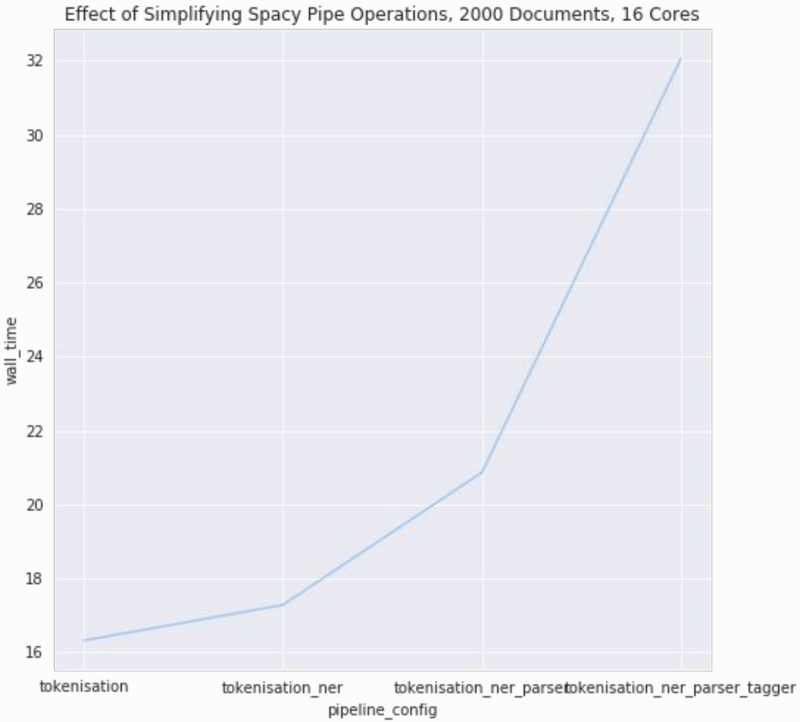
- **Prototype your analysis on a smaller, representative sample of the data.** Ensure your pipeline works before you scale it. Scaling computation necessarily widens the feedback loop; errors become costly etc.
- **Move all the expensive compute "up" the pipeline, perform once.** Ideally the computation is performed once, and then analysed many times. As is the case in NLP, a single spacy document can be analysed many different ways.
- **Use spacy's pipe method to batch documents across multiple cores.** Native multi-processing (<=v2.1 and re-introduced in v2.2.2).
- **Disable "expensive" spacy pipeline components if they're not needed.** Eg. remove dependency parser if it's not needed



5.3 Some Rough Benchmarking



Multi-processing matters, use spacy's pipe method where possible.



6.0 Additional Resources

otso.ai

6.1 Additional Resources

- **Code.** From tonight's talk, featuring a downsampled version of the main trove corpus, can be found at https://github.com/samhardyhey/qld-ai-nlp-dev
- **Spacy IRL.** Conference based around spacy. Lots of talks around practical NLP issues, all recorded and available on youtube.
- **"NLP Twitter".** Bit of a rabbit hole. Excellent way to stay on top of latest developments though (RE: latest research).
- **Linguistic Fundamental for NLP: 100 Essentials (Bender, 2019?).** Solid reading if you're coming from a programming background and want to learn about linguistic fundamentals.
- **Neural Network Methods for Natural Language Processing (Goldberg, 2017).** What it says on the tin; NN as specifically applied to NLP. Dated from 2017, but still extremely relevant in the "age of the transformer" IMO

otso.ai

Thanks!

References

[1] https://spectrum.ieee.org/tech-talk/robotics/artificial-intelligence/natural-language-processing-dates-back-to-kabbalist-mystics
[2] https://www.youtube.com/watch?v=e12danHhlic](https://www.youtube.com/watch?v=e12danHhlic
[3] https://linguistics.stackexchange.com/questions/1802/what-are-the-fundamental-differences-between-natural-language-processing-and-com
[4] https://github.com/sebastianruder/NLP-progress](https://github.com/sebastianruder/NLP-progress)
[5] An analysis of gender bias studies in natural language processing, Marta R. Costa-jussà
[6] Ethical by Design: Ethics Best Practices for Natural Language Processing, Jochen L. Leidner and Vassilis Plachouras
[7] https://github.com/ICLRandD/Blackstone
[8] https://github.com/allenai/scispacy
[9] https://www.semanticscholar.org/
[10] https://nlp.stanford.edu/IR-book/html/htmledition/tokenization-1.html
[11] https://www.nltk.org/book/ch05.html
[12] http://www.linguisticsnetwork.com/affixation-in-english/
[13] https://www.sciencedirect.com/topics/computer-science/syntactic-structure
[14] https://spacy.io/usage/visualizers
[15] https://spacy.io/usage/linguistic-features
[16] https://trove.nla.gov.au/
[17] http://www.austlii.edu.au/
[18] https://www.gutenberg.org/
[19] https://pandora.nla.gov.au/
[20] https://open-platform.theguardian.com/documentation/