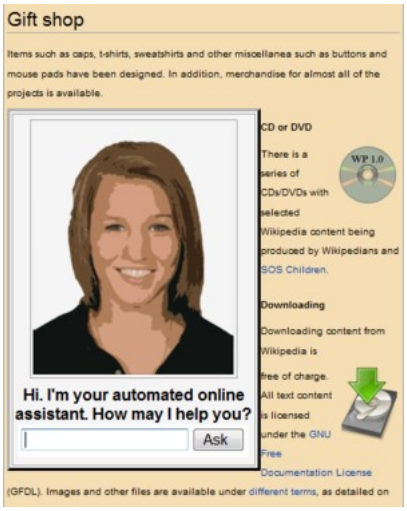
Natural language processing

Natural language processing (**NLP**) is a subfield of computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data.

Challenges in natural language processing frequently involve speech recognition, natural language understanding, and natural language generation.



An automated online assistant providing customer service on a web page, an example of an application where natural language processing is a major component.[1]

Contents

History

The history of natural language processing generally started in the 1950s, although work can be found from earlier periods. In 1950, Alan Turing published an article titled "Intelligence" which proposed what is now called the Turing test as a criterion of intelligence.

The Georgetown experiment in 1954 involved fully automatic translation of more than sixty Russian sentences into English. The authors claimed that within three or five years, machine translation would be a solved problem.[2] However, real progress was much slower, and after the ALPAC report in 1966, which found that ten-year-long research had failed to fulfill the expectations, funding for machine translation was dramatically reduced. Little further research in machine translation was conducted until the late 1980s, when the first statistical machine translation systems were developed.

Some notably successful natural language processing systems developed in the 1960s were SHRDLU, a natural language system working in restricted "blocks worlds" with restricted vocabularies, and ELIZA, a simulation of a Rogerian psychotherapist, written by Joseph Weizenbaum between 1964 and 1966. Using almost no information about human thought or emotion, ELIZA sometimes provided a startlingly human-like interaction. When the "patient" exceeded the very small knowledge base, ELIZA might provide a generic response, for example, responding to "My head hurts" with "Why do you say your head hurts?".

During the 1970s, many programmers began to write "conceptual ontologies", which structured real-world information into computer-understandable data. Examples are MARGIE (Schank, 1975), SAM (Cullingford, 1978), PAM (Wilensky, 1978), TaleSpin (Meehan, 1976), QUALM (Lehnert, 1977), Politics (Carbonell, 1979), and Plot Units (Lehnert 1981). During this time, many chatterbots were written including PARRY, Racter, and Jabberwacky.

Up to the 1980s, most natural language processing systems were based on complex sets of hand-written rules. Starting in the late 1980s, however, there was a revolution in natural language processing with the introduction of machine learning algorithms for language processing. This was due to both the steady increase in computational power (see Moore's law) and the gradual lessening

of the dominance of Chomskyan theories of linguistics (e.g. transformational grammar), whose theoretical underpinnings discouraged the sort of corpus linguistics that underlies the machine-learning approach to language processing.[3] Some of the earliest-used machine learning algorithms, such as decision trees, produced systems of hard if-then rules similar to existing hand-written rules. However, part-of-speech tagging introduced the use of hidden Markov models to natural language processing, and increasingly, research has focused on statistical models, which make soft, probabilistic decisions based on attaching real-valued weights to the features making up the input data. The cache language models upon which many speech recognition systems now rely are examples of such statistical models. Such models are generally more robust when given unfamiliar input, especially input that contains errors (as is very common for real-world data), and produce more reliable results when integrated into a larger system comprising multiple subtasks.

Many of the notable early successes occurred in the field of machine translation, due especially to work at IBM Research, where successively more complicated statistical models were developed. These systems were able to take advantage of existing multilingual textual corpora that had been produced by the Parliament of Canada and the European Union as a result of laws calling for the translation of all governmental proceedings into all official languages of the corresponding systems of government. However, most other systems depended on corpora specifically developed for the tasks implemented by these systems, which was (and often continues to be) a major limitation in the success of these systems. As a result, a great deal of research has gone into methods of more effectively learning from limited amounts of data.

Recent research has increasingly focused on unsupervised and semi-supervised learning algorithms. Such algorithms are able to learn from data that has not been hand-annotated with the desired answers, or using a combination of annotated and non-annotated data. Generally, this task is much more difficult than supervised learning, and typically produces less accurate results for a given amount of input data. However, there is an enormous amount of non-annotated data available (including, among other things, the entire content of the World Wide Web), which can often make up for the inferior results if the algorithm used has a low enough time complexity to be practical.

In the 2010s, representation learning and deep neural network-style machine learning methods became widespread in natural language processing, due in part to a flurry of results showing that such techniques[4][5] can achieve state-of-the-art results in many natural language tasks, for example in language modeling,[6] parsing,[7][8] and many others. Popular techniques include the use of word embeddings to capture semantic properties of words, and an increase in end-to-end learning of a higher-level task (e.g., question answering) instead of relying on a pipeline of separate intermediate tasks (e.g., part-of-speech tagging and dependency parsing). In some areas, this shift has entailed substantial changes in how NLP systems are designed, such that deep neural network-based approaches may be viewed as a new paradigm distinct from statistical natural language processing. For instance, the term *neural machine translation* (NMT) emphasizes the fact that deep learning-based approaches to machine translation directly learn sequence-to-sequence transformations, obviating the need for intermediate steps such as word alignment and language modeling that were used in statistical machine translation (SMT).

Rule-based vs. statistical NLP

In the early days, many language-processing systems were designed by hand-coding a set of rules,[9][10], e.g. by writing grammars or devising heuristic rules for stemming. However, this is rarely robust to natural language variation.

Since the so-called "statistical revolution"[11][12] in the late 1980s and mid 1990s, much natural language processing research has relied heavily on machine learning.

The machine-learning paradigm calls instead for using statistical inference to automatically learn such rules through the analysis of large *corpora* of typical real-world examples (a *corpus* (plural, "corpora") is a set of documents, possibly with human or computer annotations).

Many different classes of machine-learning algorithms have been applied to natural-language-processing tasks. These algorithms take as input a large set of "features" that are generated from the input data. Some of the earliest-used algorithms, such as decision trees, produced systems of hard if-then rules similar to the systems of hand-written rules that were then common. Increasingly, however, research has focused on statistical models, which make soft, probabilistic decisions based on attaching real-valued weights to each input feature. Such models have the advantage that they can express the relative certainty of many different possible answers rather than only one, producing more reliable results when such a model is included as a component of

a larger system.

Systems based on machine-learning algorithms have many advantages over hand-produced rules:

- The learning procedures used during machine learning automatically focus on the most common cases, whereas when writing rules by hand it is often not at all obvious where the effort should be directed.
- Automatic learning procedures can make use of statistical-inference algorithms to produce models that are robust to unfamiliar input (e.g. containing words or structures that have not been seen before) and to erroneous input (e.g. with misspelled words or words accidentally omitted). Generally, handling such input gracefully with hand-written rules—or, more generally, creating systems of hand-written rules that make soft decisions—is extremely difficult, error-prone and time-consuming.
- Systems based on automatically learning the rules can be made more accurate simply by supplying more input data. However, systems based on hand-written rules can only be made more accurate by increasing the complexity of the rules, which is a much more difficult task. In particular, there is a limit to the complexity of systems based on hand-crafted rules, beyond which the systems become more and more unmanageable. However, creating more data to input to machine-learning systems simply requires a corresponding increase in the number of man-hours worked, generally without significant increases in the complexity of the annotation process.

Major evaluations and tasks

The following is a list of some of the most commonly researched tasks in natural language processing. Note that some of these tasks have direct real-world applications, while others more commonly serve as subtasks that are used to aid in solving larger tasks.

Though natural language processing tasks are closely intertwined, they are frequently subdivided into categories for convenience. A coarse division is given below.

Syntax

Grammar induction[13]
: Generate a formal grammar that describes a language's syntax.

Lemmatization
: The task of removing inflectional endings only and to return the base dictionary form of a word which is also known as a lemma.

Morphological segmentation
: Separate words into individual morphemes and identify the class of the morphemes. The difficulty of this task depends greatly on the complexity of the morphology (i.e. the structure of words) of the language being considered. English has fairly simple morphology, especially inflectional morphology, and thus it is often possible to ignore this task entirely and simply model all possible forms of a word (e.g. "open, opens, opened, opening") as separate words. In languages such as Turkish or Meitei,[14] a highly agglutinated Indian language, however, such an approach is not possible, as each dictionary entry has thousands of possible word forms.

Part-of-speech tagging
: Given a sentence, determine the part of speech for each word. Many words, especially common ones, can serve as multiple parts of speech. For example, "book" can be a noun ("the book on the table") or verb ("to book a flight"); "set" can be a noun, verb or adjective; and "out" can be any of at least five different parts of speech. Some languages have more such ambiguity than others. Languages with little inflectional morphology, such as English, are particularly prone to such ambiguity. Chinese is prone to such ambiguity because it is a tonal language during verbalization. Such inflection is not readily conveyed via the entities employed within the orthography to convey intended meaning.

Parsing
: (see also: Stochastic grammar) Determine the parse tree (grammatical analysis) of a given sentence. The grammar for natural languages is ambiguous and typical sentences have multiple possible analyses. In fact, perhaps surprisingly, for a typical sentence there may be thousands of potential parses (most of which will seem completely nonsensical to a human). There are two primary types of parsing, Dependency Parsing and Constituency Parsing. Dependency Parsing focuses on the relationships between words in a sentence (marking things like Primary Objects and predicates), whereas Constituency Parsing focuses on building out the Parse Tree using a Probabilistic Context-Free Grammar (PCFG).

Sentence breaking (also known as sentence boundary disambiguation)

Given a chunk of text, find the sentence boundaries. Sentence boundaries are often marked by periods or other punctuation marks, but these same characters can serve other purposes (e.g. marking abbreviations).

Stemming

Word segmentation

Separate a chunk of continuous text into separate words. For a language like English, this is fairly trivial, since words are usually separated by spaces. However, some written languages like Chinese, Japanese and Thai do not mark word boundaries in such a fashion, and in those languages text segmentation is a significant task requiring knowledge of the vocabulary and morphology of words in the language. Sometimes this process is also used in cases like Bag of Words (BOW) creation in data mining.

Terminology extraction

The goal of terminology extraction is to automatically extract relevant terms from a given corpus.

Semantics

Lexical semantics

What is the computational meaning of individual words in context?

Machine translation

Automatically translate text from one human language to another. This is one of the most difficult problems, and is a member of a class of problems colloquially termed "AI-complete", i.e. requiring all of the different types of knowledge that humans possess (grammar, semantics, facts about the real world, etc.) in order to solve properly.

Named entity recognition (NER)

Given a stream of text, determine which items in the text map to proper names, such as people or places, and what the type of each such name is (e.g. person, location, organization). Note that, although capitalization can aid in recognizing named entities in languages such as English, this information cannot aid in determining the type of named entity, and in any case is often inaccurate or insufficient. For example, the first letter of a sentence is also capitalized, and named entities often span several words, only some of which are capitalized. Furthermore, many other languages in non-Western scripts (e.g. Chinese or Arabic) do not have any capitalization at all, and even languages with capitalization may not consistently use it to distinguish names. For example, German capitalizes all nouns, regardless of whether they are names, and French and Spanish do not capitalize names that serve as adjectives.

Natural language generation

Convert information from computer databases or semantic intents into readable human language.

Natural language understanding

Convert chunks of text into more formal representations such as first-order logic structures that are easier for computer programs to manipulate. Natural language understanding involves the identification of the intended semantic from the multiple possible semantics which can be derived from a natural language expression which usually takes the form of organized notations of natural language concepts. Introduction and creation of language metamodel and ontology are efficient however empirical solutions. An explicit formalization of natural language semantics without confusions with implicit assumptions such as closed-world assumption (CWA) vs. open-world assumption, or subjective Yes/No vs. objective True/False is expected for the construction of a basis of semantics formalization.[15]

Optical character recognition (OCR)

Given an image representing printed text, determine the corresponding text.

Question answering

Given a human-language question, determine its answer. Typical questions have a specific right answer (such as "What is the capital of Canada?"), but sometimes open-ended questions are also considered (such as "What is the meaning of life?"). Recent works have looked at even more complex questions.[16]

Recognizing Textual entailment

Given two text fragments, determine if one being true entails the other, entails the other's negation, or allows the other to be either true or false.[17]

Relationship extraction
: Given a chunk of text, identify the relationships among named entities (e.g. who is married to whom).

Sentiment analysis (see also multimodal sentiment analysis)
: Extract subjective information usually from a set of documents, often using online reviews to determine "polarity" about specific objects. It is especially useful for identifying trends of public opinion in the social media, for the purpose of marketing.

Topic segmentation and recognition
: Given a chunk of text, separate it into segments each of which is devoted to a topic, and identify the topic of the segment.

Word sense disambiguation
: Many words have more than one meaning; we have to select the meaning which makes the most sense in context. For this problem, we are typically given a list of words and associated word senses, e.g. from a dictionary or from an online resource such as WordNet.

Discourse

Automatic summarization
: Produce a readable summary of a chunk of text. Often used to provide summaries of text of a known type, such as articles in the financial section of a newspaper.

Coreference resolution
: Given a sentence or larger chunk of text, determine which words ("mentions") refer to the same objects ("entities"). Anaphora resolution is a specific example of this task, and is specifically concerned with matching up pronouns with the nouns or names to which they refer. The more general task of coreference resolution also includes identifying so-called "bridging relationships" involving referring expressions. For example, in a sentence such as "He entered John's house through the front door", "the front door" is a referring expression and the bridging relationship to be identified is the fact that the door being referred to is the front door of John's house (rather than of some other structure that might also be referred to).

Discourse analysis
: This rubric includes a number of related tasks. One task is identifying the discourse structure of connected text, i.e. the nature of the discourse relationships between sentences (e.g. elaboration, explanation, contrast). Another possible task is recognizing and classifying the speech acts in a chunk of text (e.g. yes-no question, content question, statement, assertion, etc.).

Speech

Speech recognition
: Given a sound clip of a person or people speaking, determine the textual representation of the speech. This is the opposite of text to speech and is one of the extremely difficult problems colloquially termed "AI-complete" (see above). In natural speech there are hardly any pauses between successive words, and thus speech segmentation is a necessary subtask of speech recognition (see below). Note also that in most spoken languages, the sounds representing successive letters blend into each other in a process termed coarticulation, so the conversion of the analog signal to discrete characters can be a very difficult process. Also, given that words in the same language are spoken by people with different accents, the speech recognition software must be able to recognize the wide variety of input as being identical to each other in terms of its textual equivalent.

Speech segmentation
: Given a sound clip of a person or people speaking, separate it into words. A subtask of speech recognition and typically grouped with it.

Text-to-speech
: Given a text, transform those units and produce a spoken representation. Text-to-speech can be used to aid the visually impaired.[18]

See also

- Automated essay scoring
- Biomedical text mining

- Compound term processing
- Computational linguistics
- Computer-assisted reviewing
- Controlled natural language
- Deep learning
- Deep linguistic processing
- Foreign language reading aid
- Foreign language writing aid
- Information extraction
- Information retrieval
- Language and Communication Technologies
- Language technology
- Latent Dirichlet allocation (LDA)
- Latent semantic indexing
- List of natural language processing toolkits
- Naomi Sager
- Native-language identification
- Natural language programming
- Natural language search
- Query expansion
- Reification (linguistics)
- Semantic folding
- Speech processing
- Spoken dialogue system
- Text-proofing
- Text simplification
- Thought vector
- Truecasing
- Question answering
- Word2vec

References

1. Implementing an online help desk system based on conversational agent (http://portal.acm.org/citation.cfm?id=1643823.1643908) Authors: Alisa Kongthon, Chatchawal Sangkeettrakarn, Sarawoot Kongyoung and Choochart Haruechaiyasak. Published by ACM 2009 Article, Bibliometrics Data Bibliometrics. Published in: Proceeding, MEDES '09 Proceedings of the International Conference on Management of Emergent Digital EcoSystems, ACM New York, NY, USA. ISBN 978-1-60558-829-2, doi:10.1145/1643823.1643908 (https://doi.org/10.1145%2F1643823.1643908)
2. Hutchins, J. (2005). "The history of machine translation in a nutshell" (http://www.hutchinsweb.me.uk/Nutshell-2005.pdf) (PDF).
3. Chomskyan linguistics encourages the investigation of "corner cases" that stress the limits of its theoretical models (comparable to pathological phenomena in mathematics), typically created using thought experiments, rather than the systematic investigation of typical phenomena that occur in real-world data, as is the case in corpus linguistics. The creation and use of such corpora of real-world data is a fundamental part of machine-learning algorithms for natural language processing. In addition, theoretical underpinnings of Chomskyan linguistics such as the so-called "poverty of the stimulus" argument entail that general learning algorithms, as are typically used in machine learning, cannot be successful in language processing. As a result, the Chomskyan paradigm discouraged the application of such models to language processing.
4. Goldberg, Yoav (2016). A Primer on Neural Network Models for Natural Language Processing (https://www.jair.org/media/4992/live-4992-9623-jair.pdf). Journal of Artificial Intelligence Research 57 (2016) 345–420
5. Ian Goodfellow, Yoshua Bengio and Aaron Courville. http://www.deeplearningbook.org/ Deep Learning]. MIT Press.
6. Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu (2016). https://arxiv.org/abs/1602.02410 Exploring the Limits of Language Modeling
7. Do Kook Choe and Eugene Charniak (EMNLP 2016). https://aclanthology.coli.uni-saarland.de/papers/D16-1257/d16-1257 Parsing as Language Modeling
8. Vinyals, Oriol, et al. (NIPS2015). https://papers.nips.cc/paper/5635-grammar-as-a-foreign-language.pdf
9. Winograd, Terry (1971). Procedures as a Representation for Data in a Computer Program for Understanding Natural Language. http://hci.stanford.edu/winograd/shrdlu/
10. Roger C. Schank and Robert P. Abelson (1977). Scripts, plans, goals, and understanding: An inquiry into human knowledge structures
11. Mark Johnson. How the statistical revolution changes (computational) linguistics. (http://www.aclweb.org/anthology/W09-0103) Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics.
12. Philip Resnik. Four revolutions. (http://languagelog.ldc.upenn.edu/nll/?p=2946) Language Log, February 5, 2011.
13. Klein, Dan, and Christopher D. Manning. "Natural language grammar induction using a constituent-context model (http://papers.nips.cc/paper/1945-natural-language-grammar-induction-using-a-constituent-context-model.pdf)." Advances in neural information processing systems. 2002.
14. Kishorjit, N., Vidya Raj RK., Nirmal Y., and Sivaji B. (2012) "Manipuri Morpheme Identification (http://aclweb.org/anthology//W/W12/W12-5008.pdf)", Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing (SANLP), pages 95–108, COLING 2012, Mumbai, December 2012

15. Yucong Duan, Christophe Cruz (2011), *Formalizing Semantic of Natural Language through Conceptualization from Existence (http://www.ijimt.org/abstract/100-E00187.htm)*. Archived (https://web.archive.org/web/20111009135952/http://www.ijimt.org/abstract/100-E00187.htm) 2011-10-09 at the Wayback Machine International Journal of Innovation, Management and Technology(2011) 2 (1), pp. 37-42.
16. "Versatile question answering systems: seeing in synthesis (https://www.academia.edu/2475776/Versatile_question_answering_systems_seeing_in_synthesis)", Mittal et al., IJIIDS, 5(2), 119-142, 2011.
17. PASCAL Recognizing Textual Entailment Challenge (RTE-7) https://tac.nist.gov//2011/RTE/
18. Yi, Chucai; Tian, Yingli (2012), "Assistive Text Reading from Complex Background for Blind Persons", *Camera-Based Document Analysis and Recognition*, Springer Berlin Heidelberg, pp. 15–28, CiteSeerX 10.1.1.668.869 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.668.869), doi:10.1007/978-3-642-29364-1_2 (https://doi.org/10.1007%2F978-3-642-29364-1_2), ISBN 9783642293634

Further reading

- Bates, M (1995). "Models of natural language understanding" (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC40721). *Proceedings of the National Academy of Sciences of the United States of America*. **92** (22): 9977–9982. doi:10.1073/pnas.92.22.9977 (https://doi.org/10.1073%2Fpnas.92.22.9977). PMC 40721 (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC40721). PMID 7479812 (https://www.ncbi.nlm.nih.gov/pubmed/7479812).
- Steven Bird, Ewan Klein, and Edward Loper (2009). *Natural Language Processing with Python*. O'Reilly Media. ISBN 978-0-596-51649-9.
- Daniel Jurafsky and James H. Martin (2008). *Speech and Language Processing*, 2nd edition. Pearson Prentice Hall. ISBN 978-0-13-187321-6.
- Mohamed Zakaria Kurdi (2016). *Natural Language Processing and Computational Linguistics: speech, morphology, and syntax*, Volume 1. ISTE-Wiley. ISBN 978-1848218482.
- Mohamed Zakaria Kurdi (2017). *Natural Language Processing and Computational Linguistics: semantics, discourse, and applications*, Volume 2. ISTE-Wiley. ISBN 978-1848219212.
- Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press. ISBN 978-0-521-86571-5. Official html and pdf versions available without charge. (http://nlp.stanford.edu/IR-book/)
- Christopher D. Manning and Hinrich Schütze (1999). *Foundations of Statistical Natural Language Processing*. The MIT Press. ISBN 978-0-262-13360-9.
- David M. W. Powers and Christopher C. R. Turk (1989). *Machine Learning of Natural Language*. Springer-Verlag. ISBN 978-0-387-19557-5.
