1

Speech and Translation Technologies

Explanations

MARK SELIGMAN

1.1 Introduction

The need for cross-language communication in healthcare is clear: Every day and everywhere, thousands of conversations take place between patients and caregivers – not only doctors and nurses, but administrators, volunteers, and others – whose native languages don't match. The circumstances vary widely, and the requirements for translation differ along with them. Some patients are literate, and some are not; some speak the caregivers' language sufficiently for effective communication concerning care, and some do not. Some patients are able to visit caregivers in person – or vice versa – while some must communicate remotely by phone, dedicated video, or internet audio and video.

Technology promising to assist this communication is developing explosively. The major linguistic technologies – machine translation (MT) of text, automatic speech recognition (ASR), text-to-speech (TTS) – have all improved dramatically in the era of neural networks, and so have the enabling elements of infrastructure – wireless communication, cloud computing, and mobile devices. By now, one would expect various forms of automatic translation and speech-enabled systems to have taken the healthcare world by storm, but adoption has in fact been sluggish. We'll examine the reasons for the speed bumps in Chapter 2, along with possible measures to surmount them.

One key factor in the lagging adoption, however, is the difficulty of understanding the relevant technologies, and thus the natural hesitation to trust them. Accordingly, this chapter aims to promote informed use by bridging the understanding gap for healthcare workers.

We begin with speech – its recognition (Section 1.2), its synthesis (Section 1.3), and related issues. Moving on to MT (Section 1.4), we'll first note the availability of systems covering only pretranslated phrases. We'll then go on to examine the major types of MT with broader coverage – "full MT,"

whether rule-based, statistical, or neural. As a bonus, we'll add extended discussion of transformer-based neural processing at the current state of the art. We'll conclude with some requisite cautions, and with a send-off to Chapter 2 shifting focus to practical applications of these technologies in the healthcare context.

1.2 Automatic Speech Recognition

1.2.1 Classical Automatic Speech Recognition

Automatic speech recognition has made dramatic progress in the last two decades. Throughout the 2000s, *speaker-dependent* ASR remained dominant: To achieve acceptable accuracy using commercially available ASR, each speaker had to provide speech samples, initially twenty minutes or more. In most systems, the speech signal to be converted into text was sliced into short segments, so that the system could estimate the probability of certain text sequences given a sequence of sound slices, generally using hidden Markov models (HMMs).[1] These estimates yielded possible words or word fragments and their probability rankings; and one could go on to estimate which *word* sequences were most likely, using compilations of word sequence probabilities called language models.[2] The search through the associated set of possibilities – the associated space of possible words and word sequences – was usually managed through some variant of Viterbi search techniques.[3]

By means of these techniques, and with sufficient speaker-specific and domain-specific recordings and accurate transcripts as training material, accuracies well above 90 percent became feasible in favorable environments. Necessary recording time dropped in a few years from twenty-plus minutes to less than a minute as processing power steadily increased according to Moore's law – the observation that computers' processing power doubles every two years or so[4] – and as usable recording databases became much larger. As a result, speaker-*independent* training had finally arrived by the early 2010s: That is, training time per new speaker had dropped to zero!

[1] "Hidden Markov Model." *Wikipedia*, Wikimedia Foundation, July 18, 2022, at 05: 21(UTC), https://en.wikipedia.org/wiki/Hidden_Markov_model.
[2] "Language Model." *Wikipedia*, Wikimedia Foundation, August 5, 2022, at 09: 29(UTC), https://en.wikipedia.org/wiki/Language_model.
[3] "Viterbi Algorithm." *Wikipedia*, Wikimedia Foundation, 12 March 2022, at 20: 26(UTC), https://en.wikipedia.org/wiki/Viterbi_algorithm.
[4] "Moore's law." *Wikipedia*, Wikimedia Foundation, July 30, 2022, at 18: 02(UTC), https://en.wikipedia.org/wiki/Moore%27s_law.

1.2.2 Neural Automatic Speech Recognition

Then neural speech recognition appeared on the scene: By the late 2010s, deep neural networks (DNNs) had essentially replaced HMM-based systems. Neural network models are fundamentally learners of input-to-output functions: When given certain patterns as input, they learn to yield certain patterns as output. (We'll look further into the details in Section 1.4.2.3.) And so, for ASR, when given suitably preprocessed speech signals, they can learn to deliver the most probable text transcripts. However, since speech recognition involves mediating between sequential patterns for both input (sequences of sounds) and output (sequences of graphemes – that is, letters or characters – and words), neural architectures specialized for sequences are essential. Until recently, recurrent and convolutional architectures were preferred – the first designed, when computing sound-to-text probabilities for the next step along a sequence in progress, to accumulate the output of all prior steps and include these as input, and the second designed to exploit a window moving across the sequence. These have now made room for transformer-based neural setups. These exploit a method called attention to focus upon the elements in a segment that will provide the most meaningful context to enable prediction of new sequences. (Transformers and attention are further discussed in Section 1.4.2.3.)

1.2.3 Automatic Speech Recognition Issues and Directions

1.2.3.1 Automatic Speech Recognition Issues

Numerous problems remain. Much speech, whether collected in real time or from recordings, is spontaneous rather than based upon written materials and consequently contains hesitations, stutters, repetitions, fragments, and other features unfriendly to recognition. Speech often occurs in noisy environments. It often involves multiparty conversations, with several voices that tend to overlap. The voices may be speaking different dialects and may even mix languages.

To address these and other issues, ASR development is continually in progress beyond neural network techniques themselves. Numerous possible architectural variations and component interactions can be tried according to the use case. For example, several varieties of noise reduction modules can deliver cleaner audio input (Li et al., 2014).

Integration of knowledge sources will also be a fruitful ongoing research direction. Presently, ASR still usually lacks any attempt to understand the objects and relationships in the speech situation.

1.2.3.2 Automatic Speech Recognition Directions

Considerations of understanding raise the question of future use cases for ASR. As one example, consider self-driving cars equipped with noise-resistant speech recognition: A car will "know" about its dynamic environment, having acquired from "experience" (multiple instances) visual "concepts" like CAR, TRUCK, and STREET, and their spatial and causative relations. And so, when recognizing user questions or commands concerning cars, trucks, streets, and so on, the car will be able to use knowledge about the referents – and not only the audio – to raise or lower probabilities of currently recognized text in context. And a car's concepts could include not only visual percepts but also sounds, vibrations, and lidar or radar data – a wide range of sensor data. In coming years, this incorporation of perceptually grounded knowledge is likely to transform all areas of artificial intelligence, speech recognition not least. The results will affect speech translation; transcription of all audio and video (real-time and otherwise); and, in fact, every use case demanding ASR – roughly, every use case involving speech.

To enable an informal impression of current speech recognition accuracy, we supply, in Appendix 1.1, healthcare-oriented ASR examples for English, using two current commercially available systems.

1.3 Speech Synthesis (Text-to-Speech)

Synthetic speech reached an acceptable quality level – understandable if colorless and unmistakably artificial – in the nineties. The problem was considered largely solved; and, partly for that reason, text-to-speech remained relatively static while speech recognition was rapidly and noticeably improving. We'll look at "classical" text-to-speech first, then move on to the current neural era.

1.3.1 Classical Text-to-Speech

1.3.1.1 Concatenative Text-to-Speech

The most widely used classical technology – still in use for some purposes – was concatenative: short, recorded audio segments associated with speech sounds (*phonemes* like /t/ or /o/ and their subparts or groupings) were stitched together (concatenated) to compose words and larger units.[5]

[5] "Speech Synthesis#Concatenation Synthesis." *Wikipedia*, Wikimedia Foundation, August 12, 2022, at 14: 44(UTC), https://en.wikipedia.org/wiki/Speech_synthesis#Concatenation_synthesis.

The segments in question were collected from large databases of recorded speech. Utterances were segmented into individual phones, syllables, words, and so on, usually employing a special-purpose speech recognition system yielding an alignment between sound elements and those linguistic units. An index of the units was compiled, based on the segmentation and on acoustic parameters (factors) including pitch, duration, and position among other units. And then, to build a target utterance given a text, the best chain of candidate units was selected, typically using a decision tree (sequence of program-based "questions") while extending the chain. Good results could be achieved, but maximum naturalness required large recording databases, up to dozens of hours. (Alternatives to such concatenative text-to-speech could synthesize utterances from scratch by artificially generating waveforms – the graphic representations of waves, describing them in terms of frequency and amplitude, which can be converted into actual sound. The resulting speech was less natural, but waveform methods had advantages, for example, in size, so that they lent themselves to implementations in small devices, even toys.)

1.3.1.2 General Text-to-Speech Issues

Concatenative or otherwise, any speech synthesis system confronts several issues.

Allophones and Coarticulation Phonemes are generally pronounced differently (as allophones, or phoneme variants) according to their place in words or phrases. For instance, in US English, phoneme /t/ may be pronounced with or without a puff of air (called *aspiration*, present in *top* but absent in *pot*). Moreover, even those variants – and all other speech sounds – will vary further in context according to the neighboring sounds (i.e., to coarticulation effects): For instance, the puffed /t/ sounds different before different vowels. (For this reason, diphones, or pairs of phonemes, are frequently used as speech sound groupings.) Coarticulation changes arising from some sound sequences can be dramatic in certain styles or registers, as when /t/+/y/ in *don't you* becomes the /ʧ/ of *doncha*. If classical TTS handled such cases – they usually didn't – it was through indicative spellings ("doncha") or through programs implementing handwritten combinatory rules.

Disambiguation Another problem is posed by text sequences that can be pronounced entirely differently according to their use in a sentence, like "record" in "For the *re*cord, . . . " versus "We need to re*cord* this meeting."

Some analysis of sentences is needed to select the appropriate variant and resolve the ambiguity – that is, to perform *disambiguation*. In classical TTS, this need was often met by symbolic (handwritten) programs for sentence analysis.

Normalization Yet another challenge is presented by text elements whose pronunciation isn't specified in text at all but is instead left to the knowledge of the reader-out-loud. Numbers and dates are typical examples: 7/2/21 might be pronounced as "July second, twenty twenty-one" in the US – though variants are many, even leaving aside the matter of European writing conventions. Some ways must be found to convert such elements to pronounceable text – to normalize the input.

Pronunciation problems Foreign or unfamiliar words ("Just hang a U-ie on El Camino") present obvious difficulties for text-to-speech. They're normally addressed either through compilation of specialized or custom dictionaries or through use of a *guesser* – a program that uses rules (then) or machine learning (now) to guess the most likely pronunciation.

Prosody Some treatment is needed of *prosody* – movement of pitch (melody), duration (rhythm), and volume (loudness). In the classical era, the prosody of a sentence was superimposed on speech units via various digital signal processing techniques. For instance, via the Pitch Synchronous Overlap and Add (PSOLA) technique, the speech waveform is divided into small overlapping segments that can be moved further apart to decrease the pitch, or closer together to increase it. Segments could be repeated multiple times to increase the duration of a section or eliminated to decrease it. The final segments were combined by overlapping them and smoothing the overlap. The means of predicting the appropriate prosody were relatively simple – for example, by reference to punctuation – so the results were often repetitive and lacking in expression.

Extraprosodic speech features Extraprosodic speech features like breathiness, vocal tension, creakiness, and so on were only occasionally treated in research, for example, by simulating the physics of the voice tract. Using models of vocal frequency jitter and tremor and of airflow noise and laryngeal asymmetries, one system was engineered to mimic the timbre of vocally challenged speakers, giving controlled levels of roughness, breathiness, and strain (Englert et al., 2016).

1.3.2 Neural Text-to-Speech

As mentioned, neural technology learns input-to-output functions – usually from corpora of input-output examples. For neural speech synthesis, the synthesis job can be understood as two input-to-output problems or stages:

1. Given text (perhaps revised or augmented with markup), what should be the corresponding acoustic features (numbers indicating factors like segment pitch, duration, etc.)?
    - The acoustic features are represented as spectrograms, which show frequency changes over time: In an X/Y graph, the vertical (Y) axis shows frequency, and the horizontal (X) axis shows time. (These days, a modified frequency scale is often substituted for raw frequency: the mel frequency scale – mel for "melody" – which takes account of human perception.)
2. Given acoustic features, what actual sound should be generated? This is the function of a vocoder.[6]

However, the stages can be combined to yield an end-to-end neural text-to-speech solution.

The prerequisites for neural text-to-speech began as recently as 2016, when DeepMind demonstrated networks able to perform the second stage by generating speech from acoustic features.[7] In 2017, the technology was used by others (Sotelo et al., 2017) to produce an initial end-to-end solution – generating speech directly from text. At the same time, Google and Facebook offered Tacotron and VoiceLoop, which could perform the first stage – that is, generate acoustic features, as opposed to sound, from input text. Completing the R&D pathway, Google proposed Tacotron2 as a more mature end-to-end solution, combining a revised acoustic feature generator (the first stage) with the WaveNet vocoder (the second stage).[8]

Now that current end-to-end systems can generate speech whose color (timbre) and overall resemblance approaches that of humans, this methodology has been widely adopted (Tan et al., 2021). Good models for given speakers or languages can be created with little engineering. They're robust, since there are no components that can fail. And unlike classical concatenative models, they require no large databases at run time.

[6] "Vocoder." *Wikipedia, Wikimedia* Foundation, 7 August 2022, at 18: 38(UTC), https://en.wikipedia.org/wiki/Vocoder.
[7] "WaveNet." *Wikipedia, Wikimedia* Foundation, 18 July 2022, at 17: 07(UTC), https://en.wikipedia.org/wiki/WaveNet.
[8] "Deep Learning Speech Synthesis." *Wikipedia*, Wikimedia Foundation, June 6, 2022, at 17: 58 (UTC), https://en.wikipedia.org/wiki/Deep_learning_speech_synthesis.

1.3.2.1 Neural Text-to-Speech Issues

But, of course, challenges remain:

- Learning of models takes much time and computation. Resolution efforts have emphasized architectural variation: Transformer-based architecture (Section 1.4.2.3.1) can replace older methods, with several advantages, including efficiency.
- If training data is insufficient or low in quality, speech quality suffers. The quality problem is related to failures of alignment between text and speech sounds, so focus has been on improving alignment by leveraging the known relations between these elements.
- Control points are absent: What you hear is what you get. Research has stressed methods of learning representations of certain speech features as embeddings, or points in multidimensional (vector) space (Section 1.4.2.2). The points can represent emotions (like anger or sadness) as expressed through speech features like pitch or rhythm. Because that representation remains separate from, for example, the pronunciation, many combinations and blends are possible.
- Prosody and pronunciation tend to be flat, since they're averaged over large collections of training data. At or after synthesis time, users can interactively post-tune preliminary flat (emotionless, bland, boring) renderings via suitable user interfaces. In addition, TTS models can be made to generate speech with various speaker styles and characteristics by utilizing embeddings representing speakers and speaking styles.

1.3.2.2 Neural Vocoders

We mentioned that neural speech synthesis involves two stages, where the second is sound generation, as performed by a vocoder. That vocoder can exploit neural networks, as do the popular WaveNet ("WaveNet") and HiFi-GAN (Kong et al., 2021) vocoders.

1.4 Machine Translation

We now shift focus to MT. We'll glance at translation based on fixed phrases, postponing most discussion for Chapter 2, before shifting attention to various techniques for full (wide-ranging, relatively unrestricted) translation: rule-based, statistical, and neural. As an optional coda for AI-curious readers, we'll examine state-of-the-art neural translation techniques involving transformers – neural networks for sequence prediction that handle context in a powerful new way.

As we've seen, transformers can be applied advantageously to speech recognition and speech synthesis as well.

1.4.1 Machine Translation Based on Fixed Phrases

Several healthcare-oriented speech translation systems have been designed to handle pretranslated phrases only, rather than to attempt full MT of wide-ranging input. This design decision enhances reliability because it depends on (usually professional) translation in advance; and it aids customization per use case in that relevant phrases can be brought into the system as needed.

Speech translation systems of this type include a set of fixed and pretranslated phrases, each supplied with a prepared target-language translation. Within such a set, the task of speech recognition is to find the best match for the incoming source-language phrase so as to enable transmission of its prepared translation via text or text-to-speech. (Matches will often be inexact, so techniques for finding near misses will be required.) The translation may be augmented with, or even substituted by, audiovisual elements – images, videos, or audio clips. Chapter 2 offers further discussion of phrase-only speech translation systems, with description of sample systems.

1.4.2 Full Machine Translation: Beyond Fixed Phrases

We now survey development of MT from its beginnings in the 1950s to the current state of the art. Conveniently enough, progress in the field can be divided into three eras or paradigms:[9] those of rule-based, statistical, and neural MT. We'll devote a subsection to each paradigm. Each can be usefully viewed in terms of its treatment of meaning, or semantics: Rule-based methods have generally emphasized handmade semantic symbols; statistical methods have generally avoided semantic treatment or employed vector-based semantics, as will be explained; and neural methods have until now handled meaning as implicit within networks.

We aren't undertaking a full history of MT research and development. For that purpose, see instead for example Hutchins (2010). We postpone discussion of speech translation until Chapter 2.

[9] The word paradigm, when referring to a consensus among researchers about the legitimate concepts and procedures for a scientific enterprise, was introduced by (Kuhn, 1996). Here the progression of paradigms or eras tracks the shift from one way of handling machine translation to another.

1.4.2.1 Rule-Based Machine Translation

We begin our survey of MT with a review of rule-based approaches. These employ handwritten rules relating to grammar and word composition (morphology), side by side with handwritten programs, so that the style might instead have been termed handmade MT.

Handmade approaches are rare in current MT development, where neural approaches (Section 1.4.2.3) are now overwhelmingly favored. Still, legacy MT systems continue to employ them[10]; and examination of them is conceptually helpful in understanding neural approaches, which would otherwise appear as oracles – as "black boxes" whose inner workings are invisible and mysterious, into which one language enters and from which another language miraculously emerges.

Within the rule-based paradigm, then, three subapproaches can be distinguished: direct, transfer-based, and interlingua-based.

Throughout, we'll be referring to the source language (SL, the language we're translating *from*) and the target language (TL, the language we're translating *to*).

Intermediate Structures: Syntactic versus Semantic In comparing the three rule-based approaches, one important question is whether the approaches do or don't automatically derive steppingstones between the SL and TL. We'll call these *intermediate structures*.

Another significant consideration is the composition of any such go-between structures: Do they represent *syntactic* or *semantic* features of an utterance, or some mixture? Figure 1.1 illustrates this distinction.

Consider first the analysis of the Japanese phrase on the left. In their original order, the English glosses of the relevant Japanese words would be "car, (object marker), driving, do, person" – that is, "car-driving person," "person who drives/is driving a car." The analysis shows that we are dealing with a noun phrase; that it is composed of a verb phrase on the left and a noun on the right; that the verb phrase in turn contains a certain sort of phrase; and so on. This is strictly a part-to-whole analysis – a *syntactic* analysis, where syntax refers to the analysis of the parts of a speech segment (for example, a sentence) and their relation to the whole segment. It says nothing explicit about the *meaning* of the phrase.

By contrast, on the right, we do see an attempt to capture the meaning of this phrase. PERSON is this time shown as a semantic (meaning-related) object, presumably one which could be related within a graph relating classes,

[10] "Word Magic." URL: https://word-magic-translator-home-edition.software.informer.com/.

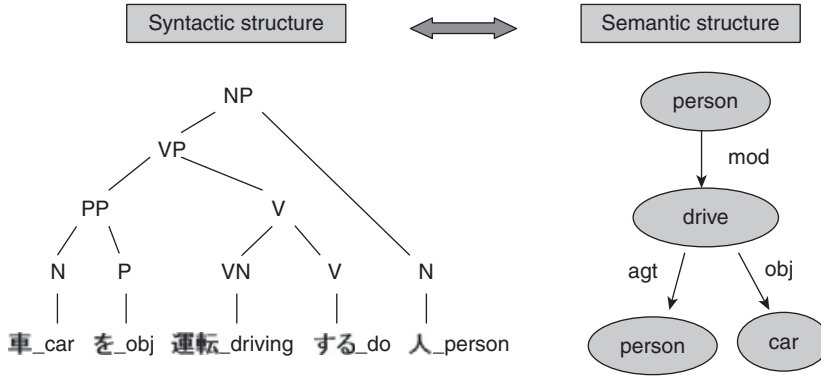

Figure 1.1 Contrasting syntactic and semantic intermediate structures

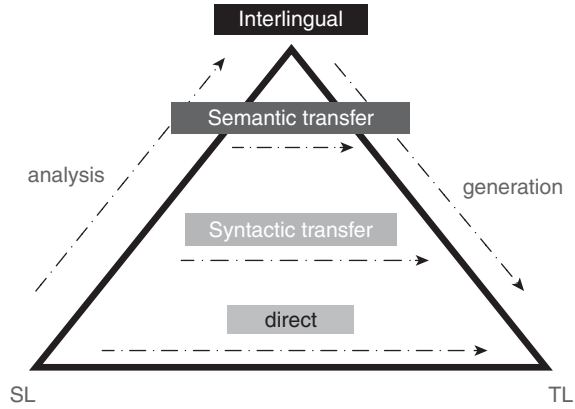

Figure 1.2 The Vauquois Triangle

subclasses, and instances – called an *ontology* – to other semantic objects such as ANIMALS, LIVING-THINGS, and so on. The PERSON in question is *modified* – a semantic rather than syntactic relationship – by the *action* DRIVE, and that modifying action has an *agent* (the same PERSON, though the identity is not shown) and an *object*, CAR.

In practice, such intermediate structures often mix syntactic (part-to-whole) and semantic (meaning-related) features, as we will see.

Vauquois Triangle We're ready now to contrast the three main approaches within the rule-based MT paradigm. For orientation, we refer to an often-used diagram of the relationships between direct, transfer-based, and interlingua-based methods (Figure tech.1.2), the *Vauquois Triangle* (Boitet, 2000).

The diagram depicts various paths for departing from the SL (at lower left) and arriving at the TL (at lower right).

Direct Translation We've drawn attention to this question for rule-based MT systems: are *intermediate structures* derived as go-betweens or steppingstones between SL and TL? The distinguishing feature of *direct* translation methods is precisely the absence of such midway points.

As a first step, *surface* elements of the SL – that is, the words and expressions in the input text – will undergo lookup to discover TL elements that can serve as their respective translations. (Several candidates might be found per element.) Programs will then be invoked to "massage" the target elements to compose a complete translation based upon them: to choose among translation candidates; to order the selected target elements properly; and to make necessary adjustments for TL morphology (word-building) and syntax (grouping and related modifications), for example, by handling agreement (making plural adjectives agree with plural nouns, for instance), adding function words or word parts (morphemes), and so on.

For such direct translation approaches, the diagram depicts a horizontal line between SL and TL which remains low in the triangle – low because, as mentioned, translation methods higher in the diagram use steppingstones (intermediate structures) on the way to a final translation, whereas direct methods do without them.

We've already seen examples of such steppingstones: the sample syntactic and semantic structures above. These intermediate structures are considered more *abstract* than the surface (text) elements; and height in the diagram is interpreted as degree of abstraction. (We'll say more about the interpretation of "abstraction" in a moment.) The intermediate structures include those derived through programs which perform *analysis* of the SL input (shown by the ascending line on the left): The structures produced by analysis should indicate the construction and meaning of the original SL input in ways not obvious from the surface language.

Transfer-Based Translation Above the horizontal line labeled "direct" is a line labeled "syntactic transfer." *Transfer-based* translation methods use *two* main intermediate structures. The first is the output of source-language analysis, as just described. The second intermediate structure should represent the construction and meaning of the input structure's translation into the TL. As such, it is intended to serve as the starting point for *generation* (construction) of the TL text (shown by the descending line on the right) and is derived from the analysis output through the *transfer process* for which transfer-based methods

are named. Transfer processes are somewhat analogous to the processes of direct translation in that they, too, begin by selecting TL elements that will translate source elements, and then go on to "massage" by reordering, adding, or subtracting, and so on. However, instead of massaging surface language elements, they massage the associated analysis output structure, for example by replacing one substructure with another to account for structural differences between source and target. (For example, the English structure <X> LIKE<Y> might be replaced by <Y> PLEASE <X> in Spanish – in translating "Carlos likes books" to yield "A Carlos le gustan los libros," literally, "To Carlos, books please him.")

We said that intermediate structures are intended to be increasingly *abstract* in the following special sense: The more abstract an intermediate structure, the greater the number of SL utterances which may have given rise to it during analysis or the greater the number of target utterances which might result during generation.[11]

Interlingua-Based Translation If the tendency toward abstraction is taken to its extreme, analysis aims to produce a maximally meaning-oriented (semantic) result – one which could in principle result from any source utterance having an equivalent meaning, regardless of sentence or word structure. The result should then be an *interlingua representation*, one intended to represent the semantics for *both* SL and TL, and ideally for many, or even all, additional languages. Once this degree of abstraction has been reached, intermediate structures on the source and target side are no longer distinct, so there will be no need of a transfer process to mediate between them. For this reason, *interlingua-based* translation methods are shown at the apex of the Vauquois Triangle, where horizontal transfer lines will no longer fit.

Having outlined rule-based or handmade translation methods – direct, transfer-based, or interlingua-based – we can comment on their treatment of semantics.

Semantics in Rule-Based Machine Translation Of course, no translation could take place without at least implicit consideration of meaning. In purely direct rule-based MT, the meaning of an expression is shown implicitly only by

[11] In many linguistic discussions, "abstraction" is discussed in terms of "depth," as in "deep structure." This terminology can be confusing, and not only because elements *higher* in the Vauquois triangle would be described as "deeper." Several metaphors are in competition: "deeper" may mean "dominant in a phrase structure," as a verb phrase symbol may dominate a verb and its object; "superordinate in an ontology," as a class like AIRCRAFT may be superordinate to its subclasses like HELICOPTERS; "earlier in a derivation sequence," as analysis of a source utterance precedes TL generation; and so on.

its translations: One could say that the translations *are* the meanings. There are typically several possible translations for any given expression, and examination can reveal semantic relations like SL polysemy (an expression has multiple meanings: one or more SLs map to several groupings of synonymous TLs) and SL synonymy (several expressions mean the same: when several SLs map to one grouping of synonymous TLs).

However, for direct translation systems or any others, we can go on to examine the role, if any, of *explicit* semantic methods. And we can observe that, while direct MT methods do concentrate upon the surface (text) elements of SL and TL, explicit information concerning the *meanings* of words and phrases can still be useful, for example to aid in the selection of the correct word meaning, and thus the correct translation, for *ambiguous* expressions, or expressions (like English *bank*) with multiple meanings – that is, for *lexical disambiguation*. As is widely known, ambiguity has long plagued the MT enterprise. The difficulty of avoiding the meaning "writing instrument" when translating "The pig is in the pen" prompted an influential early misjudgment that automatic translation would prove a dead end.[12]

An example appeared in the direct MT system of Word Magic for English-to-and-from-Spanish, in which translation lexicons listed not only surface expressions but *word-senses*, for example, *bank1* ("financial institution"), *bank2* ("shore"), *bank3* ("row, e.g. of switches"), and so on, where each listed word-sense pointed to a set of synonymous Spanish translations, in which one member was the default translation. During analysis, the appropriate word-sense – that is, meaning – for the current translation segment was chosen according to handwritten rules taking account of the context. For maximum generality, the disambiguation rules referred to *semantic classes* (e.g., VEHICLES) rather than individual semantic instances (e.g., CAR.1); and those classes were collected and arranged in an ontology (categorization graph). Among direct rule-based approaches, this treatment is typical (Hutchins, 2005).

However, within the rule-based MT paradigm, while some direct systems have used semantic symbols to good advantage, such elements are most associated with transfer-based and interlingua-based methodologies.

The ASURA system for English, German, and Japanese, an early speech translation system, included a transfer-based MT component intended to operate at the semantic level, in order to better bridge the gap between the disparate languages involved. Consider Figure 1.3, a structure produced by the transfer process during translation of "Could you make the hotel arrangements?" into

[12] "History of Machine Translation#The 1960s, the ALPAC report and the seventies." *Wikipedia*, Wikimedia Foundation, July 9, 2022, at 18: 01(UTC), https://en.wikipedia.org/wiki/History_of_machine_translation#The_1960s,_theALPAC_report_and_the_seventies.

```
;;;=========== Transfer Result ============
 [[SEM [[RELN REQUEST]
        [AGEN !X3[[LABEL *SPEAKER*]]]
        [RECP !X1[[LABEL *HEARER*]]]
        [OBJE [[RELN MACHEN-V]
               [TENSE PRES]
               [AGEN !X1]
               [OBJE [[PARM !X2[]]
                      [RESTR [[RELN HOTELBUCHUNG-N]
                              [ENTITY !X2]]]
                      [INDEX [[DETERM DEFART]
                              [NUMBER SING]
                              [OWNER [[LABEL *UNKNOWN*]]]]]]]]]
        [ATTD INTERROGATIVE]]]
  [PRAG [[RESTR [[IN [[FIRST [[RELN POLITE]]]
                      [REST [[FIRST [[RELN EMPATHY-DEGREE]
                                     [LESS !X1]
                                     [MORE !X3]]]
                             [REST [[FIRST [[RELN POLITE]]]
                                    [REST !X4[]]]]]]]]]
                [OUT !X4]]]
         [HEARER !X1]
         [INTIMACY LOW]
         [POLITENESS [[DEGREE 3]]]
         [SPEAKER !X3]]]]
```

Figure 1.3 A hybrid intermediate structure from the ASURA system

German (Seligman, 1993). The structure contains the semantic symbols REQUEST and POLITE alongside syntactic symbols like MACHEN-V ("to make," a verb) and HOTELBUCHUNG-N ("hotel booking," a noun).

As might be expected, the most extensive use of explicit symbolic semantic tokens has been in interlingua-based MT. Here a mature example is the ATLAS system for English and Japanese, developed at Fujitsu under the direction of Hiroshi Uchida (1986). Uchida is also the founder of the most extensive multilingual and multipartner interlingua-based research effort, the Universal Networking Language (UNL) project.[13] Its foundation is a rich set of word senses, originally based upon that of a complete English dictionary. These can be combined, via special relational symbols like CAUSE, to enable construction of UNL representations for phrases, sentences, and so on. Figure 1.4, for instance, shows the combination representing the following sentence and its

[13] "UNL project." URL: www.undlfoundation.org/undlfoundation/.

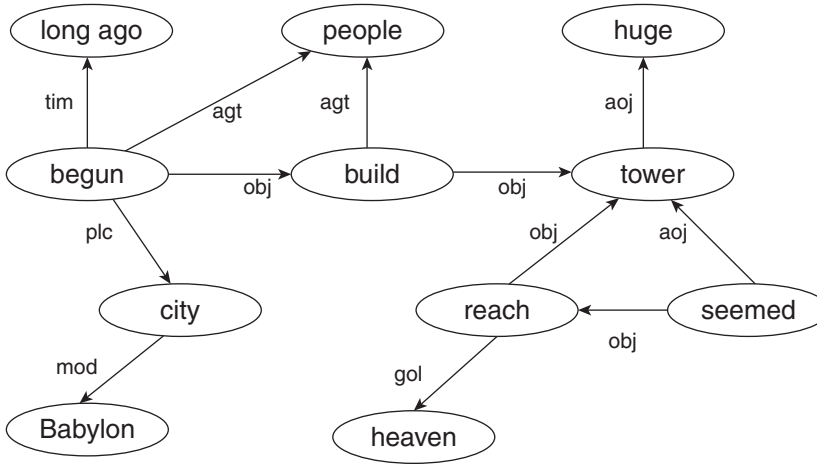

Figure 1.4 A sentence representation in the UNL interlingua

```
We have single, and twins and also Japanese rooms available on the eleventh.
a:give–information+availability+room (room–type=(single&twin&Japanese_style), time=md11)

I'd like a twin room, please.
c:accept+features+room(room–type=twin)

A twin room is fourteen thousand yen.
a:give–information+price+room(room–type=twin, price=(currency=yen, quantity=14000))
```

Figure 1.5 Sentence representations in the IF interlingua

many paraphrases and translations: "Long ago, in Babylon, the people began to build an enormous tower, which seemed to reach the sky."

Interlingua-based structures have been useful in speech translation research. See (Seligman and Waibel, 2018) and (Levin et al., 1998) regarding the Interchange Format (IF) structures used by the C-STAR consortium (Figure 1.5 shows three examples) and concerning a separate interlingua used in IBM's MASTOR project (Gao et al., 2006).

1.4.2.2 Statistical Machine Translation

A dramatic rise of *statistical machine translation* (SMT) (Koehn, 2009) erupted in the 1990s.

In initial implementations, statistical information was treated as a supplement or add-on to the existing rules and programs of rule-based MT (Brown et al., 1990, 1993). However, the new paradigm soon gravitated toward

methods that in some respects recalled those of direct rule-based MT. Rather than manipulate abstract structures like those of transfer-based methods – structures representing some mixture of compositional and semantic commonalities among surface structures – statistical methods returned to operations upon the surface structures themselves. As in direct rule-based methods, the first step is to determine which TL surface segments might serve as translations for SL surface segments; and later steps relate to the ordering of target elements, possible additions or subtractions from them, possible grammatical adjustments, and so on. But while in rule-based methods these steps depend on rules and programs created by hand, in SMT they depend upon probabilities discovered in *parallel corpora* of human translations (for example, a large collection of English parliamentary transcripts in which each utterance is aligned with its French translation). The goal in SMT is to produce the most *probable* translation of a source segment given that training set (*corpus*), so actual production of a translation (*decoding*) becomes an *optimization* process – a search for the best solution among many candidates, often visualized as *hill climbing*: the probabilities of alternative translations are iteratively compared, and with each matchup, the better alternative is chosen as a step uphill. The goal is to arrive at the highest probability "peak" (and avoid getting stuck on a lower one).

In most SMT, the translations of words and phrases *are* their meanings (just as they are in "pure" or unadorned direct rule-based MT). SMT's translations are indicated in a system's *phrase table*, a listing of SL-to-TL correspondences (e.g., English *cool* to French *frais*), each with a probability determined during training (Figure 1.6). The rows in a table can be examined to discover semantic relations like polysemy (one expression, many meanings) and synonymy (many expressions sharing a meaning).

Vector-based semantics. Throughout its decade-long reign, mainstream SMT exploited explicit semantic symbols only rarely. In compensation, *vector-based* semantic treatments gradually became influential.

Source language expression	Target language expression	Probability
cool	frais	.34
cool	chouette	.21
nippy	frais	.88
man	homme	.68

Figure 1.6 Part of a phrase table for statistical machine translation

Vector-based semantic research aims to leverage the statistical relationships among text segments (words, phrases, etc.) to place the segments in an *abstract space*, within which closeness represents similarity of meaning (Turney and Pantel, 2010).

"Abstract space" sounds impressive but intimidating; however, everyday comparisons can reduce the fear factor. For example, any spreadsheet with several rows (representing, e.g., available flavors of an ice cream order) and several columns (available sizes of an order) exemplifies a "space" with two dimensions – up-down and right-left – in which the cell entry in row 2, column 3 ("strawberry, large") indicates a specific combination, seen as a "location" or "point" within that "space" (set of choices). We could stack such spreadsheets vertically to make room for a third dimension (perhaps available containers, as in cone vs. cup); and so on, in theory, to any number of dimensions or factors.

Vectors themselves, meanwhile, are just one-dimensional lists of numbers representing combinations of factors, with one number coding each factor: <strawberry, large, cone> might be coded as the vector <2, 1, 1>.

Closeness or similarity in such a "space" of choices can be represented as distance between "points" in the space (comparable to locations or cells in a spreadsheet): two ice cream orders that share several factors (flavor, size, or container) are closer (more similar) in the sheet than those with fewer commonalities.

This insight can enable comparison of words or expressions with respect to their meanings. Intuitively, words that occur in similar contexts and participate in similar relations with other words should turn out to be semantically similar. The intuition goes back to Firth's (1957) declaration that "You shall know a word by the company it keeps," and has been formalized as the *distributional hypothesis*. The clustering in this similar-neighbors space yields a hierarchy (ranking) of similarity relations, comparable to that of a handwritten ontology (symbol categorization graph). Figure 1.7 (Mikolov et al., 2013) shows two examples from English with corresponding examples from Spanish.

Representation of a given segment's meaning as a location in such a vector space can be viewed as an alternative to representation as a symbol located within a categorization graph. The vector-based approach is much more scalable (more extensible to large-scale use) in that there is no need to build graphs manually; but relations can be harder for humans to comprehend in the absence of appropriate visualization software tools.

Historically, the vector-based approach grew out of document classification techniques, whereby a document can be categorized according to the words in it and their frequency. The converse was then proposed: A word or other linguistic unit can be categorized according to the documents it appears in, or more

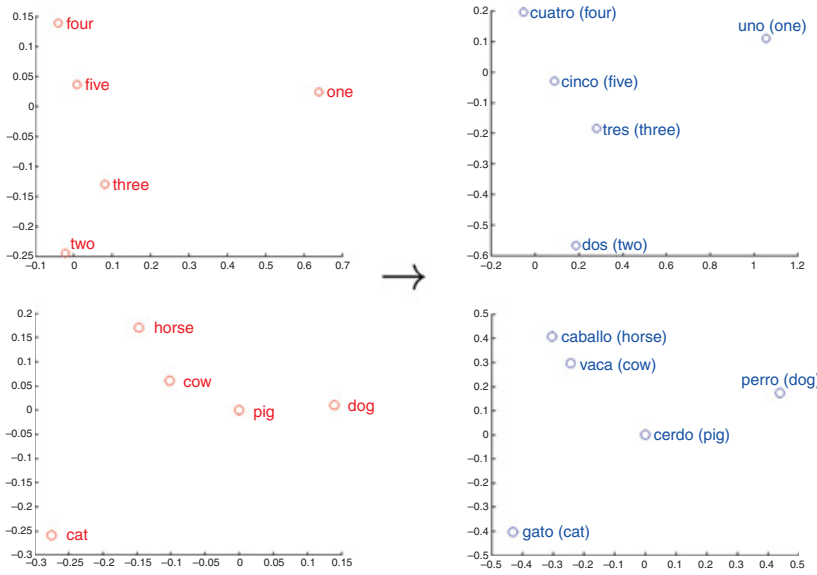

Figure 1.7 Two vector spaces for English, with corresponding Spanish spaces

generally, according to surrounding or nearby text segments of any size – minimally, just the few words surrounding it.

Vector-based semantic approaches have been used experimentally to improve statistical MT systems. Alkhouli et al. (2014) provide a clear example, in which the elements located in vector space according to their respective contexts are phrases (word groups) rather than only words. It then becomes possible to measure distances between phrases, interpretable as similarity of meaning; and this interpretation in turn enables enhancement of the translation process via artificial enlargement of the relevant phrase tables – helpful because the training set (corpus) rarely contains all the examples one would wish.

1.4.2.3 Neural Machine Translation

Neural machine translation (NMT) has proved to be a late bloomer. While early neural experiments (Waibel, 1987; Waibel et al., 1987, 1991) garnered interest, especially in view of potential insights into human language processing, the computational infrastructure that would eventually make neural approaches practical did not yet exist. Now that they do, the approach has experienced an explosive renaissance: Google announced its first neural translation systems as recently as 2016 (Johnson et al., 2016); Systran has since then gone fully neural

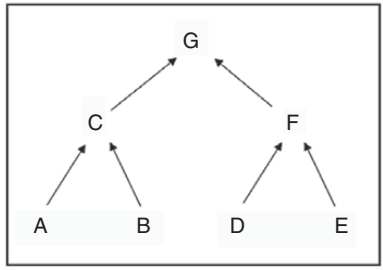

Figure 1.8 Connections among rules forming a network

(Senellart, 2018); and most other major MT vendors are converting at full speed.

A conceptual introduction to neural network operation may help to explain the methodology's application to translation. Think first of logical rules, for instance those of the predicate calculus:

If A and B, then C
If D and E, then F
If C and F, then G

If the premise-to-conclusion relations are depicted as lines, we obtain a tree-like diagram (Figure 1.8).

Imagine that the lines are electric wires, and that there is a bulb at each premise or conclusion which lights up if manually switched on, or if all incoming wires are active; and that, when a light is illuminated, the outgoing wire is activated. Switch on A, B, D, and E. Then C and F will be activated and will propagate activity to G. That is, since facts A, B, D, and E have been found to be true or in effect, fact G has been found to follow. *Et voilà*: a neural network! However, several refinements are needed to complete the picture.

- First, rather than being simply on or off, each line should have a *degree* of activation; and illumination of a conclusion bulb should require not full activation of all wires, but only summed activation passing a specified threshold.
- Second, some wires may inhibit rather than promote the conclusion – that is, their activation may subtract from the sum.
- Third, rather than only three "rules," there should be many thousands.
- And fourth, and perhaps most important, all of the network's parameters (numbers, factors) – the wires' activation levels, thresholds, and so on – should be learned from experience rather than set by hand. They may be

learned through a *supervised* process, whereby a trainer provides the expected conclusions given the switches thrown at the input, and appropriate programs work backwards to adjust the parameters; or through an *unsupervised* process, whereby adjustment depends on frequency of activation during training, perhaps assisted by hints and/or rewards or punishments.

Such networks can indeed be applied to translation, since they provide general-purpose computational mechanisms: With sufficient available wires, "rule" layers, and so on, they can in principle learn to compute any *function* – any mapping of input patterns to output patterns. Thus, they can learn to map input bulbs coding for SL segments into patterns analogous to the human-readable symbolic analysis results of an interlingua-based MT system – that is, to perform operations analogous to the analysis phase of such a system. (In NMT, the analysis phase is called *encoding*, and produces only human-opaque numbers.) Likewise, the networks can also learn to map those result patterns into the surface structures of the TL – that is, to perform operations (called *decoding*) analogous to a transfer-based system's generation phase. And they can learn the alignment between surface elements of the source segment with those of the target segment (that is, can learn which SL segments correspond to which TL segments), information helpful during TL generation (Figure 1.9). In Section 4.2.3.1, we'll see how it's done.

Neural networks were born to learn abstractions. The "hidden" layers in a neural network, those which mediate between the input and output layers, are designed to gradually form abstractions at multiple levels by determining which combinations of input elements, and which combinations of combinations,

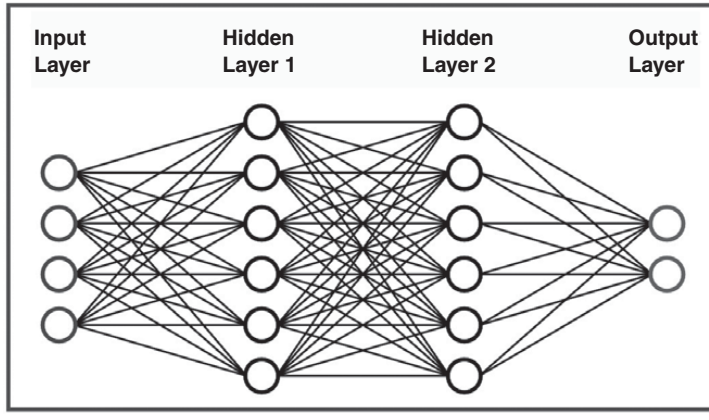

Figure 1.9 A neural network with input, output, and two hidden layers

are most significant in determining the appropriate output. (In our conceptual introduction above, each abstraction level was viewed as a stage in a chain of implied "rules." Rules close to the input layer of the network use surface elements specific to particular inputs as their "premises" or givens, while those further from the input use "premise" combinations taken from many inputs.) The more hidden layers, the more levels of abstraction become possible; and this is why *deep* neural networks are better at abstracting than shallow ones. This advantage has been evident in theory for some time; but deep networks only became practical when computational processing capacity became sufficient to handle multiple hidden layers.

Where MT is concerned, this hidden learning raises the possibility of training neural translators to develop internal meaning representations automatically and implicitly (Woszczyna et al., 1998). A new neural-network-based approach to meaning then suggests itself: Within a network, nodes or pathways shared by input elements having the same translation or translations can be seen as representing the shared meanings. Input elements sharing a translation can originate in a single SL (when in that language the source elements are synonyms in the current context) or in several SLs (when across the input languages in question the source elements are synonymous in their respective contexts). And, in fact, the shared translations, too, can be unilingual or multilingual.

Thus, if translation is trained over several languages, semantic representations may emerge that are abstracted away from – that become relatively independent of – the languages used in training. Taken together, they would compose a *neurally learned* interlingua, a language-neutral semantic representation comparable to the *handmade* symbolic interlingua discussed above in relation to rule-based systems. A successful neural interlingua could facilitate handling of languages for which data is sparse, thus opening a path to truly universal translation at manageable development costs. Several teams have begun work in this direction (Le et al., 2016; Kurzweil, 2016; Firat et al., 2016), and early results are already emerging: Google, for instance, has published on "zero-shot" NMT, so named because the approach allows translation between languages for which zero bilingual data was included in training corpora (Johnson et al., 2016); and SYSTRAN, in a similar spirit, has already announced combined translation systems for romance languages (Senellart, 2018). Not to be outdone, Meta (the company formerly known as Facebook) has recently announced a comparable push toward universal translation (Ramirez, 2022). Zero-shot NMT works because the encoding (analysis) phase of translation has been generalized across all currently trained SLs, while the decoding (generation) phrase has similarly been generalized across

all currently trained TLs. Thus any current source can be paired with any current target. Expectations would be low, however, if completely untrained SLs or TLs were tried.

Transformers in Neural Machine Translation For many readers, the above account of NMT will suffice. Still, in the spirit of dispelling the mystery, we'll go on to provide an optional bonus: an intermediate-level account of the inner workings of neural translation at the state of the art, focusing on recent excitement over the *transformer* architecture (i.e., learning setup) and its advantages. We've repeatedly mentioned transformers as neural networks that can exploit a powerful technique called attention to predict sequences by analyzing their elements' contexts. Now we're ready to scrutinize the role of that technique in learning large language models (LLMs) like GPT-3 – subjects of intense research in the artificial intelligence community at the time of writing. While few healthcare workers may participate directly in this research, it will be helpful if those charged with selecting speech and translation components are conversant with it. Artificial intelligence is resurgent, and demystification should be healthy for most professionals.

Analyzing Sequences: The Role of Context As we've seen, all three of the major components that concern us here involve analysis of sequences, and more specifically, transformation of one sequence into another: For speech recognition, we transform a sequence of sound segments into a sequence of text elements; for speech synthesis we do the reverse, transforming a text sequence into a sequence of sound segments; and for MT, we normally transform a text sequence in the SL into a text sequence in the TL. ("Normally" because some research attempts to transform sound sequences directly into sound sequences, without passing through text on the way.) It will be convenient to focus our exposition of forefront research on MT. However, the techniques to be examined can serve to predict sequences quite generally – *single-strand sequences* (those with only one row of elements) as well as the aligned, *double-strand sequences* of most immediate interest (in which two interrelated rows are in question).

Our MT life would be sweet if we could simply replace each source word with its unique translation at its original place in the source sequence. However, as we've seen, there are several problems with this simplistic approach, all depending heavily on the source *context* – the surrounding source words. First, source words will in general be ambiguous: They may have several possible translations, possibly including *no* translation. Second, the order of target words may be different from that of the source words. Third, agreement may

be required between certain elements of the target sequence. Finally, there will be pronouns and other referring words whose translation will depend on resolving the words they refer to.

How can we enable each word to be aware of its full context as we identify its translation counterpart? Until 2017, the standard answer was to step through the source word sequence one word at a time – for English, from left to right – while trying to "remember" earlier words and their translations. Information on all prior words and their translations was repeatedly fed to the process translating the current word. The setups that managed this recycling are called *recurrent neural networks* (RNNs), already mentioned in passing. They handled contextualization reasonably well for short sentences but less well when tackling longer ones, for several reasons.

- Memory of earlier elements tends to fade as the sequence progresses: The system forgets what happened early in the input as it progresses toward later elements. Consequently, only relatively recent context can have the desired, and crucial, influence.
- A related matter is the *vanishing gradient problem*. Neural networks learn by repeatedly measuring their errors from trial to trial so that they can adjust networks incrementally in the direction of the right outcome – up or down a metaphorical hill, or *gradient*; but if the differences between trials becomes too small and the hill flattens, such gradual adjustment becomes difficult, and learning grinds to a halt. This flattening is too frequent with recurrent techniques.
- Given the consecutive processing, elements *later* in the input can have no influence at all on the current word's analysis; and yet the entire sequence may have been accessible from the outset.
- The continuous recycling makes the entire progression resource-intensive and time-consuming.

Researchers attempting to alleviate these issues realized that *not all context is created equal*. For analysis of the current word, some neighbor words provide more significant context than others. So the relative context-worthiness of a word's neighbors should be estimated, and contextual influence on translation should be granted to them proportionately. But how can this be done?

"Attention" as Context-Worthiness We've already introduced the concept of vector-based semantics, in which words are categorized as semantically similar according to their respective contexts – their word neighbors. In neural MT, the vectors (embeddings) representing input words are just rows

of numbers, one number for each dimension (factor) in the abstract similarity "space." (These are supplied in advance, for instance by the BERT language model.[14]) Each word's vector represents its "location" in that "space": If there were only two dimensions, then a vector with two numbers referring to a standard X/Y axis would suffice, and we'd see the word's point somewhere in the plane thus defined; but the same principle applies for any number of dimensions. And here's the point: It turns out to be straightforward mathematically to measure the neighbor-based *similarity* of two words by calculating the distance between their vectors. Accordingly, we can let this sort of neighbor-based similarity be our measure for the context-worthiness, within the relevant segment, of each segment-mate word with respect to the current word. Context-worthiness, thus understood, is called *attention* in this technical sense; and it is in this sense that attention has captured the attention of the AI world.

Attention was initially used to augment the operation of RNNs; but in 2017, a seminal paper appeared: "Attention Is All You Need" (Vaswani et al., 2017). It showed that thoroughgoing use of attention could make unnecessary the massive recycling applied by RNNs: Instead, contextual influence could be calculated for each word separately. And this could be done by separate processors, and all at the same time – that is, in parallel! What's more, miraculous follow-on benefits were revealed: Context could become *much* larger and more complete, since it now became possible to consider the influence of segment-mate words at distances limited only by the length of the segment, rather than considering only the words recent enough to be clearly remembered. Then, too, similarity could be estimated not only for earlier words, but also for words *later* in a long segment. Parallel operation meant hugely faster operation than recurrent recycling; and hugely faster operation meant that huge amounts of data could be processed – essentially, *all* the text on the Internet! (And later, images and other types of data as well.) Meanwhile, the processing power that was saved could be spent on enlarging the neural networks themselves: They could be much wider and much deeper, with the number of connection strengths, and so on (i.e., of network *parameters*) to be learned during training reaching the billions. These fringe benefits jointly led to far greater abstraction and predictive power. What earlier was described as language models – we've encountered them earlier – had now become LLMs, *large* language models. First of these is Generative Pre-trained Transformer, version three – now famous in the field as GPT-3.

[14] "BERT language model." *Wikipedia*, Wikimedia Foundation, August 12, 2022, at 22: 00(UTC), https://en.wikipedia.org/wiki/BERT_(language_model).

Transformers can indeed be exploited for MT – we'll elaborate presently – but also, as we've seen, for speech recognition and speech synthesis.

Transformers as General-Purpose Predictors That's not all, however. They provide a general-purpose sequence prediction mechanism, so that any data that can be represented sequentially can be – and has been – fed to them and predicted by them. Images, video, audio, and action directives can all be chunked; and, given an initial sequence of chunks, GPT-3 and its successors (Wiggers, 2022) can predict the likely continuation (or several alternatives). Partial images can be completed or generated from scratch starting from a simple prompt. So can partial poems or novels, though quality is of course debatable. The system has in effect learned myriad schemas or templates, in which the fillers – the values of variables – are internally represented quite abstractly. Importantly, associations among different data types can also be learned – notably, between text and related images, so that perceptually grounded linguistic generalizations are formed: Certain abstract categories of images are associated with certain abstract categories of linguistic elements (Synced, 2021).

It turns out that, with sufficient input data and parameters, a single LLM can perform a wide range of tasks with varying degrees of measurable success.[15] In view of this progress, debate among AI researchers is ongoing: Is the general-purpose prediction ability gained by transformers the first step toward true general intelligence? What, if anything, is missing for true reasoning and understanding? Still, caution is warranted: Strikingly cogent predicted sequences are often accompanied by jarringly meaningless ones. And certainly, the appearance of understanding should not be mistaken for the real thing. On the other hand, I've suggested (Seligman, 2019) that a threshold would be crossed with the advent of *perceptually grounded* natural language processing, as opposed to processing based solely on text. That advent is now upon us. LLMs associating text and images are here, and those based on video with audio cannot be far off. These will bring the promise of true, if limited, *intentionality* – meaningful connection between linguistic elements and the perceived world.

Remember, too, that some of the best current systems in natural language processing – we're still focusing on translation – have not yet incorporated transformers at all, at least in system descriptions so far made public. For

[15] "Gato (Deep Mind)." *Wikipedia*, Wikimedia Foundation, June 25, 2022, at 21: 22(UTC), https://en.wikipedia.org/wiki/Gato_(Deep Mind).

example, the DeepL automatic translator,[16] developed by DeepL SE of Cologne, Germany, has achieved the impressive results displayed in Appendix II with the convolutional neural network (CNN) architecture, a competitor to RNNs, in which context is learned by moving a window around in the sequence under analysis.

So attention is quite generally useful for tracking the relevance or interdependency of sequence elements. That relevance can be tracked *across* sequences, as when relating source sequences to target sequences to recognize potential translation relations among source and target words; or *within* a given sequence, for example, within the source or target sequence (in which case one speaks of *self*-attention). Relevance can also be assessed for various aspects of a task: for example, in analysis of the source sentence, with respect to syntactic dependencies (like the relation between subjects and predicates, or nouns and their associated adjectives), or to semantic co-reference (as when *my aunt's pen* and *it* refer to the same entity). Each sort of relevance can be handled by a dedicated transformer *head*, giving rise to *multi-headed transformers*.

Transformers in Neural Machine Translation Equipped with this general understanding of attention in transformers, we can return to the NMT process specifically. We pick up the story at the encoding phase, which aims for an abstracted analysis of the entire input, comparable to the result of handprogrammed analysis in the transfer-based MT style, and fit for passing to the decoding (TL generation) phase. Actually, several encoders are normally used, for reasons to be explained. Since they operate one after another, they can be pictured as a stack of *encoder layers*, in which (we'll say) the highest encoder layer is the earliest in the process, and later layers progress downward toward decoding and eventual translated output (Figure 1.10). (N.b., Encoder layers, and later decoder layers, shouldn't be confused with the neuron ("bulb") layers within a single neural network.)

In any one of these encoder layers, multi-headed self-attention is applied to augment each word with various sorts of contextualized information. One essential factor in a word's context is its actual location in the input sequence; so that information must be added to the word's enrichment by blending into the word's vector a *position vector* representing, via some mathematical magic, the word's numbered position in the sequential order. Also added for good measure is another vector representing the current word as it emerged from any

[16] "DeepL Translator." *Wikipedia*, Wikimedia Foundation, August 10, 2022, at 17: 37(UTC), https://en.wikipedia.org/wiki/DeepL_Translator)).

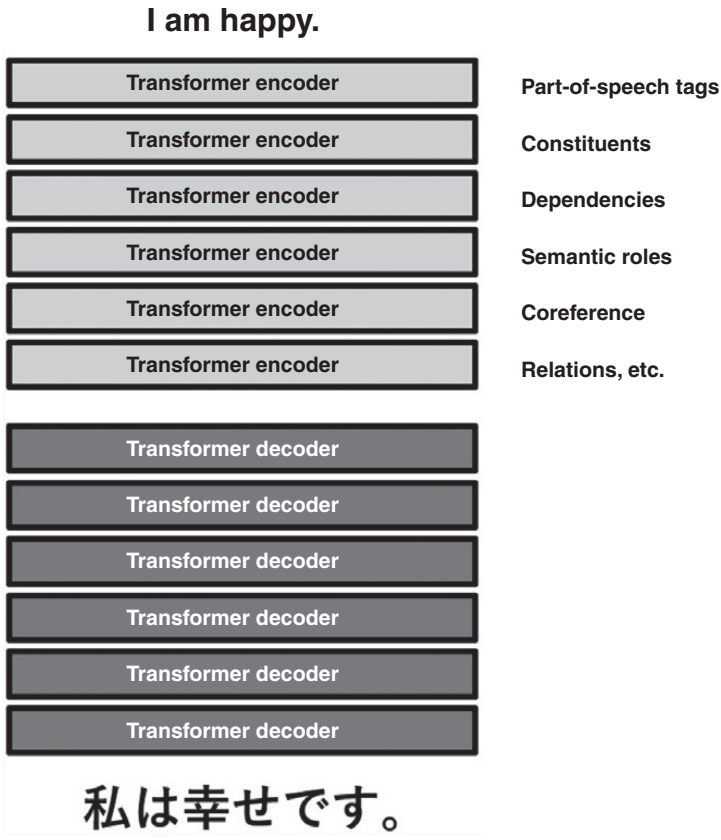

Figure 1.10 Encoder and decoder layers in a transformer-based MT system

earlier encoder layers in the stack, in effect preserving some memory of past analysis.

Then, to complete an encoder layer's operation, the self-attention result for each word vector is run through a neural network to integrate the several information sources. This integration network is of the sort described above, where a row of "bulbs" representing "premises" is input and activation passes "forward" through neural network layers until "bulbs" representing "conclusions" are activated – a *feed-forward* neural network. These "conclusions" represent the encoder layer's integrated analysis of each word, performed for each word independently, so that the system's parallelism is never broken.

But again, there will typically be several encoder layers. Multiple *encoder* layers are employed for the same reason as multiple *neuron* ("bulb") layers are used within individual neural networks: abstraction. Earlier layers tend to learn

concrete aspects of the relevant material, those close to the facts, while later ones tend to progressively generalize and address more global aspects. And, in fact, fascinating studies have confirmed this progression: Earliest encoder layers do seem most effective at recognizing part-of-speech tags (noun, verb, etc.), while subsequent layers – as we progress from earliest to latest – seem most efficient for identifying constituents (noun phrases, verb phrases); dependencies (e.g., between a verb and its direct or indirect objects); semantic roles (actor, location, etc.); coreference (pronouns referred to what?); and still more abstract roles. (The "probing" methods for making this determination are themselves of great interest, in view of the pervasive and frustrating opacity of neural networks (Tenney et al., 2019).)

The last layer in the *en*coder layer stack embodies the system's final and most abstracted analysis of the SL input. This can be passed to the earliest *de*coder layer – since decoder layers, like encoder layers, are normally stacked, again for reasons relating to abstraction. For decoders, however, the degree of abstraction progresses from more abstract to more concrete, culminating in the maximally specific decoder layer embodying the TL translation output.

Attention across Languages But how do SL words become TL words? Once again: through attention, in our technical sense. While attention in encoder layers entailed only *self*-attention – the learning of context-worthiness judgments among words *within* the source sequence – decoder layers also exploit such attention judgments *between* source and target word sequences. They indicate, for instance, that, when translating "rabbit" into German in "The rabbit ran because I scared it," we should attend to both "rabbit" and "it," because *source*-language self-attention has earlier found them to refer to the same entity. Both words then influence selection of "Hase" in the context of "Der Hase rannte, weil ich ihn erschreckt hatte."[17] This cross-sequence and cross-language attention is enabled by including a double-strand *encoder-decoder attention element* in each decoder layer, sandwiched between elements we've already encountered in *en*coder layers: a single-strand self-attention element (which analyzes the *TL* sequence on its own terms) and a feed-forward neural network (which integrates the various influences on each word vector) (Figure 1.11).

The decoder layers handle not only word translation – the alignment of source and target words, which will ultimately lead to target word selection – but also target word ordering and agreement (e.g., of nouns and their adjectives). Recall that each target word contains positional and dependency (e.g.,

[17] "The Transformer Neural Network Architecture Explained. 'Attention Is All You Need'." AI Coffee Break with Letitia. URL: www.youtube.com/watch?v=FWFA4DGuzSc&t=438s. July 5, 2020.

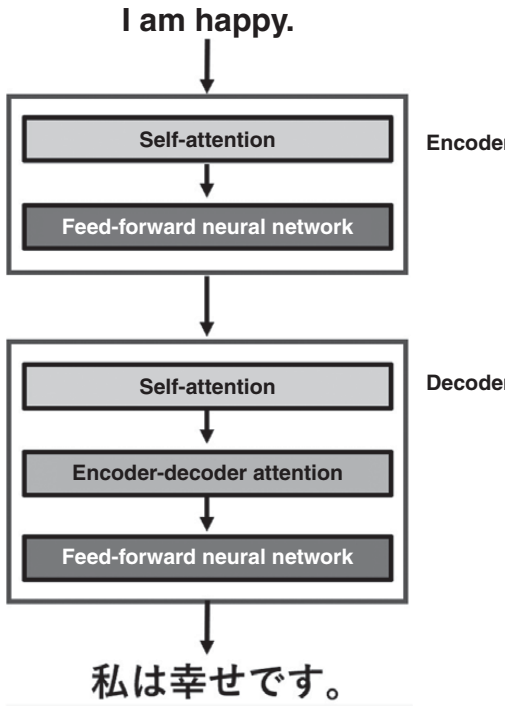

Figure 1.11 Subelements of encoder and decoder layers in a transformer-based MT system

noun-to-adjective) information. In the full TL context, this information will suffice to influence ordering and selection of agreeing dependent words (e.g., forms of an adjective that agree with the associated noun in terms of singular vs. plural, male vs. female or neutral, etc.).

Delivery. The decoder stack's grand finale is the delivery of a TL word sequence. Each fully processed TL word vector, given its place in target-word similarity "space," yields a set of probabilities (a *probability distribution*), assigning each word in the target dictionary a probability score[18]. In our example above, "Hase" might receive a probability in the high nineties as the translation for "rabbit," while an unrelated word like "über" (German for English "over" or "above") would score very low. Once the most probable target word is selected from each set and all target words have found their positions in the sequence

[18] As arranged by a program with a puzzling name: *sofmax*. Its job is to ensure that a handful of probabilities, for example those for possible translations a given TL word, add up to 1.0. Here, if some translation probabilities are very high, others must be low.

according to their internal position indications, translation is complete. There you have it! *Et voilà! Bitte sehr! ¡Y ya está!* で、終わり*!*

(Again, the transformer-based neural sequence-to-sequence processing for speech recognition or speech synthesis will be quite comparable, though operating on sound segments rather than on words or images.)

To give an informal impression of the text translation accuracy achievable at the time of writing, we supply in Appendix II healthcare-oriented translation examples for English-to-Spanish and English-to-Japanese. Each sample is accompanied by a back-translation, enabling English-only readers to estimate the translation accuracy. Of course, back-translation itself is subject to error; but when the error rate is sufficiently small, such feedback remains valuable. Chapter 2 further discusses feedback and its importance.

1.5 Conclusion

As previewed, we've surveyed the methods and issues of several quickly developing technologies relevant to healthcare use cases: ASR, speech synthesis or TTS, and MT. With respect to MT, after a look at systems covering only pretranslated phrases, we went on to explain the major types of automatic translation with broader coverage – "full MT," whether rule-based, statistical, or neural. And finally, as an optional bonus for readers curious about recent developments in the artificial intelligence field, we focused attention (appropriately enough) on transformer-based neural processing.

Also as forecast, we've postponed for Chapter 2 discussion of practical applications for healthcare of speech and translation technologies, with special interest in their combined use for speech translation.

By dispelling the mysteries surrounding these truly epochal technologies, we hope to promote their wider use. However, utilization must also be responsible and cautious. Miscommunications concerning healthcare can be consequential, even deadly. Thus *reliability* – not only measurable accuracy but user confidence – will be essential. *Customization* per use case, too, will be vital, as Chapter 2 will emphasize.

References

Alkhouli, T., A. Guta, and H. Ney. 2014. "Vector Space Models for Phrase-Based Machine Translation." In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar, October 25, pages 1–10.

Boitet, C. 2000. "Bernard Vauquois' Contribution to the Theory and Practice of Building MT Systems: A Historical Perspective." In *Early Years in Machine Translation: Memoirs and Biographies of Pioneers*, William John Hutchins (Ed.), Studies in the History of the Language Sciences 97, pages 331–349. Amsterdam: John Benjamins Publishing.

Brown, P. F., J. Cocke, S. A. Della Pietra, et al. 1990. "A Statistical Approach to Machine Translation." *Computational Linguistics*, 16 (2), June, pages 79–85.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. 1993. "The Mathematics of Statistical Machine Translation: Parameter Estimation." *Computational Linguistics*, 19 (2), June, pages 263–311.

Englert, M., G. Madazio, I. Gielow, J. Lucero, and M. Behlau. 2016. "Perceptual Error Identification of Human and Synthesized Voices." *Journal of Voice*, 30 (5): 639.e17–639.e23. DOI: http://10.1016/j.jvoice.2015.07.017.

Firat, O., K. Cho, and Y. Bengio. 2016. "Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California, June, pages 866–875.

Firth, J. R. 1957. "A Synopsis of Linguistic Theory 1930–1955." In *Studies in Linguistic Analysis*. Oxford: Philological Society, pages 1–32. Reprinted in F. R. Palmer (Ed.), 1968, *Selected Papers of J.R. Firth 1952–1959*. London: Longman.

Gao, Y., Liang G., B. Zhou, et al. 2006. "IBM MASTOR SYSTEM: Multilingual Automatic Speech-to-Speech Translator." In *Proceedings of the First International Workshop on Medical Speech Translation, in Conjunction with NAACL/HLT*. New York, June 9.

Hutchins, W. J. 2005. "Towards a Definition of Example-Based Machine Translation." In *MT Summit X: Proceedings of Workshop on Example-Based Machine Translation*. Phuket, Thailand, pages 63–70.

Hutchins, W. J. 2010. "Machine Translation: A Concise History." *Journal of Translation Studies*, 13 (1–2), *Special Issue: The Teaching of Computer-Aided Translation*, Chan Sin Wai (Ed.), pages 29–70.

Johnson, M., M. Schuster, Q. V. Le, et al. 2016. "Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation." https://arxiv.org/abs/1611.04558.

Koehn, P. 2009. *Statistical Machine Translation*. Cambridge: Cambridge University Press.

Kong, J., J. Kim, and J. Bae. 2021. "HiFi-GAN: Generative Adversarial Networks for Efficient and High-Fidelity Speech Synthesis." arXiv:2010.05646v2. https://arxiv.org/abs/2010.05646.

Kuhn, T. S. 1996. *The Structure of Scientific Revolutions*. 3rd ed. Chicago, IL: University of Chicago Press.

Kurzweil, R. 2016. "Google's New Multilingual Neural Machine Translation System Can Translate between Language Pairs Even Though It Has Never Been Taught to Do So." www.kurzweilai.net/googles-new-multilingual-neural-machine-translation-system-can-translate-between-language-pairs-even-though-it-has-never-been-taught-to-do-so. November 25.

Le, T.-H., J. Niehues, and A. Waibel. 2016. "Toward Multilingual Neural Machine Translation with Universal Encoder and Decoder." In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT) 2016*. Seattle, WA, December 8–9.

Levin, L., D. Gates, A. Lavie, and A. Waibel. 1998. "An Interlingua Based on Domain Actions for Machine Translation of Task-Oriented Dialogues." In *Proceedings of the Fifth International Conference on Spoken Language Processing, ICSLP-98*. Sydney, Australia, November 30–December 4.

Li, J., L. Deng, Y. Gong, and R. Haeb-Umbach. 2014. "An Overview of Noise-Robust Automatic Speech Recognition." *IEEE Transactions: Audio, Speech, and Language Processing*, 22 (4), pages 745–777.

Mikolov, T., Q. V. Le, and I. Sutskever. 2013. "Exploiting Similarities Among Languages for Machine Translation." arXiv preprint.

Ramirez, V. B. 2022. "Metaphor's Going After a Universal Translator. Its AI Now Works for 200 Languages." https://singularityhub.com/2022/07/07/metas-going-after-a-universal-translator-its-ai-now-works-for-200-languages. July 7.

Seligman, M. 1993. *A Japanese–German Transfer Component For ASURA*. ATR Technical Report TR-I-0368.

Seligman, M. 2019. "The Evolving Treatment of Semantics in Machine Translation." In *Advances in Empirical Translation Studies*, Christine (Meng) Ji, (Ed.), Cambridge: Cambridge University Press.

Seligman, M. and A. Waibel. 2018. "Advances in Speech-to-Speech Translation Technologies." In *Advances in Empirical Translation Studies*, Christine (Meng) Ji, (Ed.), Cambridge: Cambridge University Press.

Senellart, J. 2018. "Training Romance Multi-way Model." http://forum.opennmt.net/t/training-romance-multi-way-model/86.

Sotelo, J., S. Mehri, K. Kumar, et al. 2017. "Char2wav: End-to-End Speech Synthesis." International Conference on Learning Representations (ICLR). https://mila.quebec/wp-content/uploads/2017/02/end-end-speech.pdf.

Synced. 2021. "OpenAI Releases GLIDE: A Scaled-Down Text-to-Image Model That Rivals DALL-E Performance." https://syncedreview.com/2021/12/24/deepmind-podracer-tpu-based-rl-frameworks-deliver-exceptional-performance-at-low-cost-173/. December 24.

Tan, X., T. Qin, F. Soong, and T.-Y. Liu. 2021. "A Survey on Neural Speech Synthesis." arXiv: 2106.15561v3 [eess.AS] 23 July. Also in *Computer Science*, June 29, 2021. https://arxiv.org/pdf/2106.15561.pdf.

Tenney, I., D. Das, and E. Pavlick. 2019. "BERT Rediscovers the Classical NLP Pipeline." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, July, pages 4593–4601.

Turney, P. D. and P. Pantel. 2010. "From Frequency to Meaning: Vector Space Models of Semantics." *Journal of Artificial Intelligence Research* 37 (2010), pages 141–188.

Uchida, H. 1986. "Fujitsu Machine Translation System: ATLAS." In *Future Generation Computer Systems*, 2 (2), June, pages 95–100.

Vaswani, A., N. Shazeer, N. Parmar, et al. 2017. "Attention is All You Need." arXiv:1706.03762v5.

Waibel, A. 1987. "Phoneme Recognition Using Time-Delay Neural Networks." *Meeting of the Institute of Electrical, Information, and Communication Engineers (IEICE)*. Tokyo, Japan, December.
Waibel, A., T. Hanazawa, G. Hinton, and K. Shikano. 1987. "Phoneme Recognition Using Time-Delay Neural Networks." ATR Interpreting Telephony Research Laboratories, October 30.
Waibel, A., A. N. Jain, A. E. McNair, et al. 1991. "JANUS: A Speech-to-Speech Translation System Using Connectionist and Symbolic Processing Strategies." In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP) 1991*. Toronto, Canada, May 14–17.
Wiggers, K. 2022. "*The emerging types of language models and why they matter*." https://techcrunch.com/2022/04/28/the-emerging-types-of-language-models-and-why-they-matter/. 5:30 AM PDT, April 28.
Woszczyna, M., M. Broadhead, D. Gates, et al. 1998. "A Modular Approach to Spoken Language Translation for Large Domains." In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas (AMTA)* 98. Langhorne, PA, October 28–31.

Appendix I Automatic Speech Recognition (ASR) Samples

We show two ASR results, for readers' inspection and informal evaluation:

1. iPhone X_R, Software Version 15.5, native (standard) speech recognition
2. Microsoft Windows 10, native (standard) speech recognition

Both results are based upon continuous dictation of the following healthcare-related text, copied without changes from www.cdph.ca.gov/Programs/CID/DCDC/Pages/COVID-19/PregnantandBreastfeedingWomenGuidance.aspx as originally published on May 18, 2021.

Summary

This document provides guidance for people who are pregnant and breastfeeding during the COVID-19 pandemic. The California Department of Public Health will update this guidance as new information becomes available.

Pregnancy: Based on what we know at this time, the Centers for Disease Control and Prevention (CDC) state "pregnant people are at an increased risk for severe illness from COVID-19 and death, compared to non-pregnant people. Additionally, pregnant people with COVID-19 might be at increased risk for other adverse outcomes, such as preterm birth (delivering the baby earlier than 37 weeks). It is especially important for pregnant people, and those who live with them, to protect themselves from getting COVID-19."

Breastfeeding: The Centers for Disease Control and Prevention (CDC) and the Academy of Pediatrics state that parents with COVID-19 can breastfeed. When breastfeeding, precautions should be taken to reduce the risk of passing COVID-19 to their baby.

For more information, visit Breastfeeding and Caring for Newborns.

iPhone Native ASR

Dictated to Google Keep note-taking app. Hand held 10 inches from mouth. Natural but clear pronunciation, with some white noise from refrigerator in neighboring room. Punctuation was dictated, for example as "period," "comma," or "colon," but not capitals or formatting, for example, for **bold** font.

Summary
This document provides guidance for people who are pregnant and breast-feeding during the COVID-19 pandemic. The California Department of Public health will update this guidance as new information becomes available.

Pregnancy: based on what we know this time, the centers for disease control and prevention (CDC) state "pregnant people are at an increased risk for severe illness from COVID-19 and death, compared to non-pregnant people. Additionally, pregnant people with COVID-19 might be at increased risk for other adverse outcomes, such as preterm birth (delivering the baby earlier than 37 weeks). It is especially important for pregnant people, and those who live with them, to protect themselves from getting COVID-19."

Breast-feeding: the centers for disease control and prevention (CDC) and the Academy of pediatrics state did parents with COVID-19 can breast-feed. When breast-feeding, precautions should be taken to reduce the risk of passing COVID-19 to their baby.

For more information visit breast-feeding and caring for newborns.

Microsoft Word on Windows 10

Dictation used standard Microsoft ASR on a Lenovo Yoga 730-13-inch laptop, with input via Microphone Array (Realtek High-Definition Audio (SST)) at 100 percent volume.

Note: Dictation was paused and restarted at two points: after "pregnant and breast-feeding" and "can breast feed." The initial words of the immediately following sentences were apparently missed as a result.

Summary
This document provides guidance for people who are pregnant and breastfeedingWith public health will update this guidance as new information becomes available.

Pregnancy: based on what we know at this time, the Centers for Disease control and prevention (CDC) state "pregnant people are at an increased risk for severe illness from COVID-19 and death, compared to non pregnant people. Additionally, pregnant people with COVID-19 might be at increased risk for other adverse outcomes, such as preterm birth (delivering the baby earlier than 37 weeks). It is especially important for pregnant people, and those who live with them,to protect themselves from getting COVID-19."

Breastfeeding: the Centers for Disease control and prevention (CDC) and the Academy of Pediatrics state that parents with COVID-19 can breast feed. The risk of passing COVID-19 to their baby.

For more information, visit breast feeding and caring for newborns.

Appendix II Translation Samples

Source Language Text to Be Translated

Our text sample for translation was the same as for the speech recognition sampling of Appendix I, copied without changes from https://www.cdph.ca.gov/Programs/CID/DCDC/Pages/COVID-19/PregnantandBreastfeedingWomenGuidance.aspx as originally published on May 18, 2021.

We'll show the following translations and back-translations, all by DeepL (DeepL SE, Cologne, Germany), and all without editing or alterations:

1. Translation into Spanish
2. Back-translation from Spanish to English
3. Translation into Japanese
4. Back-translation from Japanese to English

1. **Spanish Translation**

Resumen

Este documento proporciona orientación para las personas que están embarazadas y amamantando durante la pandemia de COVID-19. El Departamento de Salud Pública de California actualizará esta guía a medida que se disponga de nueva información.

Embarazo: En base a lo que sabemos en este momento, los Centros para el Control y la Prevención de Enfermedades (CDC) afirman que "las personas embarazadas tienen un mayor riesgo de padecer enfermedades graves por COVID-19 y de morir, en comparación con las personas no embarazadas. Además, las personas embarazadas con COVID-19 podrían tener un mayor riesgo de sufrir otros resultados adversos, como el parto prematuro (dar a luz antes de las 37 semanas). Es especialmente importante que las personas embarazadas, y quienes conviven con ellas, se protejan de contraer COVID-19".

Lactancia: Los Centros para el Control y la Prevención de Enfermedades (CDC) y la Academia de Pediatría afirman que los padres con COVID-19 pueden dar el pecho. Al amamantar, se deben tomar precauciones para reducir el riesgo de transmitir COVID-19 a su bebé.

Para obtener más información, visite Lactancia y cuidado de los recién nacidos.

2. **English Back-Translation from Spanish**

Summary

This document provides guidance for persons who are pregnant and breastfeeding during the COVID-19 pandemic. The California Department of Public Health will update this guidance as new information becomes available.

Pregnancy: Based on what we know at this time, the Centers for Disease Control and Prevention (CDC) states that "pregnant persons are at increased risk of severe illness from COVID-19 and death compared to non-pregnant persons. In addition, pregnant persons with COVID-19 may be at increased risk for other adverse outcomes, such as

preterm labor (giving birth before 37 weeks). It is especially important for pregnant women, and those living with them, to protect themselves from contracting COVID-19."

Breastfeeding: The Centers for Disease Control and Prevention (CDC) and the Academy of Pediatrics state that parents with COVID-19 can breastfeed. When breastfeeding, precautions should be taken to reduce the risk of transmitting COVID-19 to your baby.

For more information, visit Breastfeeding and Newborn Care.

3. **Japanese Translation**

概要
この文書は、COVID-19のパンデミック時に妊娠・授乳している人のためのガイダンスを提供するものです。カリフォルニア州公衆衛生局は、新しい情報が入手可能になり次第、このガイダンスを更新します。

妊娠中 現時点でわかっていることに基づいて、米国疾病対策予防センター (CDC) は、「妊娠中の人は、妊娠していない人に比べて、COVID-19による重症化や死亡のリスクが高くなる」と述べています。さらに、COVID-19に感染した妊娠中の人は、早産 (37週より早い出産) など、他の有害事象のリスクが高まる可能性があります。妊娠中の人と、その人と一緒に生活している人は、COVID-19に感染しないように身を守ることが特に重要です。"

母乳育児について 米国疾病対策予防センター (CDC) と小児科学会は、COVID-19を持つ親は母乳で育てることができるとしています。授乳の際には、COVID-19が赤ちゃんに感染するリスクを減らすための予防措置を取る必要があります。

詳細については、母乳育児と新生児の世話をご覧ください。

4. **English Back-Translation from Japanese**

Overview
This document provides guidance for pregnant and lactating women during a COVID-19 pandemic. The California Department of Public Health will update this guidance as new information becomes available.

Pregnancy Based on what is known at this time, the Centers for Disease Control and Prevention (CDC) states that "pregnant individuals are at increased risk of severe illness or death from COVID-19 compared to non-pregnant individuals. In addition, pregnant women infected with COVID-19 may be at increased risk for other adverse events, such as premature delivery (birth earlier than 37 weeks). It is especially important for pregnant women and those living with them to protect themselves from becoming infected with COVID-19."

Breastfeeding The Centers for Disease Control and Prevention (CDC) and the American Academy of Pediatrics state that parents with COVID-19 can breastfeed. When breastfeeding, precautions should be taken to reduce the risk of COVID-19 infecting the baby.

For more information, see Breastfeeding and Caring for Your Newborn.