✔ if You Like the posts You can Follow Me here
Vipul Patel
LinkedIn https://www.linkedin.com/in/vipulppatel/

zusammenlegbar foldable, folding, collapsible ‖kokoonmenevä, -painuva ‖sammanläggbar, -klappbar ‖складываемый, складной
zusammenlegen to fold ‖taivuttaa kokoon, panna ∾ ‖sammanlägga ‖складывать, сгибать
Zusammenlegung *f* ⊕concentration ‖keskitys ‖koncentrering, hopläggning ‖концентрация ж, группировка ж
zusammennageln ⊓to nail ‖naulata yhteen ‖hopspika ‖сбивать гвоздями
zusammennähen ⊖to sew ‖ommella yhteen ‖sy ihop, hopsy ‖сшивать
zusammenpassen ♂to fit ‖sopeuttaa yhteen, sovittaa toisiinsa ‖sammanpassa ‖пригонять, прилаживать между собой, спаривать
zusammenpressen ⊼to compress ‖puristaa kokoon ‖sammanpressa ‖сжимать
zusammenscheren ⊓⌓to join by slit and tongue ‖liittää ruodeliitoksin ‖hop-, sammannåta ‖соединять шпунт заплечиками в гнездо с заруб-кой
zusammenschiebbar sliding ‖kokoonmenevä, sisäjatkoinen ‖samman-, ihop|skjutbar ‖сдвигаемый
zusammenschmelzen ⚒→legieren
zusammenschweißen ▽to weld together ‖hitsata yhteen ‖samman-, hopsvetsa ‖сваривать, соединять сваркой
Zusammensetzarbeit *f* ▽assembling, mounting ‖kokoomistyö, asennus ‖hopsättningsarbete *n* ‖работа ж по сборке, монтажная работа
zusammensetzen to build up, assemble, mount ‖puskupäät ‖sammanstötande ändar ‖сходящиеся концы
Zusammensturz *m* ⌓⚒falling-down, collapsing, tumbling down ‖luhistuminen, sortuminen ‖samman-, instörtande *n*, ras *n* ‖обвал м, оползень м, падение с, обрушение с
zusammenstürzen to collapse ‖luhistua, sortua ‖rasa, sammanstörta ‖рушиться, обрушиться, пасть
zusammenziehen to contract ‖kutistaa, kuroa ‖sammandraga ‖стачивать, суживать, сжимать, стягивать ‖**sich z.** to contract ‖kutistua, kuroutua ‖sammandraga sig ‖сокращаться, суживаться, сжиматься, стягиваться
Zusammenziehmaschine *f* *(fässer)* ♀⊓hoop-driving machine, trussing ∾ ‖kokoonpano-, vannehtimiskone ‖ihopsättningsmaskin ‖стяжной ворот для бочек
Zusammenziehung *f* shrinking, contraction ‖kutistuminen ‖sammandragning, kontraktion ‖стягивание с, сужение, сокращение
Zusammenzinken *n* ⊓dovetailing ‖lohenpyrstöliitos ‖hopsinkning, sinkfog ‖вязание с на шипах, соединение с сковороднем
Zusatz *m* *(hinzufügung)* addition, adjunct ‖lisä(ys) ‖tillsats ‖примесь ж, добавление с, добавка ж, прибавка ж ‖**nach Z. von . . .** ⚌after addition of . . . ‖. . . -a lisättyä ‖efter tillsats av . . . ‖после прибавки . . .
Zusatz- additional, booster, supplementary, accessory, extra ‖lisä-, apu- ‖tillägs- ‖добавочный, вспомогательный
Zusatzartikel *m* additional article ‖lisäke ‖tilläggsartikel ‖добавочная часть, добавочный предмет
Zusatzbatterie *f* ⚡boosting battery ‖lisäkennosto, -paristo ‖tillsatsbatteri *n* ‖вспомогательная батарея
Zusatzgerät *n* attachment, accessory instrument, supplementary apparatus ‖lisälaite ‖tillskottsredskap *n* ‖принадлежность ж, добавочный вспомогательный инструмент, дополнительный прибор
Zusatzmaschine *f* ⊛auxiliary machine ⚡booster ‖lisä-, apukone ‖tillsatsmaskin ‖вспомогательная машина, вольтодобавочная машина
Zusatzstoff *m* ⚌admixing material ‖lisäaine, lisäke ‖tillsats ‖материал м для примеси
Zusatztransformator *m* ⚡→Drehtransformator
zuschärfen ▽to sharpen ‖teroittaa ‖vässa, skärpa ‖заострять
Zuschärfung *f* *(kante)* chamfer ‖viiste ‖tillskärpning ‖заострение с, скашивание с
Zuschauerraum *m* ⌓auditorium ‖katsomo ‖åskådarrum *n*, hörsal, auditorium *n* ‖аудитория ж, зрительное зало
Zuschlag *m* ⚒flux, admixture ‖sulate, lisäaines (sulatuksessa) ‖beskickningsmedel *n*, smältningstillsats, fluss ‖плавень м, флюс м
zuschlagen: kurz z. ▽to strike briskly, to hammer ∾ ‖napauttaa, kopauttaa, takoa lyhyin iskuin ‖slå på, ∾ till, hamra ‖бить, ударять молотком сильными короткими ударами
Zuschläger *m* ▽☺smith's assistant, striker ‖päällelyöjä, pajamies ‖påslåare, påslagare

Machine Translation

Vasily Zubarev

zusammenlegbar foldable, folding, collapsible ‖kokoonmenevä, -painuva ‖sammanläggbar, -klappbar ‖складываемый, складной
zusammenlegen to fold ‖taivuttaa kokoon, panna ∾ ‖sammanlägga ‖складывать, сгибать
Zusammenlegung *f* ⊕concentration ‖keskitys ‖koncentrering, hopläggning ‖концентрация ж, группировка ж
zusammennageln ⊓to nail ‖naulata yhteen ‖hopspika ‖сбивать гвоздями
zusammennähen ⊖to sew ‖ommella yhteen ‖sy ihop, hopsy ‖сшивать
zusammenpassen ♂to fit ‖sopeuttaa yhteen, sovittaa toisiinsa ‖sammanpassa ‖пригонять, прилаживать между собой, спаривать
zusammenpressen ⊼to compress ‖puristaa kokoon ‖sammanpressa ‖сжимать
zusammenscheren ⊓⌓to join by slit and tongue ‖liittää ruodeliitoksin ‖hop-, sammannåta ‖соединять шпунт заплечиками в гнездо с заруб-кой
zusammenschiebbar sliding ‖kokoonmenevä, sisäjatkoinen ‖samman-, ihop|skjutbar ‖сдвигаемый
zusammenschmelzen ⚒→legieren
zusammenschweißen ▽to weld together ‖hitsata yhteen ‖samman-, hopsvetsa ‖сваривать, соединять сваркой
Zusammensetzarbeit *f* ▽assembling, mounting ‖kokoomistyö, asennus ‖hopsättningsarbete *n* ‖работа ж по сборке, монтажная работа
zusammensetzen to build up, assemble, mount ‖puskupäät ‖sammanstötande ändar ‖сходящиеся концы
Zusammensturz *m* ⌓⚒falling-down, collapsing, tumbling down ‖luhistuminen, sortuminen ‖samman-, instörtande *n*, ras *n* ‖обвал м, оползень м, падение с, обрушение с
zusammenstürzen to collapse ‖luhistua, sortua ‖rasa, sammanstörta ‖рушиться, обрушиться, пасть
zusammenziehen to contract ‖kutistaa, kuroa ‖sammandraga ‖стачивать, суживать, сжимать, стягивать ‖**sich z.** to contract ‖kutistua, kuroutua ‖sammandraga sig ‖сокращаться, суживаться, сжиматься, стягиваться
Zusammenziehmaschine *f* *(fässer)* ♀⊓hoop-driving machine, trussing ∾ ‖kokoonpano-, vannehtimiskone ‖ihopsättningsmaskin ‖стяжной ворот для бочек
Zusammenziehung *f* shrinking, contraction ‖kutistuminen ‖sammandragning, kontraktion ‖стягивание с, сужение, сокращение
Zusammenzinken *n* ⊓dovetailing ‖lohenpyrstöliitos ‖hopsinkning, sinkfog ‖вязание с на шипах, соединение с сковороднем
Zusatz *m* *(hinzufügung)* addition, adjunct ‖lisä(ys) ‖tillsats ‖примесь ж, добавление с, добавка ж, прибавка ж ‖**nach Z. von . . .** ⚌after addition of . . . ‖. . . -a lisättyä ‖efter tillsats av . . . ‖после прибавки . . .
Zusatz- additional, booster, supplementary, accessory, extra ‖lisä-, apu- ‖tillägs- ‖добавочный, вспомогательный
Zusatzartikel *m* additional article ‖lisäke ‖tilläggsartikel ‖добавочная часть, добавочный предмет
Zusatzbatterie *f* ⚡boosting battery ‖lisäkennosto, -paristo ‖tillsatsbatteri *n* ‖вспомогательная батарея
Zusatzgerät *n* attachment, accessory instrument, supplementary apparatus ‖lisälaite ‖tillskottsredskap *n* ‖принадлежность ж, добавочный вспомогательный инструмент, дополнительный прибор
Zusatzmaschine *f* ⊛auxiliary machine ⚡booster ‖lisä-, apukone ‖tillsatsmaskin ‖вспомогательная машина, вольтодобавочная машина
Zusatzstoff *m* ⚌admixing material ‖lisäaine, lisäke ‖tillsats ‖материал м для примеси
Zusatztransformator *m* ⚡→Drehtransformator
zuschärfen ▽to sharpen ‖teroittaa ‖vässa, skärpa ‖заострять
Zuschärfung *f* *(kante)* chamfer ‖viiste ‖tillskärpning ‖заострение с, скашивание с
Zuschauerraum *m* ⌓auditorium ‖katsomo ‖åskådarrum *n*, hörsal, auditorium *n* ‖аудитория ж, зрительное зало
Zuschlag *m* ⚒flux, admixture ‖sulate, lisäaines (sulatuksessa) ‖beskickningsmedel *n*, smältningstillsats, fluss ‖плавень м, флюс м
zuschlagen: kurz z. ▽to strike briskly, to hammer ∾ ‖napauttaa, kopauttaa, takoa lyhyin iskuin ‖slå på, ∾ till, hamra ‖бить, ударять молотком сильными короткими ударами
Zuschläger *m* ▽☺smith's assistant, striker ‖päällelyöjä, pajamies ‖påslåare, påslagare

I open Google Translate twice as often as Facebook, and the instant translation of the price tags is not a cyberpunk for me anymore. That's what we call reality. Hard to imagine, that this is the result of a centennial fight to build the algorithms of the machine translation with no visible success during half of that period. Those precise developments set the basis of all modern language processing systems

— from search engines to voice-controlled microwaves. Talking about the evolution and the structure of the online translation today.

The translating machine of P. P. Troyanskii (Illustration made from descriptions. No photos left, obviously)

Story begins in 1933. Soviet scientist Peter Troyanskii presented "the machine for the selection and printing of words when translating

from one language to another" to the Academy of Sciences of the USSR. The invention was super simple — cards in four different languages, typewriter, and an oldschool film camera.

The operator took the first word from the text, found a corresponding card, took a photo and typed its morphological characteristics (noun, plural, genitive) on the typewriter. The typewriter's keys encoded one of the features. The tape and the camera's film used simultaneously, making a set of frames with words and their morphology.

The resulting tape was sent to linguists and turned into a belletristic text. So only native language was required to work with it. The machine brought the "intermediate language" (interlingua) to life for the first time in history, embodying what Leibniz and Descartes had only dreamed of.

Despite all this, as it had always happened in the USSR, the invention was considered "useless". Troyanskii died of Stenocardia after trying to finish it for 20 years. No one in the world knew about the machine until two Soviet scientists found his patents in 1956.

It was at the beginning of the Cold War. On January 7th 1954, at IBM headquarters in New York, the Georgetown–IBM experiment started. IBM 701 computer automatically translated 60 Russian sentences into English for the first time in history. *"A girl who didn't understand a word of the language of the Soviets punched out the Russian messages on IBM cards. The "brain" dashed off its English translations on an automatic printer at the breakneck speed of two and a half lines per second,"* — reported the IBM press release.

IBM 701

However, the triumphant headlines hid one little detail. No one mentioned the translated examples were carefully selected and tested to exclude any ambiguity. For everyday use, that system was no better than a pocket phrasebook. Nevertheless, the Arms race launched; Canada, Germany, France, and especially Japan, all joined the race for machine translation.

The vain struggles to improve machine translation lasted for forty years. In 1966, the US ALPAC committee, in its famous report, called the machine translation expensive, inaccurate and unpromising. They had instead recommended focusing on dictionary development, which eliminated US researchers from the race for almost a decade.

Even so, a basis for the modern Natural Language Processing was created only by the scientists and their attempts, research, and developments. All of today's search engines, spam filters, and personal assistants appeared thanks to the bunch of countries spying on each other.

Rule-based machine translation (RBMT)

The first ideas of rule-based machine translation appeared in the 70s. The scientists peered over the interpreters' work, trying to compel the tremendous sluggish computers to repeat those actions. These systems consisted of:

1. Bilingual dictionary (RU -> EN)
2. A set of linguistic rules for each language (nouns ending in certain suffixes such as -heit, -keit, -ung are feminine)

That's it. If needed, systems could be supplemented with the hacks, such as lists of names, spelling correctors, and transliterators.

PROMPT and Systran are most famous examples of RBMT systems. Just take a look at the Aliexpress to feel the soft breath of this golden age.

But even they had some nuances and subspecies.

[collapse all] [show all] **1** comment

Direct Machine Translation

This is the most straightforward type of machine translation. It divides the text into words, translates them, slightly corrects the morphology, and harmonizes syntax to make the whole thing sound right, more or less. When the sun goes down, trained linguists write the rules for each word.

The output returns some kind of translation. Usually, it's quite shitty. The linguists wasted for nothing.

Modern systems do not use this approach at all. The linguists are grateful.

Transfer-based Machine Translation

In contrast to direct translation, we prepare first — determining the grammatical structure of the sentence, as we were taught at school, and manipulate whole constructions, not words — afterwards. That

helps to get quite decent conversion of the word order in translation. In theory.

It still resulted in verbatim translation and exhausted linguists, in practice. On one side, simplified general grammar rules, and on the other, it became more complicated because of the increased number of word constructions in comparison with single words.

Interlingual Machine Translation

The source text is transformed to the intermediate representation, unified for all the world languages (interlingua). That's the same interlingua Descartes dreamed of: a metalanguage, which follows the universal rules and transforms the translation into a simple "back and forth" task. Next, interlingua converts to any target language and here comes the singularity!

Because of the conversion, Interlingua is often confused with transfer-based systems. The difference is the linguistic rules specific to every single language and interlingua and not the language pairs. This means, we can add a third language to the interlingua system and translate between all three, and can't do the same in transfer-based systems.

It looks perfect, but it's not, in real life. It was extremely hard to create such universal interlingua — a lot of scientists have worked on it their whole life. They've not succeeded, but thanks to them now we have morphological, syntactic and even semantic levels of representation. The only [Meaning-text theory]() costs a fortune!

The idea of intermediate language will be back. Let's wait awhile.

As you can see, all RBMT are dumb and terrifying, and that's the reason they are rarely used unless for specific cases such as the weather report translation, etc. Among the advantages of RBMT often mentioned are morphological accuracy (it doesn't confuse the words), reproducibility of results (all translators get the same result), and ability to tune it to the subject area (to teach the economists or programmers specific terms).

Even if anyone were to succeed in creating an ideal RBMT and the linguists enhanced it with all the spelling rules, there are always some exceptions — all the irregular verbs in English, separable

prefixes in German, suffixes in Russian, and situations when people just say it differently. Any attempt to take into account all the nuances wastes millions of man hours.

Don't forget about homonyms. The same word can have a different meaning in a different context, which leads to a variety of translations. How many meanings can you catch here: *I saw a man on a hill with a telescope.*?

The languages did not develop based on the fixed set of rules, which linguists loved. They were much more influenced by the history of invasions in past three hundred years. How should I explain that to a machine?

Forty years of the Cold War didn't help in finding any distinct solution. RBMT was dead.

Example-based Machine Translation (EBMT)

Japan was especially interested in fighting for the machine translation. There was no Cold War, but there were reasons: very few people in the country knew English. It promised to be quite an issue at the upcoming globalisation party. So the Japanese were extremely motivated to find a working method of machine translation.

Rule-based English-Japanese translation is extremely complicated — language structure is completely different, almost all words have to be rearranged and new ones added. In 1984, Makoto Nagao from Kyoto University came up with the idea of **using ready-made phrases instead of repeated translation**.

Let's imagine, we have to translate a simple sentence — "I'm going to the cinema." We already translated another similar sentence — "I'm going to the theater," and we have the word "cinema" in the dictionary. All we need is to figure out the difference between the two sentences, translate the missing word, and then not fuck it up. The more examples we have, the better the translation.

I'm building phrases in unfamiliar languages exactly the same way!

EBMT showed the light of day to the scientists from all over the world: it turns out, you can just feed the machine with existing translations and not spend years forming rules and exceptions. Not a revolution yet, but clearly the first step towards it. Revolutionary invention of the statistical translation will happen in five years.

Statistical Machine Translation (SMT)

At the turn of 1990, at the IBM Research Center was first shown a machine translation system which knew nothing about the rules and linguistics as a whole. It analyzed similar texts in two languages and tried to understand the patterns.

The idea was simple yet beautiful. An identical sentence in two languages split into words, which matched afterwards. This operation repeated about 500 million times to count, e.g. how many times the word "Das Haus" translated as "house", "building", "construction", etc. If most of the time the source word was translated as "house", we were using this. Note that we did not set any rules nor use any dictionaries — all conclusions were done by machine, guided by stats and the logic "if people translate that way, so will I". And the statistical translation was born.

The method was much more efficient and accurate than all the previous ones. And no linguists were needed. The more texts we use, the better translation we get.

There was still one question left — how the machine correlates the word "Das Haus," and the word "building," and how do we know these are the right ones?

Translations of **leave**

verb

verlassen	leave, abandon, exit, quit, forsake, desert
lassen	let, leave, allow, stop, let go, let be
hinterlassen	leave, leave behind, bequeath
abgeben	leave, dispense, deliver, submit, emit, give off
gehen	go, walk, leave, move, go down, quit
überlassen	leave, entrust, leave up to, leave with, intrust, let have
belassen	leave
abreisen	leave, depart, check out
stehen lassen	let stand, leave, leave behind, leave untouched, abandon
zurücklassen	leave, leave behind
austreten	escape, leave, withdraw, come out, resign, opt out
abfahren	leave, depart, go, ski down, wear, move off
liegen lassen	leave, leave out, leave lying around, leave behind
fortgehen	go away, leave, depart, quit
vergessen	forget, forget about
abgehen	leave, depart, come off, go off, exit, sail
abfliegen	depart, leave, take off, fly off, go
sein lassen	leave, stop, leave alone, drop
ausfahren	extend, come out, go for a ride, deliver
vermachen	bequeath, leave, will, make over
übrig bleiben	remain, leave, be left over
vererben	bequeath, pass on, leave, hand down
fortfahren	continue, proceed, go on, resume, leave, go away
sich entfernen	move away, leave, go away, recede, depart, move off
hinausfahren	drive out, go out, leave

Google's statistical translation from the inside. It shows not only the probabilities but also counts the reverse statistics.

The answer — we don't. At the start, the machine assumes that the word "Das Haus" equally correlates with any word from the translated sentence. Next, when "Das Haus" appears in other sentences, the number of correlations with the "house" increases. That's the "word alignment algorithm," a typical task for university-level machine learning.

The machine needs millions and millions of sentences in two languages to collect the relevant statistics for each word. How do we get them? Well, let's just take the abstracts of the European Parliament and the United Nations Security Council meetings — they're available in the languages of all member countries and now available for download: UN Corpora and Europarl Corpora.

Word-based SMT

In the beginning, the first statistical translation systems worked with splitting the sentence into words, as it was straightforward and logical. IBM's first statistical translation model was called Model one. Quite elegant, right? Guess what they called the second one?

Model 1: "the bag of words"

Model one used a classical approach — to split into words and count stats. The word order wasn't taken into account. The only trick was translating one word into multiple words. E.g. "Der Staubsauger" could turn into "Vacuum Cleaner," but that didn't mean it would turn out vice versa.

Here're some simple implementations in Python: [shawa/IBM-Model-1] (https://github.com/shawa/IBM-Model-1).

Model 2: considering the word order in the sentences

The lack of knowledge about languages' word order has become a problem for Model 1, which is very important in some cases. Model 2 dealt with that; it memorized the usual place the word takes at the output sentence and shuffled the words for the more natural sound at the intermediate step.

Things got better, still shitty tho.

Model 3: extra fertility

New words appear in the translation quite often; such as articles in German or using "do" when negating in English. "Ich will keine Persimonen" → "I **do** not want Persimmons." To deal with it, two more steps added in Model 3.

1. The NULL token insertion, if the machine considers the necessity of a new word;
2. Choosing the right grammatical particle or word for each token.word alignment

Model 4: word alignment

Model 2 considered the word alignment, but knew nothing about the reordering. E.g., adjectives often switch places with the noun, and no matter how good the order is memorized, it won't make the output better. Therefore, Model 4 takes into account so-called "relative order" — the model learns if two words always switch places.

Model 5: bugfixes

Nothing new. Model 5 got some more parameters for the learning and fixed the issue with conflicting word positions.

Despite its revolutionary nature, word-based systems still failed to deal with cases, gender and homonymy. Every single word was translated in a single-true way, according to the machine. Such systems are not used anymore, replaced by the more advanced phrase-based methods.

Phrase-based SMT

The method based on all the word-based translation principles: statistics, reorder, and lexical hacks. Although, for the learning, it split the text not only to words but also phrases. The n-grams, to be precise, which are a contiguous sequence of n words in a row. Thus, the machine learned to translate steady combinations of words, which noticeably improved accuracy.

The trick is, the phrases are not always simple syntax constructions, and the quality of the translation drops significantly if anyone who is aware of linguistics and the sentences' structure interfers. Frederick Jelinek, the pioneer of the computer linguistics, joked about it once: "Every time I fire a linguist, the performance of the speech recognizer goes up."

Besides improving accuracy, the phrase-based translation provided more options in choosing the bilingual texts for learning. For the word-based translation, the exact match of the sources was critical, which excluded any literary or free translation. The phrase-based translation has no problem in learning from them. To improve the translation, researchers even started to parse the news websites in different languages for that purposes.

Starting 2006 everyone started to use this approach. Google Translate, Yandex, Bing, and other high-profile online translators worked as phrase-based right until 2016. Each of you can probably recall the moments when Google either translated the sentence flawlessly or resulted in complete nonsense, right? Phrase-based feature.

The good old rule-based approach consistently provided a predictable though terrible result. The statistical methods were surprising and puzzling. Google Translate turns "three hundred" into "300" without any hesitation. That's called a statistical anomaly.

Phrase-based translation has become so popular, that when you hear "statistical machine translation" that is what is actually meant. Up until 2016, all the studies laud the phrase-based translation as the state-of-art. Back then, no one even thought that Google already kindled its stoves, to change the whole our image of machine translation.

Syntax-based SMT

This method should also be mentioned, briefly. Many years before the emergence of the neural networks, the syntax-based translation was considered as "the future or translation," but the idea did not take off.

The adepts of the syntax-based translation believed in a possibility of merging it with the rule-based method. It's necessary to do quite a precise syntax analysis of the sentence — to determine the subject,

the predicate, other parts of the sentence, and then to build a sentence tree. Using it, the machine learns to convert syntactic units between languages and translate the rest by words or phrases. That would have solved the word alignment issue once and for all.

Example taken from the Yamada and Knight [2001] and this great slide show.

The problem is, the syntactic parsing works like shit, despite that humanity considers it solved a while ago (as we have the ready-made libraries for many languages). I tried to use the syntactic trees for tasks a bit more complicated than to parse the subject and the predicate. And every single time I gave up and used another method.

Write in comments if you succeed using it at least once.

Neural Machine Translation (NMT)

A quite amusing paper on using neural networks in machine translation was published in 2014. The Internet didn't notice it at all, except Google — they took out their shovels and started to dig. Two years later, in November 2016, Google made a game-changing announcement.

The idea was close to transferring the style between photos. Remember the apps like Prisma, which enhanced the pics of some famous artist's style? There was no magic. The neural network was taught to recognize the artist's paintings. Next the last layers containing the network's decision were removed. The resulting stylized picture is just the intermediate image that network got. That's the network's fantasy, and we consider it beautiful.

If we can transfer the style to the photo, what if we try to impose another language to the source text? The text would be that precise "artist's style," and we try to transfer it while keeping the essence of the image (to wit, the essence of the text).

Imagine I'm trying to describe my dog — average size, sharp nose, short tail, always barks. I gave you the set of the dog's features, and if the description is precise, you can draw it, even though you have never seen it.

Now, imagine the source text as the set of specific features. Basically, it means to encode it, and let the other neural network to decode it back to the text, but, in another language. The decoder only knows its language. It has no idea about of the features' origin, but it can express them, e.g., in Spanish. Continuing the analogy, no matter how you draw the dog — with crayons, watercolor or your finger. You paint it as you can.

Once again — one neural network can only encode the sentence to the set of features, and another one can only decode them back to the text. Both have no idea about the each other, and each of them knows only its own language. Recall something? Interlingua is back. Ta-da.

The question is, how to find those features. It's obvious when we're talking about the dog, but how to deal with the text? Thirty years ago scientists already tried to create the universal language code, and it ended in a total failure.

Nevertheless, we have deep learning now. And that's its essential task! The primary distinction between the deep learning and classic neural networks lays precisely in an ability to search for those specific features, without any idea of their nature. If the neural

network is big enough and there are a couple of thousand video cards at hand, it's possible to find those features in the text as well.

Theoretically, we can pass the features gotten from the neural networks to the linguists, so that they open brave new horizons for themselves.

The question is, what type of neural network should be used for encoding and decoding. Convolutional Neural Networks (CNN) fit perfectly for the pictures since it operates with the independent blocks of pixels. But there are no independent blocks in the text; every next word depends on the surroundings. Text, speech, and music are always consistent. So the Recurrent neural networks (RNN) would be the best choice to handle them, since they remember the previous result — the prior word, in our case.

Now RNN is used everywhere — Siri's speech recognition (we're parsing the sequence of sounds, where next depends on the previous), keyboard's tips (memorize the prior, guess the next), music generation, and even chatbots.

For the nerds like me: in fact, the neural translators' architecture varies widely. The regular RNN was used at the beginning, then upgraded to bi-directional, where the translator considered not only words before the source one, but also the next one. That was much more effective. Then it followed with the hardcore multilayer RNN with LSTM-units for the long-term storing of the translation context.

In two years neural networks surpassed everything that appeared in the past 20 years of translation. Neural translation contains 50%

fewer word order mistakes, 17% fewer lexical mistakes and 19% fewer grammar mistakes. The neural networks even learned to harmonize the gender and the case in different languages. No one taught them to do so.

The most noticeable improvements occurred in fields where the direct translation was never used. Statistical machine translation methods always worked using English as the key source. Thus, if you translated from Russian to German, the machine first translated the text to English and then from English to German, which leads to double loss. The neural translation doesn't need that — only a decoder is required so it can work. That was the first time direct translation between languages with no common dictionary became possible.

Google Translate (since 2016)

In 2016 Google turned on the neural translation for nine languages. They developed their system named Google Neural Machine Translation (GNMT). It consists of 8 encoder and 8 decoder layers RNN, as well as attention connections from the decoder network.

They not only divided sentences, but also words. That was their way to deal with one of the major NMT issues — rare words. NMT is helpless when the word is not in their lexicon. Let's say, "Vas3k". I doubt anyone taught the neural network to translate my nickname. In that case, GMNT tries to break words into word pieces and recover the translation of them. Smart.

Hint: Google Translate used for website translation in the browser still uses the old phrase-based algorithm. Somehow, Google isn't upgrading it, and the differences are quite noticeable compared to the online version.

Google uses a crowdsourcing mechanism in the online version. People can choose the version they consider as most correct, and if lots of users like it, Google will always translate this phrase that way and mark it with the special badge. Works fantastically for short everyday phrases such as, "Let's go to the cinema," or, "Waiting for you." Google knows conversational English better than me :(

Microsoft's Bing works exactly like Google Translate. But Yandex is different.

Yandex Translate (since 2017)

Yandex launched its neural translation system in 2017. Its main feature, as declared, was hybridity. Yandex combines neural and statistical approaches to translate the sentence, then choose the best one with their favorite CatBoost algorithm.

The thing is, the neural translation often fails when translating short phrases, since it uses context to choose the right word, and it could be hard if the word appeared very few times in a training data. In such cases, a simple statistical translation finds the right word fast and dumb.

Yandex doesn't tell the details; it fends us off with marketing press-releases. OKAY.

It looks like Google either uses SMT for the translation of words and short phrases. They don't mention that in articles, but it's quite noticeable if you look at the difference between the translation of short and long expressions. Besides, SMT is used for displaying the word's stats, obviously.

The conclusion and the future

Everyone's still excited about the idea of "Babel fish" — the instant speech translation. Google steps towards it with its Pixel Buds, but in fact, it's still not what we were dreaming of. The instant speech translation is different from the usual one. It's necessary to know when to start to translate and when to shut up and listen. I haven't seen suitable approaches to solve this yet. Unless Skype.

And here's one more empty area — all the learning is limited to the set of the parallel text blocks. The deepest neural networks still learn at the parallel texts. We can't teach the neural network without providing it with a source. People, instead, can complement their lexicon with reading books or articles, even if not translating them to their native language.

If people can do it, the neural network can do it too, in theory. I found only one prototype attempting to incite the network, which knows one language, to read the texts in other ones, in order to gain experience. I'd try it myself, but I'm silly. Apparently, that's it.