

**O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI VA
KOMMUNIKATSIYALARNI RIVOJLANTIRISH VAZIRLIGI**

**MUHAMMAD AL XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI NUKUS
FILIALI**

**Telekommunikatsiya texnologiyalari va kasbiy ta'limi fakulteti
«Dasturiy injiniring» kafedrasining «5330600- Dasturiy injiniring»
mutaxassisligining**

4-bosqich talabasi G'afurov Mirzobekning

BITIRUV MALAKAVIY ISHI

**Mavzusi: Tarjimon va lug'at dasturlari modellari va algoritmlari yordamida
qoraqolpoq va o'zbek tillari uchun ochiq kodli dasturiy vosita ishlab chiqish.**

Ilimiy rahbar: _____ prof. Uteuliev N. U.

Kafedra mudiri: _____ prof. Uteuliev N. U.

NUKUS - 2019 y

**O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI VA
KOMMUNIKATSIYALARNI RIVOJLANTIRISH VAZIRLIGI**

**MUHAMMAD AL XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
NUKUS FILIALI**

Telekommunikatsiya texnologiyalari va kasbiy ta'limi fakulteti

«Dasturiy injiniring» kafedrası

TASDIQLAYMAN

Kafedra mudiri _____

«___» _____ 2018 y.

Komiljanov Sardorbek ning

**“Tarjimon va lug’at dasturlari modellari va algoritmlari yordamida
qoraqolpoq va o’zbek tillari uchun ochiq kodli dasturiy vosita ishlab chiqish”
mavzusidagi bitiruv malakaviy ishiga.**

TOPSHIRIQ

**Bitiruv malakaviy ishining mavzusi TATU NF-ning «__» _____ 2018 y.
buyrig’i bilan tasdiqlandi.**

BMI ni topshirish muddati: «__» _____ 2019-yil.

BMI ni o'rinlashga tegishli ma'lumotlar: o'qish materiallari, ma'ruza materiallari, ilmiy adabiyotlar va internet resurs materiallari.

Bitiruv malakaviy ishining tarkibi:

KIRISH

- 1. I-BOB. TARJIMON VA LUG'AT TIZIMLARINI QURISH VA ULARNING ISHLASH PRINTSPLARI**
- 2. II. TARJIMON VA LUG'AT DASTURLARINI ISHLAB CHIQISH UCHUN MODELLARI VA ALGORITMLAR TAHLILI**
- 3. III. AVTOMATLASHTIRILGAN TARJIMA TIZIMI NATIJALARI TAHLILI**

XULOSA

FOYDALANILGAN ADABIYOTLAR

Topshiriq berilgan sana

« ____ » _____ 2018-yil.

Ilimiy rahbar: _____

Topshiriq oldi : _____

Bitiruv malakaviy ishining paragraflari bo'yincha maslahatchilar.

Paragraflar	Ilmiy maslahatchi	Imzo, sana	
		Topshiriqni berdi	Topshiriqni oldi
1	N. Uteuliev		
2	N. Uteuliev		
3	N. Uteuliev		
4	N. Uteuliev		

BMI ni bajarish rejasi:

№	Bob(paragraf) termini	Bajarish muddati	Ilmiy rahbarning imzosi
1.	I-BOB. TARJIMON VA LUG'AT TIZIMLARINI QURISH VA ULARNING ISHLASH PRINTSPLARI	05.02.2019	
2.	II. TARJIMON VA LUG'AT DASTURLARINI ISHLAB CHIQISH UCHUN MODELLARI VA ALGORITMLAR TAHLILI	18.03.2019	
3.	III. AVTOMATLASHTIRILGAN TARJIMA TIZIMI NATIJALARI TAHLILI	13.04.2019	
4.	POWER POINT dasturida prezentatsiya tayyorlash	16.05.2019	

Bitiruvchi: _____

«____»_____ 2019 -yil.

Ilmiy rahbar: _____

MUNDARIJA

KIRISH.....	4
I-BOB. TARJIMON VA LUG'AT TIZIMLARINI QURISH VA ULARNING ISHLASH PRINTSPLARI.....	8
1.1. Tarjimon va lug'at tizimlari haqida umumiy tushunchalar.....	8
1.2. Tarjimon va lug'at tizimlarini qurish va ularning ishlash usullari.....	10
1.3. O'zbek va qoraqalpoq tillarida tarjimon va lug'at dasturlaridan foydalanish muammolari.....	13
1.4. Elektron lug'at dasturlari haqida umumiy tushuncha.....	15
II. TARJIMON VA LUG'AT DASTURLARINI ISHLAB CHIQISH UCHUN MODELLARI VA ALGORITMLAR TAHLILI	26
2.1. Tarjimon va lug'at dasturlari modelari.....	26
2.2. Statistik tarjima qilish modellari va algoritmlari.....	27
2.3. Tarjimon va lug'at dasturlari modelari va algoritmlari ko'magida qoraqalpoq a o'zbek tillari uchun ochiq kodli dasturiy vosita ishlab chiqishda neyron tarmoqlaridan foydalanish.....	31
2.4. Tarjima va lug'at dasturlarida ma'lumotlar bazasi va tashqi ma'lumotlar bilan ishlash.....	38
III. AVTOMATLASHTIRILGAN TARJIMA TIZIMI NATIJALARI TAHLILI.....	41
3.1. Mavjud tarjimon va lug'at tizimlarining afzalliklari va kamchiliklari. Tarjima sifatini baholash usullari.....	41
3.3. Tarjima va lug'at dasturlarini qurishda neyron tarmoqlaridan foydalanish .	46
3.4. Ishlab chiqilgan tarjimon va lug'at tizimi natijalari.....	47
ADABIYOTLAR	53
ILOVA.....	57

KIRISH

O'zbek va qoraqalpoq til juftliklari uchun yetarli parallel ma'lumotlarning kamligi bilan bir qatorda, turkchaga tarjima qilish juda murakkab muammo bo'lib, ularning morfologik murakkabligi va xilma-xilligi bilan bog'liq. Avtomatlashtirilgan tarjima axborot texnologiyalari sohasidagi dolzarb yo'nalishlardan biridir. Katta o'lchamdagi hujjat aylanmasi bilan ish yurituvchi tashkilotlar, ta'lim muassasalari va boshqa sohalar uchun turli maqsadlarda bir tildagi tekstlarni ikkinchi bir tilga ma'nosini yo'qotmagan holda tarjima qilish masalasi yechimini topmay qolmoqda. Misol uchuning dolzarbligi shundaki, to'g'ridan-to'g'ri inson qo'li bilan yozma tarzda yoki to'g'ridan to'g'ri tarjima qilish protsessini amalga oshirish birinchi sababi, ko'p vaqtni talab qiladi va ikkinchi sababi, turli xatolarni keltirib chiqarishi mumkin va tarjima qilish protsessi ma'lum qoida va algoritmlarga bo'ysunmaydi. Shu sababdan bu muammoga aniq yechim topilmagan edi.

Tarjimon tizimlari va lug'at dasturlarini ishlab chiqish protsessida an'anaviy va statistik usullari yaqingacha keng qo'llanilib kelindi. Kelajakda ushbu muammoga sun'iy intellekt tizimlari ko'magida yechim topish imkoniyati paydo bo'ldi. Neyron tarmoq an'anaviy so'z birikmalariga asoslangan tarjima tizimidan tafovutli o'laroq ko'pgina konfiguratsiya qilingan kichik subkomponentlardan tashkil topgan, neyron tarjimasi jumlani o'qiydigan va to'g'ri tarjimini chiqaradigan yagona katta neyron tarmog'ini qurishga qaratilgan.

Har xil hujjatlar bilan ish yurituvchi tashkilotlar, ta'lim muassasalari va boshqa sohalar uchun turli maqsadlarda bir tildagi tekstlarni ikkinchi bir tilga ma'nosini yo'qotmagan holda tarjima qilish masalasi turli munozaralarga sabab bo'lib qolmoqda. Misol uchuning dolzarbligi shundaki, to'g'ridan-to'g'ri inson qo'li bilan yozma tarzda yoki to'g'ridan to'g'ri tarjima qilish protsessini amalga oshirish birinchidan, tarjima qilish protsessi ma'lum qoida va algoritmlarga bo'ysunmaydi va ikkinchidan, turli xatolarni keltirib chiqarishi yoki ko'p vaqtni talab qilishi mumkin. Bu muammoga yaqin yillargacha aniq yechim topilmagan edi.

Shu sababdan sun'iy intellekt tizimlarini avtomatlashtirilgan tarjima sohasida qo'llanila boshlandi va bu sohada katta burilish bo'ldi. Shuningdek sun'iy neyron tarmoqlari tarjima qilish protsessida qo'llanila boshladi.

Tarjimon tizimlari bir qancha muammolarni yechishda foydalanilishi mumkin:

- So'zlar, frazalar va qisqa tekstlarni tez tarjima qilish;
- Ijtimoiy tarmoqlarda, messenjerlarda muloqot protsessida avtomatik tarjima;
- Nashrlar va maqolalarni o'qishda avtomatik tarjima;
- Kitoblarni bosqa tillarga tarjima qilish protsessida

Tarjimon va lug'at dasturlari modelari va algoritmlari bitiruv malakaviy ishining obyektidir. Bitiruv malakaviy ishining predmetini tarjima qilish modellari, *avtomatlashtirilgan tarjima va lug'at tizimlari* algoritmlari, statistik tarjima va tarjimon tizimlari ishlash jarayonlari va usullari tashkil qiladi. Bitiruv malakaviy ishining maqsadi tarjimon va lug'at dasturlari modelari va algoritmlari ko'magida ochiq kodli dasturiy vosita ishlab chiqish.

Bitiruv malakaviy ishining vazifalari.

- Tarjimon va lug'at dasturlari modelari va algoritmlari bilan ishlashda yuz beradigan muammolarning yechimlarini o'rganish va tahlil qilish;
- Tarjimon va lug'at dasturlari modelari va algoritmlarini boshqa tizimlar tarjimasini bilan taqqoslagan holda natijalarni tahlil qilish;
- Tarjimon va lug'at dasturlari modelari va algoritmlari asosida dastur ishlab chiqish;
- Tarjimon va lug'at dasturlarini tadbiq qilish;

Bitiruv malakaviy ishining yangiligi sifatida quyidagilarni keltirib o'tishimiz mumkin:

1. O'zbek-Qoraqalpoq tekst tarjimoni tizimi;
2. Model va algoritmlarga tayangan holda ochiq kodli (open source) Open NMT loyihasi asosida yangi talqindagi tarjimon tizimi faoliyatini boshladi;

3. Ma'lumotlar bazasi o'rnini bosuvchi parallel tekstlarni yig'ish usullari ishlab chiqildi va bit necha turkiy tillardagi parallel tekstlar yig'ildi va o'lchami belgilandi;

Bitiruv malakaviy ishida tekstlarni avtomatlashtirilgan tarjima qilish yondoshuvlari o'rganiladi. Bundan tashqari tekstlarni neyron tarmoq ko'magida tarjima qilish muammolari tahlil qilindi. O'zbek va boshqa turkiy tillar avtomatlashtirilgan tarjima modellari va ularning algoritmlari o'rganildi.

Turkiy tillar avtomatlashtirilgan tarjimasi sifatini yaxshilash uchun, parallel ma'lumotlar yetarlicha bo'lgan, o'zi o'rganuvchi neyron mashinali tarjimon tizimlari modellari asosida ishlab tuzilgan tizimlarni rivojlantirishimiz .

Tekstlarni avtomatlashtirilgan tarjima qilish tarixda kompyuter tarjimasi raqamli bo'lmagan muammolarni yechish uchun dastlab 1954 yilda AQShda mashhur Jorjtaun universitetida tajriba amalga oshirilgan.

O'zbekiston va MDHda avtomatlashtirilgan tarjima tizimlarini ustida Nazirov Sh.A. [8], Grashenko L.A., Klishiniskiy E.S., [6] va Utueliyev N.U. [3, 4, 5] singari olimlar tadqiq qilgan.

Chet ellik olimlaridan Jey Marchiano, Kalchbrenner va Blunsom (2013), Sutskever (2014), Cho (2014) va boshqa tadqiqotchilar tomonidan neyron tarmoqlari ko'magida mashina tarjima qilish tizimlari o'rganilgan .

Neyron tarmog'i texnologiyalaridan Microsoft, Google va Yandex singari kompaniyalar tomonidan tarjima jarayonlarida foydalanish yo'lga qo'yilgan. Google o'z tarjimon tizimini yaratishda o'z neyron tarmoqlarini qurishda rekurrent tarmoqlariga asoslandi va yangi GMNT (Google neyron avtomatlashtirilgan tarjimasi) tizimini ishlab chiqdi. Bu tizim ikkita 8 qatlamli neyron tarmog'idan tashkil topgan bo'lib, kiritilgan tekstni tahlil qiladi va sintezlaydi. Tarjima qilish protsessiga kiritilgan gap oldin (bir vaqtning o'zida chapdan, o'ngdan va qayta) o'qitiladi va ikkinchi neyron tarmoqqa jo'natiladi. Shu usulda ikkinchi tildagi tarjima gap tuziladi.

Bitiruv malakaviy ishi davomida tarjimon tizimi uchun kerak tekstlarni semantik tahlil qilish usullaridan, natijalarning sifatini tekshirish maqsadida ekspert inson tarjimasini bilan taqqoslandi.

Taklif qilingan model va algoritmi ko'p qatlamli neyron tarmoqni tadqiq qilishga imkon beruvchi dasturiy tizim shaklida amalga oshiriladi, shuningdek amaliy masalalarni, xususan, turkiy tillarda kompyuterli tekst tarjimasida uchrayotgan nuqsonlarni yechishga yordam beradi. Bu o'z navbatida turkiy va boshqa rasmiy tillardagi axborotlar ko'lamini orttirishga, mamlakatlar rasmiy tillarining obrosini saqlab qolishiga yordam beradi.

Bitiruv malakaviy ishida tekstlarni avtomatlashtirilgan tarjimani qilish modellari va usullari tahlil qilingan bo'lib, I-bobda tarjimon va lug'at tizimlarini qurish va ularning ishlash printsplari. Ushbu bobning qismlarida avtomatlashtirilgan tarjima teoriyasi asoslari, qoidalarga asoslangan, statistik va neyron avtomatlashtirilgan tarjima yondashuvlari tahlili, neyron avtomatlashtirilgan tarjima qilish arxitekturasi, semantika va kompyuter lingvistikasi muammolari, o'zbek-qoraqalpoq va boshqa turkiy tillarida avtomatlashtirilgan tarjima muammolari o'rganilgan. II-bobda tarjimon va lug'at dasturlarini ishlab chiqish uchun modellari va algoritmlar tahlil qilindi. III-bobda avtomatlashtirilgan tarjima tizimi natijalari tahlil qilindi.

I-BOB. TARJIMON VA LUG'AT TIZIMLARINI QURISH VA ULARNING ISHLASH PRINTSPLARI

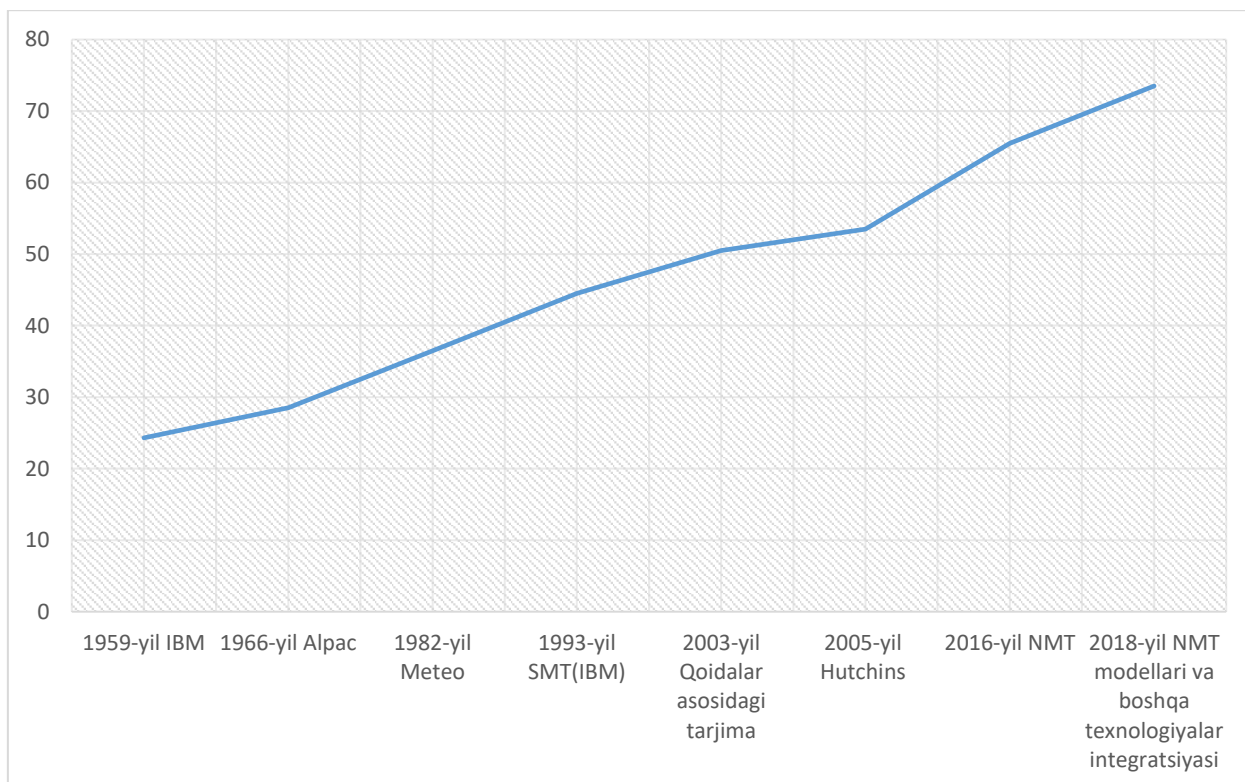
Ushbu bobda tarjimon tizimlari va lug'at dasturlarining ishlash printsplari printsplari tadqiq qilinadi. Avtomatlashtirilgan tarjima usullari va modellari qaraldi. Shuningdek sifatli tarjima masalalari tadqiq qilindi.

1.1. Tarjimon va lug'at tizimlari haqida umumiy tushunchalar

Ma'lumki, tarjima bir tildagi tekstni boshqa tilga o'girishdan iborat bo'lgan adabiy ijod turi. Bu soha azaldan millatlararo aloqalarni ta'minlashda muhim ahamiyatga ega bo'lib kelgan va hozir ham o'z mavqeini yo'qotgani yo'q. Millatlararo muloqotning eng muhim ko'rinishi deb snaladigan tarjimaning bir qancha turlari mavjud. U asliyat va qayta tiklangan tekst xususiyatiga qarab badiiy tarjima, ilmiy tarjima va boshqa turlarga ajratiladi. Asl nusxani aks ettirish xususiyatiga ko'ra tavsif, tadbil, sharh singari ko'rinishlarga ham ega bo'lishi mumkin. Bu soha bo'yicha ko'pchilik o'zbek, qoraqalpoq va xorijiy olimlarning ilmiy ishlarida bir tildagi tekst va so'zlarni boshqa tillarga tarjima qilish masalalari o'rganilgan va tarjima usullari, muammolari, talablari va aspektlari ochib berilgan. Axborot texnologiyalarining rivojlanishi tarjima sohasiga yangi "avtomatlashtirilgan tarjima va lug'at tizimlari" terminini olib kirdi. Avtomatlashtirilgan tarjima va lug'at tizimlari yoxud kompyuterlashgan tarjima bu - kompyuter ko'magida bajariladigan, bir tabiiy tildan boshqasiga maxsus kompyuter dasturi ko'magida tekstlarni (yozma ravishda yoki og'zaki ravishda) tarjima qilish protsessidir. Yana shu singari tizimlarni ishlab chiqishga bog'liq bo'lgan ilmiy-tadqiqotlar yo'nalishi ham shunday nomlanadi. Kompyuterlashgan tarjima, ko'p xollarda avtomatlashtirilgan tarjima va lug'at tizimlari deyiladi, bu esa sun'iy tafakkurga ega bo'lgan amaliy jarayon tomonidan bajariladi [1, 28].

Avtomatlashtirilgan tarjimasining g'oyasi yaqinda paydo bo'lgan bo'lsa ham, tekstni avtomatlashtirishning dastlabki fikrlari XVII-asrga to'g'ri keladi. Biroq,

XX asrga qadar ular amaliyotga qo‘llanilmagan edi. Avtomatlashtirilgan tarjima va lug‘at tizimlari 1959-yillardan boshlab rivojlana boshladi. Mashinali tarjimon tizimlari rivojlanishi bilan tekstlarni mashina ko‘magida tarjima qilish muammollari o‘z yechimini topa boshladi. Hozirgi vaqtgacha tarjimon tizimlarida sifatli tarjima masalasi o‘z yechimini topmayotgan edi. Endilikda sun‘iy intellekt tizimlari va shu paytgacha to‘plangan tajribalar yordami bilan avtomatlashtirilgan tarjimon tizimlariga bo‘lgan nuqtai nazar tubdan o‘zgardi.



1.1-rasm. Mashinali tarjimon tizimlari rivojlanish etaplari

Mashinani tarjima tarjima modellari, usullari va algoritmlarini asosiga qurilgan maxsus kompyuter tizimi (bir qancha dasturlarni va texnik vositalarni) ko‘magida amalga oshiriladi. Bu tarjima qilishning aniq yo‘nalishi bo‘yicha (ma‘lum bir tildan ikkinchisiga) ma‘lum juft tillarda tarjima so‘z va frazalarini topish uchun tekst bo‘yicha tasodifiy va aniq belgilangan harakatlar ketma-ketligini anglatadi. Bunday tizimlar toifasi ikkita asosiy gruppaga bo‘linadi: elektron lug‘at va kompyuterli (*mashinali*) tarjima qilish tizimi [4, 28].

Avtomatlashtirilgan tarjima va lug‘at tizimlarining asosi bo‘lgan tarjima modeli tushunchasini izohlab o‘taylik. Tarjima modeli – bu jadval shaklida

berilgan bo‘lib, unda bir tildagi tizimga oid bo‘lgan barcha so‘zlar va frazalarning boshqa tilga tarjima qilgandagi mavjud ma‘nolari beriladi va shu ma‘nolarning ehtimolligi ko‘rsatiladi. Tarjima modeli foydalanish uch etapda ish bajaradi: oldin parallel hujjatlar keyin ularning ichidan gaplarning juftliklari, undan keyin bo‘lsa so‘z birikmalari yoki so‘zning juftliklari [28].

Hozirgacha har xil hujjatlar bilan ish yurituvchi tashkilotlar, ta‘lim muassasalari va boshqa sohalar uchun turli maqsadlarda bir tildagi tekstlarni ikkinchi bir tilga xatosiz tarjima qilish masalasi turli munozaralarga sabab bo‘lib qolmoqda. Misol uchuning dolzarbligi shundaki, to‘g‘ridan-to‘g‘ri inson qo‘li bilan yozma tarzda yoki to‘g‘ridan to‘g‘ri tarjima qilish protsessini amalga oshirish birinchidan, ko‘p vaqtni talab qiladi va ikkinchidan, ko‘p resurs talab qilinadi.

To‘liq mashinani tarjima qilish protsessi asosiy etaplari quyidagilardan iborat bo‘lishi mumkin:

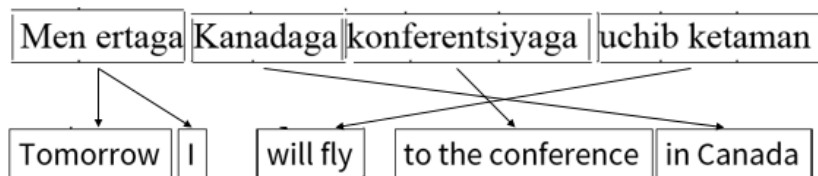
1. Tekstni kirish tilida tahlil qilish (lug‘atda so‘zlarni izlash, morfologik va sintaktik tahlil tekstni tushunish uchun modellashtirilgan);
2. Konvertatsiya (kirish tilidagi tekst tarkibidan chiqish tilidagi tekst tarkibiga o‘tish);
3. Tarjima qilinadigan tilidagi tekstni sintez qilish (tekstning sintaktik va morfologik tuzilishi tekstning qurilishiga o‘xshash).

Tarjimon tizimini yaratishdan maqsad – tarjima qilish protsessining sifatini yaxshilash, avtomatlashtirish va qulayliklar yaratishdan iborat.

1.2. Tarjimon va lug‘at tizimlarini qurish va ularning ishlash usullari

Hozirgacha kompyuter ko‘magida tarjima qilish algoritmlarini tuzishda asosan uchta yondashuvdan foydalanilib kelinmoqda, bular: qoidalarga asoslangan tarjima (*rule-based machine translation*), statistikaga asoslangan tarjima (*statistical-based machine translation*) va neyron tarmoq modellariga asoslangan (*neural machine translation*) yondoshuvlardir. Bu usullarni bir-biriga taqqoslaganda afzalliklari va minus tomonlari ham mavjud. Birinchi usul

an'anaviy bo'lib, bu usulda ishlovchi tizimlar (Promt, SYSTRAN, Linguatic va boshqalar) [7, 10] keng qo'llaniladi. An'anaviy usulda tarjima qilinadigan tekst tizim tomonidan maxsus algoritm ko'magida bir necha etapda tahlil qilinadi va tasdiqlangan tahlil tarjima variantiga biriktiriladi [1].



1.2-rasm. So'zga asoslangan an'anaviy avtomatlashtirilgan tarjima va lug'at tizimlari

Ikkinchi, statistika qonuniyatlariga asoslanib ishlovchi tarjima tizimlariga *Google translate* [17, 16], Yandex tarjimoni [12] va ABBYY [8] xizmatlarining avvalgi (2015-2016-yillargacha foydalangan) versiyalarini keltirib o'tishimiz mumkin. Bu usulning an'anaviy usuldan tafovutli tomoni, bunda tarjima uchun lingvistik algoritmdan foydalanmasdan stilistik bir-biriga mosligini taqqoslash orqali tarjima amalga oshiriladi. Statistika qonuniyatlariga asoslangan avtomatlashtirilgan tarjima va lug'at tizimlarida katta ko'lamdagi til juftliklari ikkinchi tildagi varianti bilan taqqoslanadi. Til juftliklari – bu bir tilga tegishli tekst va gaplarning ularga mos ikkinchi parallel tildagi tarjimasini hisoblanadi. “*Statistika qonuniyatlariga asoslangan avtomatlashtirilgan tarjima va lug'at tizimlari*” tushunchasi ikki tildagi tekstlarni birlashtirishdan olingan ma'lumotlardan foydalanib, gaplarning ehtimolligi yuqori bo'lgan tarjimasini izlashga asoslangan, tarjima qilish muammosining yechimga yo'naltirilgan umumiy yo'nalishi deb bilamiz. Misol uchun, Yevropa iqtisodiy birlashmasining rasmiy hujjatlari 11 tadan ziyod tilda bosib chiqariladi; BMT bo'lsa hujjatlarini bir necha tilda bosib chiqaradi. Bu materiallar statistik avtomatlashtirilgan tarjima va lug'at tizimlari uchun qimmatli resurslardir.

Qoidalarga asoslangan va statistik tarjima usullaridan tafovutli tarzda *neyron tarmoq tarjimasini*, ularning kamchiliklarini hal qilgan holda yuzaga chiqdi. Bilamizki, statistik tarjima tizimlari lingvistikadan deyarli foydalanmaydi, yoki umuman foydalanmaydi. NMT asosan katta hajmdagi parallel tekstlar bilan

ishlaydi. An'anaviy tarjima bo'lsa, ko'pincha lingvistika qoidalariga asoslanadi. Neyron tarmoq atrjimasidan tafovutli o'laroq, qoidalarga asosalangan va statistik tarjima tizimlari sifat natijalari inson tarjimasi sifati singari bo'la olmaydi. Neyron tarmoqlarini tarjima qilish protsessida qo'llash avtomatlashtirilgan tarjima sohasiga katta o'zgarish olib keldi. Neyron tarmoqlar va o'zi o'rganuvchi avtomatlashtirilgan tarjima va lug'at tizimlari algoritmlari ko'magida, tarjima protsessesida neyron tarmoq modellaridan foydalanish imkoniyati paydo bo'ldi.

Tilning xususiyatlari (misol uchun, o'qish tushunchasi) faqatgina gaplashuvchilar soni juda ko'p bo'lganligi va til va umumiy grammatik xususiyatlari (misol uchun, ishlaydigan xotira miqdori) boshqacha ekanligi tafovut qilmaydi [29].

Asosan turkiy tarjimon dasturlarining tarjima qilish algoritimini ishlab chiqishda asosan o'zbek va qoraqalpoq tillarining stilistik va grammatik qonuniyatlariga tayaniladi. Turkiy tillar oilasiga mansub tillardagi tekstlarni tarjima qilishda so'z birikmalari, neologizmlar, sinonimlar va iboralar qiyinchilik tug'diradi. Shuni hisobga olgan xolda dasturda tarjima qilinishi kerak bo'lgan tekst ichidan oldin ko'lami bo'yicha uzunlari, ya'ni eng ko'p elementlisi tartib bo'yicha tanlab olinadi va ikkinchi tilga tarjima qilinadi. Shu tahlilda so'z birikmalarining ma'no buzilishining oldi olinadi.

Tarjimon tizimining eng asosiy qismi til modellari va parallel tekstlar hisoblanadi. Tarjima natijasining sifatli bo'lishi, parallel tekstlar ko'lamiga bog'liq. Shu sababdan parallel tekstlar avtomatik yoki to'g'ridan-to'g'ri to'ldirilib boriladi. Tekstni tarjima qilish etapida, qatorlar bilan ishlashda yana bir kuchli vosita muntazam ifoda algoritmlari bo'lib, ular ko'magida ko'lami va murakkabligidan qat'iy nazar har qanday qatorlar bilan ishlash imkoniyati mavjud. Bu esa tarjima qilish protsessida yuz beradigan stilistik muammolarni yechish imkoniyatini yaratadi. Avtomatlashtirilgan tarjima va lug'at tizimlari tarjima protsessida parallel tekstlar ishlatiladi va dasturdan katta ko'lamdagi qo'shma so'zlar va iboralar bilan ishlash talab qilinadi. Til bazasidagi elementlar sonining oshishi, yuqori tezlik va katta xotirani talab qiladi.

Axborot tizimlari rivojlanishi bilan, tabiiy tilni qayta ishlash singari muammolarga duch keldi. Semantik tahlil potentsial semantikani belgilash uchun mo'ljallangan. Tizim, faqat murojaat bo'lgan shaxs haqida emas, shuningdek, boshqa foydalanuvchilar bilan aloqalari haqida ham xabardor bo'lishi kerak. Tabiiy tilning noaniqligi, ba'zi savollarni izohlashda eng maqbul usullar ko'magida tahlil qilishni talab qiladi. Avtomatlashtirilgan tarjima tizimlarga qo'yiladigan tezlik talabi, tizim uchun moslashtirilgan va tahlil qilish uchun zamonaviy algoritmlarni talab qiladi [29].

Mashinani tarjima qilish muammosini o'rganish, ushbu muammoning turli qismlarini alohida ko'rib chiqish kerak [4].

1.3. O'zbek va qoraqalpoq tillarida tarjimon va lug'at dasturlaridan foydalanish muammolari

Misol uchuning e'tiborga molik tomoni turkiy tillardagi ilmiy, badiiy, publitsistik, rasmiy uslublar ko'lamining taqqosiy tadqiqi yetarlicha amalga oshirilmaganligidadir. Ma'lumki, elektron ma'lumotlar hajmining ko'pligi bo'yicha ingliz va rus tillari muhim o'rinni egallaydi. Tekstlarni chet tillardan turkiy tillarga, turkiy tillardan chet tillarga yoki bir turkiy tildan boshqasiga (*o'zbek-qoraqalpoq, qoraqalpoq-o'zbek*) kompyuter tizimisiz tarjima qilish protsessi ancha murakkab va ko'p vaqt talab qiladigan jarayon. Aniqrog'i, tarjima qilish protsessida so'zlarning semantik tarkibi rang-barangligi, gap bo'laklari tartibining o'zgarishi (*invertsiya*), so'zlar o'rtasidagi polisemiya (*ko'p ma'nolilik*), omonimiya (*shakldoshlik*), troplar (*ma'no ko'chish turlari*), frazeologik birliklar xususiyatlarini hisobga olish lozim [3]. Bitiruv malakaviy ishida so'z, so'z birikmalari, gap va iboralarni bir tildan boshqa tilga sifatli tarjima qilish masalalari ham olib qaraladi va tahlil qilinadi.

Tarjima qilish protsessida gaplarning ma'nosini yo'qotmaslik uchun tizim oldin so'z birikmalari va frazalarni sodda so'zlardan ajrata olishi kerak bo'ladi. Buning uchun dasturdan tarjimani bir-qancha etapda olib borish, ya'ni dastlab

parallel gaplar, so'ngra so'z birikmalari va so'zlarni taqqoslash talab qilinadi. Misol uchun: o'zbekcha "*Yangi shahar vujudga keldi*" gapini qoraqalpoqcha tarjimasi "*Jan'a qala payda boldi*" deb tarjima qilish to'g'ri bo'ladi. Bu gapni so'zma-so'z tarjima qilganda "*Jan'a qala denege keldi*" singari xatolik yuz bergan bo'lar edi. Biroq yevropa tillaridan turkiy tillarga tarjima qilish uchun katta ko'lamdagi tarjima modeli asosidagi bazaga ehtiyoj tug'iladi. Shu joydan ham tarjima modelining qanchalik ahamiyatga ega ekanligini anglash mumkin [3].

Bir-biridan juda uzoq bo'lgan tillar gruppasi uchun to'laqonli tarjimani amalga oshirib bo'lmaydi. Agar tillar gruppasi bir gruppaga tegishli bo'lsa, yuqorida bildirilgan fikr o'z kuchini yo'qotadi. Biz qarayotgan tillar turkiy tillar, ular bir gruppaga tegishli.

Ma'lumki, turkiy tillar uch gruppaga bo'linadi:

- 1) qarluq gruppasi (o'zbek, uyg'ur)
- 2) qipchoq gruppasi (qozoq, tatar, qirg'iz, qoraqalpoq, boshqird va h.k.)
- 3) o'g'uz gruppasi (turk, ozarbayjon, turkman, va h.k.)

Bu joyda eng katta gruppasi qipchoq gruppasi, eng kichik gruppasi qarluq gruppasi

hisoblanadi. Turkiy tillarda gruppasi osti tillar bir-biriga juda ham yaqin. Misol uchun, turk-ozarbayjon, o'zbek-uyg'ur, qozoq-qirg'iz va h.k. Turkiy tillar til darajasida bir-biriga yaqin bo'lgani uchun bir turkiy tildan ikkinchi bir turkiy tilga tarjimani amalga oshirishda, tarjimaning asosiy qismi morfologik va morfosintaktik darajadagi tillarning modeli asosida amalga oshiriladi. Ushbu gipoteza asosida turkiy tillar uchun tarjimaning arxitekturasini yaratish maqsadga muvofiq [8, 9].

O'zbekiston Respublikasi o'quv o'rinlaridagi o'qitiladigan bir necha tillardagi o'quv darsliklar, hujjatlar va ikkinchi tilga tarjima qilingan badiiy asarlar bizga turkiy tillar tarjimon tizimi parallel ma'lumotlar bazasini kengaytirishda keng imkoniyat yaratib beradi [3-5]. Sababi neyron avtomatlashtirilgan tarjima va lug'at tizimlarining ishlash protsessida mashina o'zi o'rganuvchi tarjima va lug'at

algoritmilari uchun kerak parallel korpuslar ya'ni til juftliklarini yig'ish uchun, aynan shu singari manbalar qimmatli resurs hisoblanadi.

1.4. Elektron lug'at dasturlari haqida umumiy tushuncha

Lug'at — muayyan tilda, uning hududiy yoki ijtimoiy lahjasida mavjud bo'lgan, u yoki bu yozuvchi asarlarida uchraydigan so'zlar yig'indisi, leksika; 2) so'zlar (yoki morfemalar, so'z birikmalari, iboralar va boshqalar) muayyan tartibda (alifboli, uyali, mavzuli) joylashtirilgan, tavsiflanuvchi birliklar, ularning kelib chiqishi, ma'nolari, yozilishi, talaffuzi, uslubiy mansubligi, boshqa tillarga tarjimasini haqida ma'lumotlar jamlangan kitob. Elektron lug'at bo'lsa ananaviy lug'atning hisoblash texnikalari imkoniyatlaridan foydalangan holda ishlab chiqilgan yoxud raqamlashtirilgan ko'rinishidir.

Lug'atlar ma'naviy madaniyat sohasida muhim o'rin egallaydi, ularda jamiyatning ma'lum davrda erishgan bilimlari aks ettiradi. Lug'at bir qator ijtimoiy vazifalarni bajaradi: o'quvchiga muayyan hodisa haqida ma'lumot beradi; uni o'z va o'zga tillardagi so'zlar bilan tanishtiradi; tilni, uning lug'at tarkibini takomillashtirish va tartibga solishga yordam beradi. Lug'atlar juda kadim davrlarda pay-do bo'lgan (qarang Leksikografiya); hozirgi davrda ma'lumotaxborot to'plash va uni o'quvchiga yetkazishda muayyan ahamiyatga ega. So'zni tavsiflashdagi bir kancha farqli xususiyatlariga ko'ra, Lug'atlarni asosan 2 turga: ensiklopedik va lingvistik (filologik), lug'atlarga ajratish mumkin. Ensiklopedik lug'at so'zlarning o'zini emas, balki ular orqali bildiriladigan predmet va tushunchalarni izohlaydi, tushuntiradi.

Tarjimon dasturlar - bu matnlarni bir tildan ikkinchi tilga kompyuter yordamida tarjima qilish, orfografik xatolarni aniqlash hamda matnni tahrir qilish imkoniyatini beradi.

Avtomatik tarjima vositalaridan foydalanish.

Avtomatik tarjima dastur vositalarini shartli tarzda ikkita asosiy toifaga bilish mumkin. Birinchi toifa kompyuter lug'atlaridan iborat. Kompyuter lug'atlarining vazifasi oddiy lu'atlar vazifasi bilan bir xil: noma'lum siz mazmunini anglatadi. Kompyuter lug'atlarining afzalligi kerakli siz mazmunini avtomatik izlash va topishning qulayligi va tezligida kirinadi. Avtomatik lug'at, odatda, siz tarjimasini berilgan klavishlar kombinasiyasini bosish orqali sizlarni tarjima qilish imkonini beradi. Lug'at nafaqat sizlar, balki tipik siz birikmalarini sham izida jamlashi mumkin.

Ikkinchi toifaga to'liqmatni avtomatik tarzda tarjima qilishga imkon beruvchi dasturlar kiradi. Ular bir tildagi (xatosiz tuzilgan) matnni qabul qilib, boshqa tildagi matnni beradi. Ish protsessida dastur qamrovli lug'atlar, grammatik qoidalar majmui va dastur nuqtai nazaridagi eng sifatli tarjimini ta'minlovchi boshqa omillardan foydalanadi.

Ushbu vositalardan foydalangan holda dasturlar boshlang'ich matndagi gaplarning grammatik tarkibini tashlil qiladi, so'zlar orasidagi aloqani topadi va jumlaning boshqa tildagi to'g'ri tarjimasini qurishga intiladi. Gap qancha qisqa bilsa, tarjima shuncha to'g'ri chiqishiga imkon yaratiladi. Uzun gaplar va murakkab grammatik gap qurilishlarida tarjima sistemasi yaxshi natijaga olib kelmasligi mumkin.

Hozirda dunyoda ingliz tilidan boshqa tilga va boshqa tildan ingliz tiliga avtomatik tarjima qiluvchi dasturlar keng qo'llanilyapti. Bu ingliz tilining xalqaro muloqot borasida yetakchi rol o'ynayotganligi bilan izoshlanadi. Ingliz tili o'rganish uchun ancha qulay va sodda, ammo uning soddaligi avtomatik tarjima sistemalari uchun kutilmaganda qo'shimcha qiyinchiliklar tug'diradi. Gap shundaki, ingliz tilidagi bir xil yozilgan so'zlar ko'pincha nutqning turli qismlariga tegishli biladi. Bu gapning grammatik tashlilini qiyinlashtiradi va avtomatik tarjimadagi qo'pol xatolarning yuzaga kelishiga olib keladi.

Biz kundalik faoliyatda tarjima programmalarining ko'p turlarini uchratishimiz mumkin.

Hozirgi vaqtda Rossiyada ham, shu jumladan respublikamizda ham quyidagi vositalar keng qo'llanilmoqda:

- STYLUS (PROMT firmasi mahsuloti) - elektron tarjimoni;
- PROPIS (AGAMA firmasi mahsuloti) - elektron lug'ati;
- ORFO, KONTEKST (INFORMATIK firmasi mahsuloti va LINGVO (BIT firmasi mahsuloti) - orfografik xatolarni aniqlovchi vositalar.

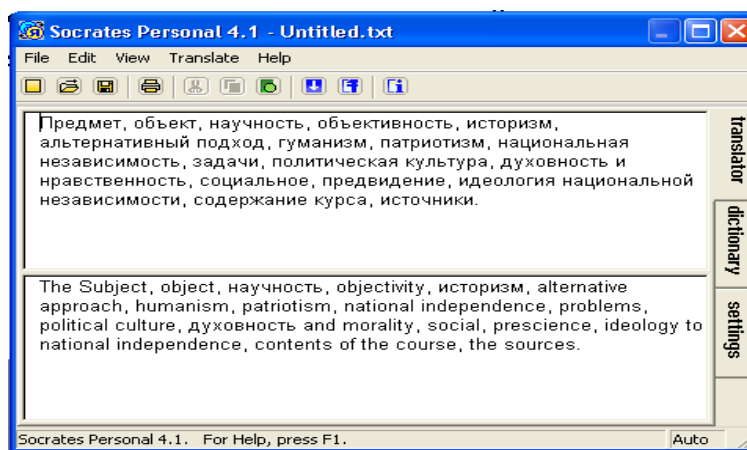
Biroq biz quyidagi eng ko'p tarqalgan va tanilgan Stylus dasturlariga to'xtaymiz.

Rus tilidan ingliz tiliga va ingliz tilidan rus tiliga avtomatik tarjima qilish sistemalaridan Socrat va Stylus kabi dasturlar keng tarqalgan. Stylus, shubshasiz, tarjimada yana ham yuqori sifat va o'zgaruvchanlikni ta'minlaydi. Stylus dasturining so'nggi versiyasi iz nomini o'zgartirdi va u endi Promt deb ataldi.

Prompt dasturi universal, shu bilan birga ixtisoslashgan lug'atlardan iborat boy tarkiblarni iz ichiga olib, ulardan foydalanishni boshqaruvchi vositalarni iz ichiga oladi. Bironta ham lug'atga kirmagan so'zla tarjimasini mustaqil tarzda aniqlab, iste'mol lug'atida saqlaydi. Bundan tashqari, Prompt dasturi xususiy ismlar va tarjima qilish talab etilmaydigan boshqa sizlar, misol uchun, qisqartma so'zlar bilan ishlash qoidalarini ko'rsatish imkonini ham beradi.

Socrat - tarjimon dasturi

Mazkur dastur yordamida faylda joylashgan yoki bevosita kiritilgan matnlarni tarjima qilish mumkin, buning uchun SOCRAT menyusi bo'limlaridan foydalaniladi. Mazkur dastur menyusi bo'limlaridan mohiyati jihatidan WINDOWS ning amaliy dasturlari menyusiga deyarli o'xshash. Farqlanish perevod va clovari bo'limlarida mavjud bo'lib



1.3-rasm. Socrat dasturining umumiy ko'rinishi.

Bu yerda tarjima qilish uchun lug'atlar majmuasi berilgan, undan tarjima qilish usuli tanlanadi. Menyu quyisida joylashgan uskunalar paneli yordamida zaruriy vazifalar tez va o'ng'ay amalgam oshirilishi mumkin.

Vid bo'limidagi informatsionnaya panel mazkur panelni darchaning pastki qismda hosil qiladi. Ma'lumot paneli tarkibiga uchta qo'yilma kiradi va ular "sichqoncha" ning o'ng tugmasini bosish orqali chiqariladi. Ishlatiluvchi lug'atlar tarjima qilishda foydalanishimiz mumkin bo'lgan va kompyuter xotirasidagi mavjud lug'atlar ro'yxatini keltiradi.

Tanish bo'lmagan so'zlar qo'yilmasi mavjud lug'atlarda aniqlanmagan noma'lum so'zlar ro'yxatini beradi. Bu so'zlarni tarjima qilmasdan zaxirada saqlash uchun zarezerovanniye slova qo'yilmasiga o'tqazildi.

Zarezervirovanniye slova qo'yilmasida tarjima qilinishi zarur bo'lmagan (misol uchun WINDOWS) so'zlar ro'yxati saqlanadi.

Hujjatni tarjima qilish

1. Fayl menyusida ochish bandini tanlang. Tarjima uchun hujjat turli xil formatlarda berilishi mumkin va ular kompyuter uchun qulay formatlarda o'tkaziladi .
2. Ochish muloqotli darchaning matnlar sohasidan kerakli faylni tanlab oling va Ochish tugmasini bosing.

3. Konvertirovat fayl muloqotli darchasida tarjima qilish yo'nalishi aniqlanadi va boshlang'ich matn ekranda hosil bo'ladi.
4. Hujjat tarjima qilish uchun tayyor. Perevod bo'limidan , Ves tekst bandi faollashtirilgandan so'ng ekranda o'ng darchada boshlang'ich hujjat chap darchada esa uning tarjimasi bo'ladi.
5. Fayl bo'limida saxranit bandi orqali tarjima biror nom ostida saqlab qo'yiladi. Natijada tarjima qilingan hujjat kerakli formatda hosil bo'ladi va uni biror matn muharririda qayta ishlash mumkin

Yuqorida keltirilgan yordamchi dasturlardan tashqari foydalanuvchining shaxsiy bloknotini tashkil qilish, tadbirkorlarning vaqtini to'g'ri taqsimlashga yordam beruvchi, ma'ruzalar uchun slaydlar hosil qiluvchi, musiqa eshitish dasturlari mavjud. Har bir foydalanuvchi o'z oldiga qo'ygan vazifalardan kelib chiqib, bu dasturlar bilan alohida tanishib chiqishi mumkin.

Stylus – tarjimon dasturi

Mazkur tarjimon dastur yordamida faylda joylashgan yoki bevosita kiritilgan matnlarni tarjima qilish mumkin, buning uchun Stylus menyusi bo'limlaridan foydalaniladi.

Mazkur dastur menyusi bo'limlari mohiyati jihatdan Windowsning amaliy dasturi menyusiga deyal o'xshash. Farqlanish Перевод va Словари bo'limlarida mavjud bo'lib, bu bu yerda tarjima qilish uchun lig`atlar majmuasi berilgan, undan tarjima qilish usuli tanlanadi.

Menyu quyisida joylashgan uskunalar paneli yordamida zaruriy vazifalar tez va o'ng`ay amalga oshirish mumkin.

Stylus dasturi

Вид bo'limidagi Информационная панель asosiy panel, Tarjima paneli Formatlash panelini darchaning pastki qismda hosil qiladi.

Ma'lumot paneli tarkibiga uchta qo'yilma kiradi va ular "sichqoncha" ning o'ng tugmasini bosish orqali chiqariladi.

Используемые словари (Ishlatiluvchi lug'atlar) tarjima qilishda foydalanishimiz mumkin bo'lgan va kompyuter xotirasidagi mavjud lug'atlar ro'yxatini keltiradi.

Незнакомые слова qo'yilmasi mavjud lug'atlarda aniqlanmagan noma'lum so'zlar ro'yxatini beradi. Bu so'zlarni tarjima qilmasdan zahirada saqlash uchun Зарезервованные слова qo'yilmasiga o'tkaziladi. Bu qo'yilmada tarjima qilinishi zarur bo'lmagan (misol uchun, Windows) so'zlar ro'yxati saqlandi.

Hujjatni tarjima qilish.

1. Файл menyusining Открыть bandi tanlanadi. Tarjima uchun hujjat turli xil formatlarda berilishi mumkin va kompyuter uchun qulay formatlarga o'tkaziladi.

2. Открыть muloqatli darjasining matnlar sohasidan kerakli faylni tanlab olib Открыть tugmasi bosiladi.

3. Конвертировать файл Открыть muloqatli darjasida tarjima qilish yo'nalishi aniqlanadi va boshlang'ich matn ekranda hosil bo'ladi.

4. Hujjat tarjima qilish uchun tayyor bo'lgacha Перевод bo'limidan Весь текст bandi faollashtiriladi. So'ng ekranning o'ng darchasida boshlang'ich hujjat, chap darchada esa uning tarjimasi hosil bo'ladi.

5. Файл bo'limida Сохранить bandi orqali tarjima biror nom ostida saqlab qo'yiladi. Natijada tarjiam qilingan hujjat kerakli formatda hosil bo'ladi va uni birbor matn muharririda qayta ishlash mumkin. Bundan tashqari foydalanuvchining shaxsiy blaknotini tashkil qilish, tadbirkorlarning vaqtini to'g'ri taqsimlashda yordam beruvchi, ma'ruzalar uchun slaydlar hosil qiluvchi, musiqa eshitish dasturlari mavjud.

PROMT tarjimon dasturining imkoniyatlari.

Prompt sistemasini universal, shu bilan birga ixtisoslashgan lug`atlardan iborat boy tarkiblarini o`z ichiga olib, ulardan foydalanish boshqaruvchi vositalarini o`z ichiga oladi. Bironta ham lug`atga kirmagan sozlar tarjimasini mustaqil tarzda aniqlab, iste'mol lug`atida saqlaydi. Bundan tashqari, Prompt programmami hususiy ismlar va tarjima qilish talab etilmaydigan boshqa so`zlar, misol uchun, qisqartma so`zlar bilan ishlash qoidalarini ko`rsatish imkonini ham beradi.

Programmaning qo`shimcha imkonyatlari fayllarning turkum tarjimalari, tekislanmagan matnlarning tezkor tarjimalari, shuningdek, Internetdagi Web sahifalarini sinxron tarjima qilish imkonini o`z ichiga oladi. Ushbu vositalar alohida ilova programmalar sifatida amalda tatbiq etilgan.

Prompt programmasining ishchi oynalari

Prompt programmasi o`rnatilgandan so`ng bosh menyuda uni ishga tushirishga izn beruvchi punktlar paydo bo`ladi.

Indikatsiya panelida (vazifalar panelining o`ng chetida) programmani tezlikda ishga tushiruvchi belgi o`rnatiladi. Sichqonchaning o`ng klavishasini ushbu belgi ustida bosish bilan sistemaning barcha ilovalarini ishga tushirishga imkon beruvchi menyu ochiladi. Asosiy programma Prompt punktini tanlanganda ishga tushiriladi.

Prompt interfeysi Windows 9x (bunda X-95, 97, 98, 2000 larni bildiradi) operatsion sistemasi talabalariga muvofiq amalga tatbiq etilgan. Programmani ishga tushirgandan so`ng ekranda menyu satri, asbob panellari satri va ishchi sohasidan iborat ilova oynasi ochiladi.

Lug`atlar bilan ishlash

Avtomatik tarjimaning sifati qanday lug`atdan foydalanilayotganligiga bog`liq. Prompt sistemasi umum iste'moldagi so`zlardan iborat bo`lgan bosh lug`at, shuningdek turli sohaning ixtisoslashgan lug`atlarini o`z ichiga oladi. Ixtisoslashgan lug`atlarga bo`lgan zaruriyat inson faoliyatining turli jabhalarida

ushbu sohaga tegishli tushunchalarni iodalovchi turli terminal qo'llanilishi bilan bog'liq. Bu terminallardan ba'zilar o'ziga xos ma'noga ega bo'lishi, bazilari kundalik turmushda, ba'zida boshqacha ma'noda ishlatilishi mumkin. Maxsus matnlar tarjimasida terminal aqat muvofiq keluvchi mazmunda ishlatilishi kerak.

Misol uchun inglizcha box so'zini olamiz. Universal lug'atlar uni yashik (quti) deb tarjima qilishlari mumkin. ammo programma ta'minoti bilan bog'liq matnlarda bu so'z muloqot oynasi ma'nosini anglatadi.

Tarjima sifatini oshirishning boshqa usuli programmaga u tarjima qilsa olmaydigan yoki tarjima qilmasligi kerak bo'lgan ba'zi so'zlarni ishlatish usulini ko'rsatishdan iborat. Buning uchun tarjima qilmaydigan so'zlar (misol uchun, Windows) ni bandlab qo'yish va programma lig'atida bo'lmagan so'zlar tarjimasini qoidalarni berish zarur.

Lug'atni tanlash

Foydalanilayotgan lug'atlar ro'yhati axborot panelidagi foydalaniladigan lug'atlar qo'shimcha varaqasida keltirilgan. Lug'atlar ko'rsatilgan tartibda ko'riladi, shu bilan birga keyingi lug'atga o'tish ko'rilayotgan lug'atda kerakli soz bo'lmagan taqdirdagina amalga oshiriladi. Tarjima sifati nafaqat lug'atlarning soni, balki ularning programmada terilish tartibiga ham ta'sir ko'rsatadi.

Shunga alohida e'tibor berish kerakki, programmadagi terilishda, odatda birinchi bo'lib iste'mol lug'ati turadi. Iste'mol lig'atlari tahrir qilish va o'zgartirish uchun ochiq bo'ladi. Shu tariqa, foydalanuvchi tomonidan to'ldirilgan va o'zgartirilgan maqolalar birinchi navbatda e'tiborga olinadi.

So'zlarni band qilish

Idoradagi ishlarni avtomatlashtirish. Ko'pgina hujjatlar faqat "qisman" tarjimani talab qiladi. Atama so'zlar va tushunchalar asliyat tilida qolishi kerak.

Misol uchun, kompyuterga doir adabyotlarda programma, operatsion sistemalar va ishlab chiqaruvchi kompaniyalari nomlarini tarjima qilish yoki ularni o'zgartirish maqbul emas.

Avtomatik tarjima sistemasi ma'lum soʻz yo soʻz birikmasini tarjima qilmasligi uchun ushbu soʻzni band qilish darkor. Band qilingan soʻzlar axborot panelidagi Band qilingan soʻzlar qoʻshimcha varaqchasidagi roʻyxatda sanab oʻtiladi. Hujjat matnida band qilingan soʻz uchrashi bilan u boshqa rangda ajralib koʻrinadi.

Band qilingan soʻzlarning yagona lugʻatidan foydalanish imkoniyati bir mavzuga aloqador bir guruh hujjatlar bilan yoki bitta katta hujjatning koʻplab qismlari bilan ishlashda nihoyatda qulaydir.

Lugʻatlarni toʻldirish va sozlash

Prompt programmasi tarkibiga kiritilgan lugʻatlar ancha boy ekanligiga qaramay, hujjatlarda uchrovchi barcha soʻzlar kiritilganligini kafolatlab boʻlmaydi. Notanish soʻzlarni programma qizil rangda ajratib koʻrsatadi.

Ammo barcha notanish soʻzlar ham lugʻatga kiravermaydi. Ular orasida band qilish lozim boʻlgan soʻzlar ham uchrashi mumkin. Shuningdek bu soʻzlar toʻgʻri yozganligini tekshirib koʻrish kerak. Agar lugʻat haqiqatdan ham toʻliq boʻlmasa, unda soʻzni isteʼmol lugʻatiga qoʻshish mumkin.

Lugʻatni malakali tarzda toʻldirish juda muhim va masʼuliyatli tadbir. Lugʻatning haddan ziyod zichligi tarjima sifatini pasaytirishi mumkin. Shuningdek bir soʻz turlicha maʼno koʻrinishlariga ega boʻlishi mumkinligini ham nazarda tutish kerak.

Prompt programmasi soʻzni lugʻatga qoʻshishda ikki: boshlangʻicha va mutaxassis rejmini koʻzda tutadi. Birinchi rejimda soʻzning yetishmaydigan barcha Grammatik shakllari avtomatik tarzda qoʻshiladi, ammo ular doim ham toʻgʻri boʻlavermaydi. Ikkinchi rejimda foydalanuvchining oʻzi barcha gramatik shakllarni beradi, ammo bu ikkala til gramatikasini yaxshi bilishni talab qiladi.

Prompt programmasining qolgan sozlovlari

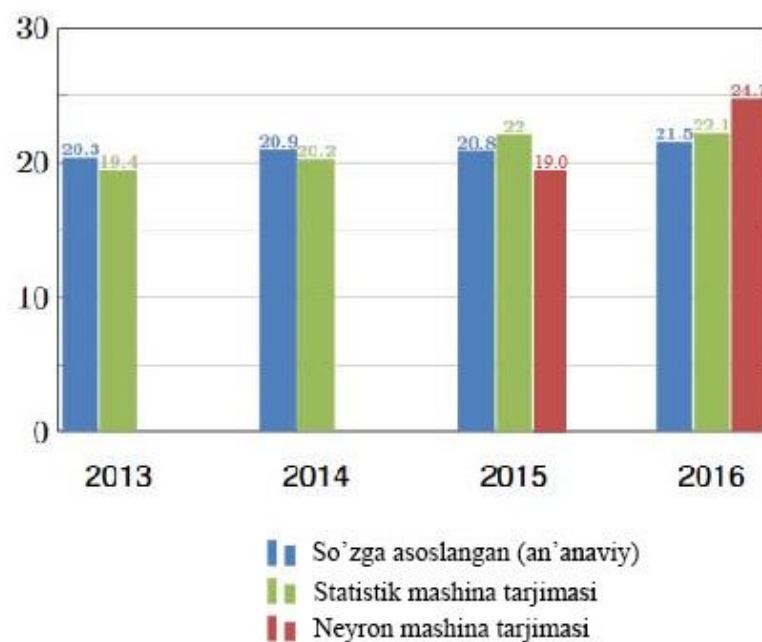
Avtomatik tarjima sistemasi ishining samarasi va sifati, asosan unda mavjud lug`atlar takibi va ularning sifatiga bog`liq. Lug`atlarning siati esa ular qanday tartib bilan to`ldirganiga qarab belgilanadi.

Elektron lug`atlarni ishlab chiqish ananaviy lug`atlardan foydalanishdagi noqulayliklar va nuqsonlarini o`rnini to`ldirib yangi imkoniyatlar yaratib beradi.

BOBGA OID XULOSA

I-bob avtomatlashtirilgan tarjima asoslari va arxitekturasini tahlil qilishga qaratilgan bo`lib, fasllarda avtomatlashtirilgan tarjima teoriyasi asoslari, qoidalarga asoslangan. Shu bilan birga statistik va neyron avtomatlashtirilgan tarjima yondashuvlar, neyron avtomatlashtirilgan tarjima qilish arxitekturasini, ko`p tilli neyron mashinasi tarjimasi, semantika va kompyuter lingvistikasi muammolari, lingvistikaga zid muammolari, o`zbek, qoraqalpoq va boshqa turkiy tillarida avtomatlashtirilgan tarjima muammolari tahlil qilindi.

Olimlarning ushbu yo`nalishda olib izlanishlarini tahlil qilish natijasida, faqat qoidalarga asoslangan va statistik tarjima yondashuvlari bilan yuqori sifatli tarjimani amalga oshirib bo`lmasligi ayon bo`ldi. Sababi parallel tekstlarni qayta ishlash, kata hajmdagi ma`lumotlarni va tekstlarni intellektual tahlil qilish masalalari yuzaga keldi.



1.10 - rasm. An'anaviy, statistik va neyron tarmoq tarjimasi ko'rsatkichlari

Bu o'z navbatida bizga sun'iy intellekt tizimlari jumladan o'zi o'rganuvchi va sun'iy neyron tarmoqlari algoritmlaridan foydalanishni talab qilmoqda.

Neyron avtomatlashtirilgan tarjima va lug'at tizimlari 2015-2016 yildan boshlab joriy qilina boshlandi. Avtomatlashtirilgan tarjima yondashuvlarini tahlil qilib neyron mashina tarjimasi tizimlari sifat ko'rsatkichlari (*1.10 - rasm*), an'anaviy va statistik tarjima bilan taqqoslaganda qisqa vaqtda sezilarli darajada o'sganiga govoh bo'lishimiz mumkin.

II. TARJIMON VA LUG'AT DASTURLARINI ISHLAB CHIQISH UCHUN MODELLARI VA ALGORITMLAR TAHLILI

Ushbu bobda avtomatlashtirilgan tarjima qilish modellari va algoritmlari tahlil qilinadi. Shuningdek, turkiy tillar tekstlarni avtomatlashtirilgan tarjima qilish modellari, usullarini ishlab chiqish muammolari tahlil qilinadi. Turkiy tillar avtomatlashirilgan tarjimon tizimlari algoritmlari ishlab chiqish masalalari o'rganiladi. Qismlarda neyron avtomatlashtirilgan tarjima va lug'at tizimlari modellari muammolari o'rganilgan.

2.1. Tarjimon va lug'at dasturlari modelari

Tarjimaning modeli – butun yoki uning bir qismi ustida tarjima qilish protsessi amalga oshirilishi mumkin bo'lgan bir qator operatsiyalarning shartli ta'rifi yoki tarjima qilish protsessini amalga oshirish protsessining shartli ko'rinishi, tarjima qilish protsessining asosiy etaplarini aks ettiruvchi va har bir etapda tarjimonning xulq-atvorini aks ettiruvchi asosiy tarkib deb sanaladi.

XX-asrning ikkinchi yarmida tarjima teoriyasidagi asosiy muammolardan biri sifatida paydo bo'ldi. Tarjima modeli tarjima qilish protsessining o'ziga xos xususiyati hisoblanib, original tekstni tahlil qilish va yangi tekstni yaratish protsessidagi tarjimonning harakatlari aks etgan. Tarjima amaliyoti uchun bunday tarjima modellarining rivojlanishining ahamiyati shundaki, model tarjimonga qiyin tarjima vazifalarini hal qilishga yordam berishi mumkin deb sanaladi. Shu bilan birga, tadqiqotchilar uchun etiborlisi, joriy tildan tarjima tiliga o'tish etapi va tarjimaning tekstini tuzish yondashuvi va undan keyingi foydalanishdan iborat edi.

Tarjima qilish protsessini tavsiflash muammosi dolzarbligicha qolmoqda, chunki barcha harakatlar tarjimonning ongida amalga oshiriladi va to'g'ridan-to'g'ri monitoring qilinmaydi. Tarjima - bu asl va tarjima tekstlarini solishtirish demakdir. Tarjima modeli ikkita jihatni hisobga olgan holda tasvirlanishi mumkin: 1) modelning umumiy tavsifi, uni qo'llashning mumkin bo'lgan ko'lamini ko'rsatuvchi; 2) ushbu model doirasidagi tarjima operatsiyalarining turlari.

Tarjima jarayonlarini modellashtirish protsessi tarjimani shakllantiruvchi uchta qismdan iborat: dastlabki kod yaratuvchisi, tarjimon (retseptor), joriy tekst o'giruvchi va tarjima qilingan tekstning o'quvchi bo'lgan tarjimon.

Odatda tarjima modeli quyidagi ikki yo'nalishdan biriga yo'naltirilgan bo'ladi: 1) ekstralingvistik qonuniyatlar yoki 2) semantik tuzilish xususiyatlari.

Bugungi kunda ko'plab tarjima modellari mavjud bo'lib, ular orasida eng keng tarqalgan: tahliliy, psixolinguistik, semantik, semiotik, sinergetik, aylantiruvchi, aylantiruvchi-semantik, ekvivalent, shuningdek, vositachilik tarjimasi modellari.

Avtomatlashtirilgan tarjima va lug'at tizimlari singari murakkab tabiiy tillarni qayta ishlash uchun sun'iy aql va neyron tarmoqlardan foydalanish juda tez rivojlanishga olib keladi.

2.2. Statistik tarjima qilish modellari va algoritmlari

Statistik mashinani tarjima qilish - bu katta hajmli til juftliklarini taqqoslash asosida tekstlarni tarjima qiluvchi yondashuv yuqorida I-bobda batafsil ko'rsatilgan.

Til juftligi - bitta tilda gaplarni o'z ichiga olgan tekstlar va ularning ikkinchi tildagi aynan unga mos ikkinchi variantlari bo'lgan tekstlardir. Statistik avtomatlashtirilgan tarjima va lug'at tizimlari «o'zi o'rganish» xususiyatiga ega bo'lib, agar ko'proq til juftligi mavjud bo'lsa va ular bir-biriga mos kelsa, statistik avtomatlashtirilgan tarjima va lug'at tizimlarining natijasi shunchalik yaxshi bo'ladi [1, 31].

Statistik yondashuvda tarjima muammoni shovqinli kanal nuqtai nazaridan ko'rib chiqiladi. Misol tariqasida ingliz tilidan o'zbek tiliga tarjima qilishni ko'rib chiqamiz.

Tarjima qilish protsessi, o'zbekcha jumlaning ingliz tilidagi va ingliz tilidagi jumlaning o'zbek tilidagi ehtimoligini, yoki o'zbekcha jumlagam mos variantlarni maksimal darajada oshiruvchi o'zbekcha jumlaning izlash orqali amalga

oshiriladi. Bayes teoremasiga ko'ra, quyidagi model o'zbekcha jumlaning ingliz tilidagi tarjima qilishning yaxshi variant deb e'tirof etiladi:

$$\max_e P(e|f) = \max_e P(e) * P(f|e)$$

Bu joyda e – tarjima taklifi va f – original taklif.

Shunday qilib, bizga *manba modeli* va *kanal modellari*, yoki *til modellari* va *tarjima modellari* kerak bo'ladi. Til modeli, tugallanuvchi 2-tilning har qanday jumlasiga (bizning holimiz rus tilida) ehtimollik kiritishini belgilashi kerak va tarjima modelida yakuniy tilda ma'lum bir jumla sharti bilan original jumlagamalik ehtimolligi baholashi kerak.

Umuman, mashinani tarjima qilish tizimi ikki rejimda ishlaydi:

Tizimning o'qitilishi: parallel tekstlarni tayyorlash korpusi olinadi va chiziqli dasturlash ko'magida tanlangan tarjima modeliga muvofiq ishning o'zbek tili bo'limida mavjud bo'lgan ingliz tili bilan ehtimolligini (misol uchun eng yuqori darajaga olib chiqadigan tarjima jadvali jadvallarining qiymatlari) so'raladi. Xuddi shu korpusning o'zbek tili qismida o'zbekcha modeli yaratiladi.

Ekspluatatsiya: olingan ma'lumotlarga asosan inglizcha tanish bo'lmagan so'z qidiriladi, o'zbek tilining til modellari va tarjima modellari tomonidan berilgan ehtimolliklar tarjimani maksimal yaqinlik darajasini ishlab chiqarishga intiladi. Bunday qidiruv uchun ishlatiladigan dastur dekoder deb ataladi.

Tabiiy tillarni qayta ishlash sohasida N gramm asosan probabilistik (taxminiy) modellarga asoslangan prognozlash uchun ishlatiladi. N -grammli model, avvalgi barcha ma'lumotlar ma'lum bo'lsa, N -grammning oxirgi so'zining ehtimolligini hisoblaydi. Tilni modellashtirish uchun ushbu yondashuvdan foydalanib, har bir so'zning ko'rinishi avvalgi so'zlarga bog'liq deb taxmin qilinadi [3, 4].

Statistik tarjimada har bir N -gramm ehtimoli uning o'rganuvchi korpusda paydo bo'lishi bilan aniqlanadi. Misol uchun, *trigram* modeli yassi tekislash bilan, x va y so'zlaridan keyin tekstda kiritilgan har bir so'zning grammatik ehtimolligini quyidagi formula bo'yicha baholaydi:

$$b(z | x y) = 0,95 * \text{chastota}(\langle\langle xyz \rangle\rangle) / \text{chastota}(\langle\langle xy \rangle\rangle) + \\ 0,04 * \text{chastota}(\langle\langle yz \rangle\rangle) / \text{chastota}(\langle\langle z \rangle\rangle) + \\ 0,04 * \text{chastota}(\langle\langle z \rangle\rangle) / \text{umumiy} - \text{soni} - \text{so'z} + \\ 0,002$$

Tarjima qilishning eng oddiy statistik modeli - soʻz uchun tarjima modelidir. IBM Model 1 deb nomlanadigan ushbu modelda, jumla bir tildan ikkinchisiga tarjima qilish uchun barcha soʻzlarni tarjima qilishga va ularni toʻgʻri tartibda til modelini taqdim qilish uchun yetarli deb qabul qilinadi.

Model № 1 tomonidan boshqariladigan yagona maʼlumotlar massivi, ikki tilli soʻzlarning oʻzaro tarjima oʻzaro bogʻliqligini ehtimolliklar jadvali:

Soʻz	Amount	bonus	compensation	payment	rate	...
Pul oʻtkazma	15%	8%	6%	71%	0%	
Toʻlov	0%	0%	0%	97%	3%	
...						

Ushbu modelda tarjima jumlasiga bilan asl jumlaning ehtimolligi:

$$P(f | e) = \sum_a P(a, f | e)$$

, bu joyda $P(a, f | e)$ – muayyan e uchun, a tarjimasiga jumlasiga muvofiq, soʻzning darajasiga mos keladigan original jumla f ning paydo boʻlishi ehtimolligi.

$$P(a, f | e) = \prod_{j=1}^m t(f_j | ea_j)$$

, bu joyda t - mos keladigan tarjima soʻzi ea_j bilan aniqlangan holda, j holatidagi asl varianti soʻzining ehtimolligi. Ikkala tarjima translatsiyalari tarjima ehtimolliklari jadvalidan olinadi.

$P(a, f | e)$ ni $P(a | e, f)$ ga ozaytirish, yaʼni, ushbu ikki jumlaning tenglashtirish uchun, bir juftlik uchun ehtimolligi har ehtimolligi berilgan bir $P(a, f | e)$ juftlikning barcha qatorlarning ehtimoli summasi bilan normallashtiriladi:

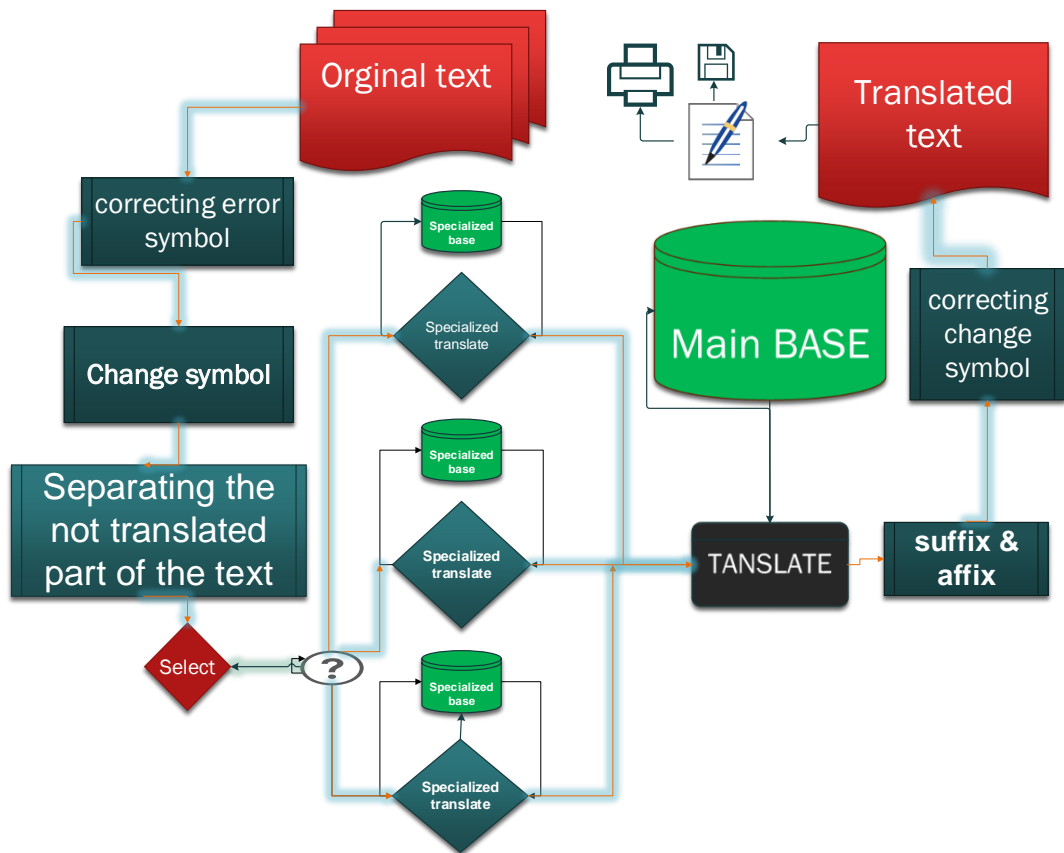
$$P(a|e,f) = \frac{P(a,f|e)}{\sum_a P(a,f|e)}$$

Muayyan ehtimolliklar bilan bir qatorda tenglashtirishlar kompleksiga ega bo'lsak, ular yuzaga keladigan har bir so'z juftligi ehtimoli bo'yicha har bir so'z juftining chastotalarini hisoblashimiz mumkin. Misol uchun, 0,5 va 1 ehtimolliklariga ega bo'lgan ikki tenglashtirishda bir juft so'z paydo bo'lsa, unda bu turdagi juftlikning og'irlik chastotasi $tc - 1,5$ teng.

Ushbu chastotadagi barcha jumalarni mumkin bo'lgan tarjima ekvivalentlarining e ehtimolliги yig'indisi bilan normalizatsiya qilib, o'zaro mos tarjima jumalari ehtimolliklari uchun yangi qiymatlarni olamiz:

$$t(f|e) = \frac{tc(f|e)}{\sum_f tc(f|e)}$$

O'zbekcha-ingilizcha tarjima algoritmi blok sxemasi:



2.2-rasm. Tarjima qilish protsessining ketma-ketligi:

1. Tarjima qilinadigan tekst kiritiladi.

2. Tarjima uchun kerakli tillar va ixtisoslik tanlanadi.
3. Tekst tarjimaga jo‘natiladi.
4. Kiritilgan tekst ichidagi belgilar qayta qodlanadi.
5. Tekstning tarjima qilinmaydigan qismi ajratib olinadi.
6. Ma’lumot qayta ishlanadi va tarjimaga tayyorlanadi.
7. Ixtisoslik va asosiy umumlashgan baza ko‘magida tarjima amalga oshiriladi.
8. So‘zlardagi qo‘shimchalar ustida amallar bajariladi.
9. Avvalgi o‘zgartirishlar joyiga qaytariladi va xatolar to‘g‘rilanadi.
10. Tayyor bo‘lgan tekst tarjima qilingan xolda fayl yoki bufer orkali foydalanuvchiga uzatiladi.

Shu bilan birga foydalanuvchi tomonida tekst noto‘g‘ri kiritilgan holatlarda so‘z va so‘z birikmalarining ma‘nosi va imlosida har xil xatolar yuz berishi va o‘zgarishi ehtimoli bo‘ladi [1].

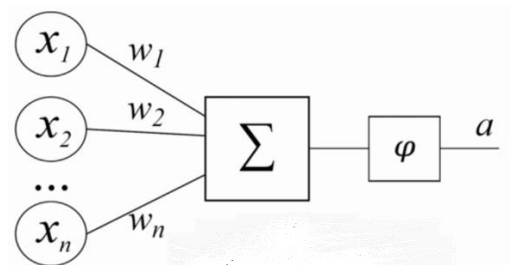
2.3. Tarjimon va lug'at dasturlari modelari va algoritmlari ko‘magida qoraqalpoq a o'zbek tillari uchun ochiq kodli dasturiy vosita ishlab chiqishda neyron tarmoqlaridan foydalanish

Tekstlarni kompyuter ko‘magida tarjima qilish axborot texnologiyalari sohasidagi dolzarb yo‘nalishlardan biri hisoblanadi. Bu yo‘nalishning dolzarbligi shundaki, tarjima qilish protsessi ma’lum qoida va algoritmlarga bo‘ysunmaydi. Shu sababdan bu masalaga aniq yechim topilmagan edi. Hozirgi vaqtgacha tarjimon tizimlarini ishlab chiqish protsessida an’anaviy va statistik usullar keng qo‘llanilib kelindi. Endilikda ushbu muammoga sun‘iy intellekt tizimlari ko‘magida yechim topish imkoniyati paydo bo‘ldi.

Neyron tarmoqlari – bu o‘zaro bir-biriga ta’sir ko‘rsatuvchi ko‘plab neyronlar to‘plamidir. Sun‘iy intellekt tizimlarini avtomatlashtirilgan tarjima sohasida qo‘llash, bu sohada katta burilish yasadi. Shu jumladan sun‘iy neyron tarmoqlari tarjima qilish protsessida qo‘llanila boshladi. Biologik neyron sun‘iy

neyron tarmoqlari uchun prototipdir. Biologik neyron murakkab tuzilishga ega. U yadro, akson, dendrit va sinapsdan tashkil topgan bo‘ladi. Akson bosh va orqa miyadan chiqib, tananing barcha to‘qima va organlariga boradi va ulardan ichki-tashqi muhit ta’irlarini qabul qilib, markaziy nerv sistemasiga o‘tkazadi. Dendrit orqa bosh miyadan tashqariga chiqmaydi, ular bir hujayrani uning atrofidagi boshqa nerv hujayralari bilan bog‘lab turadi. Sinaps bo‘lsa, bir neyron aksonini ikkinchi neyronning dendriti bilan bog‘lash vazifasini bajaradi [4].

Neyron tarmoqlar ilmiy soha sifatida 1943-yillarda Mak-Kallok va Pittslarning ilmiy ishlarida to‘xtalingan bo‘lib, 1949-yilda D. Xebba tomonidan birinchi sun‘iy neyron tarmoqlari modeli ishlab chiqilgan [4].

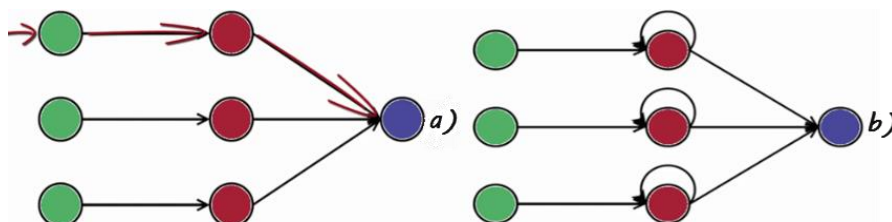


$$a = \varphi\left(\sum_{i=1}^N w_i x_i\right)$$

2.4-rasm. Mak-Kallok va Pittsning sun‘iy neyron tarmoq modeli

Bu joyda $x_{1..n}$ –kirish signallari, a –chiqish, $w_{1..n}$ –har bitta neyronga tegishli signallarning og‘irliklari, φ –aktivlashtirish funksiyasi keltirilgan.

Mak-Kallok va Pittslar o‘z ishlarida sun‘iy neyron tarmoqlarida neyronlar kirish qatlamida tashqi muhitdan signalni qabul qilib, yashirin qatlamda uni qayta ishlab, chiqish bo‘yicha fikrlarini qoldirganlar [4].

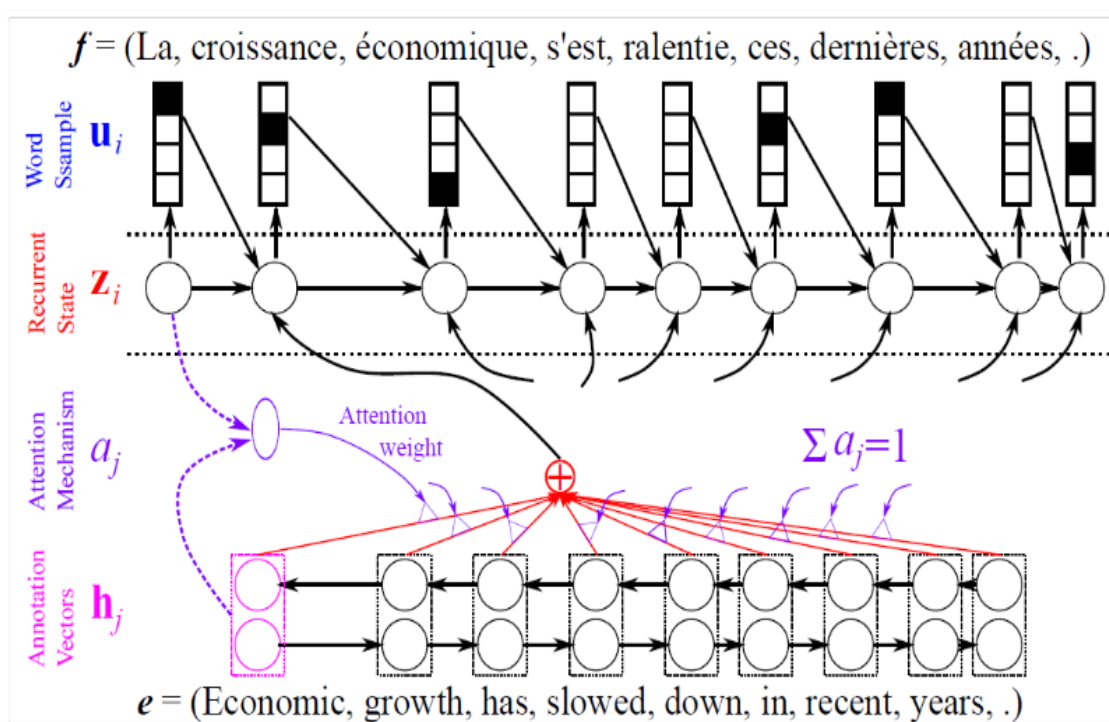


2.5-rasm. To‘g‘ridan-to‘g‘ri taqsimlanuvchi va rekurrent tarmoqlar

Neyron tarmoqlari asosan signallarni to‘g‘ridan-to‘g‘ri taqsimlanuvchi tarmoq (2-rasm. a) va rekurrent tarmoqlar (2-rasm. b) bo‘lib, ikkiga bo‘linadi.

Signallarni to‘g‘ri-to‘g‘ri taqsimlanuvchi tarmoqning rekurrent tarmoqdan tafovuti unda hech qanday sikl qatnashmaydi. Neyron tarmoqlarida yashirin qatlamlar bir yoki bir nechta yashirin qatlamlardan iborat bo‘lishi mumkin [4, 45].

Hozirgi kunda neyron tarmog‘i texnologiyalaridan Microsoft, Google va Yandex singari kompaniyalar tomonidan avtomatlashtirilgan tarjima jarayonlarida foydalanish yo‘lga qo‘yilgan. Google o‘z tarjimon tizimini yaratishda o‘z neyron tarmoqlarini qurishda rekurrent tarmoqlariga asoslandi va yangi *GMNT* avtomatlashtirilgan tarjima tizimini ishlab chiqdi [17]. Bu tizim ikkita 8 qatlamli neyron tarmog‘idan tashkil topgan bo‘lib, kiritilgan tekstni tahlil qiladi va sintezlaydi. Tarjima qilish protsessiga kiritilgan gap oldin (bir vaqtning o‘zida chapdan, o‘ngdan va qayta) o‘qitiladi va ikkinchi neyron tarmoqqa jo‘natiladi. Shu tariqa ikkinchi tildagi gap tuziladi [16].



2.6-rasm. Google neyron mashinali tarjimoni modeli

Tekstlarni neyron tarmoqlari orqali tarjima qilish modellari ichida mashhurlaridan biri, so‘zlarni xuddi vektor singari ko‘rsatadigan *Continuous-Bag-of-Word* va *Skip-Gram* bo‘lib, bu modellar oddiy neyron tarmoqlaridan foydalanib so‘zlarni ma‘nosi bo‘yicha yaqinini tanlash imkoniyatini beradi.

Tekstlarni tarjima qilishda oddiy avtomatlashtirilgan tarjima va lug'at tizimlari o'rniga sun'iy neyron tarmoqlaridan foydalanishning bir qancha afzalliklari mavjud:

- grammatika qonuniyatlari ta'sirida tarjimada uchraydigan ma'no o'zgarishini oldini olish;
- tekstlardagi xatoliklarni bartaraf qilish;
- tarjima sifatini borgan sari oshirib borish (o'zi o'rganuvchi avtomatlashtirilgan tarjima va lug'at tizimlari texnologiyalari asosida);
- gaplarning emotsionallik darajasini aniqlash va boshqalar.

Neyron tarjima va lug'at tizimlarini qurish modellari

Neyron avtomatlashtirilgan tarjima qilish nisbatan yangi paradigma. Avvalgi avtomatlashtirilgan tarjima qilish tizimlarida o'zi o'rganish algoritmi tarjima xotirasini ko'p egallagan va tarjimalar bazasi bilan bog'liq bo'lgan statistik modelda ishlagan.

Biznes strategiyasi ta'siri, boshqa narsalar qatori, neyron tarmoqlari modeliga ham ta'sir o'tkazdi. Tarjimon tizimlari yondashuvida o'zgarishlar ko'pchilik tomonidan qilingan edi, ammo ular bu haqda e'lon qilish uchun shoshilmadilar, ehtimol ular "*Neyron tarmoq*" jamiyatning kutgan sifatini oshira olmasligidan qo'rqishgan. Shu sababdan 2015-2016 yillar ichida bir necha Google, Microsoft, Systran va Yandex kompaniyalar tomonidan birin-ketin neyron mashina tarjimonlari ishga tushirilgan bo'lishi mumkin [16, 17].

Yuqorida ko'rib chiqilgan neyron tarmoqlardan foydalangan holda tarjimaning sifatini oshirishning barcha misollari bu muammolarga to'liq bog'liqdir.

Biroq, biznesning maqsadlari va vazifalari bilan avtomatlashtirilgan tarjima va lug'at tizimlari nisbatan bir-biridan biroz tafovut qiladi. Bu joyda, korporativ kompyuter tarjima tizimiga qo'yiladigan talablardan ayrimlari:

- mijozlar, hamkorlar, investorlar, xorijiy ishchilar bilan biznes-yozishmalarning tarjimasi;

- saytlarni, onlayn-do‘konlarni, mahsulotlar tavsiflarini, yo‘riqnomalarini joylashtirish;
- foydalanuvchilarning kontentini tarjima qilish (*sharhlar, forumlar, bloglar*);
- biznes-jarayonlar va dasturiy mahsulotlar va xizmatlarga tarjimalarni integratsiya qilish imkoniyati;
- terminologiyaga, maxfiylik va xavfsizlikka muvofiq tarjimaning aniqligi.

Marchiano shaxmat o‘yinini statistik mashinalarning tarjimasi qanday ishlashini ko‘rsatish uchun ishlatadi. Shaxmat dasturida cheklangan miqdordagi harakatlarni amalga oshirish mumkin bo‘lgan cheklangan oraliq mavjud. Dasturda eng yaxshi narsani topish uchun barcha mumkin bo‘lgan choralarni hisoblab chiqadi.

Neyron tarmoq tarjimasi ish printsipiga quyidagicha ta’rif berish mumkin: *“Bu pianino chalishga o‘xshaydi: xato qilganingizda, siz buni eslab qolasiz, yana urinib ko‘rasiz va uni qilgunigizgacha takrorlaysiz.”*. Neyron mashina tarjimasi tizimlari xuddi shu tarzda tarjima sifatini oshirmoqda [2].

Statistik tarjimada bo‘lgani singari, neyron avtomatlashtirilgan tarjima va lug‘at tizimlari parallel tekstlarni “inson” tarjimasi bilan avtomatik tarjima qilishni taqqoslashni talab qiladi, faqatgina uni o‘rganish protsessida yakka so‘zlar va iboralar bilan emas, balki butun jumlar bilan ishlaydi. Asosiy muammo shundaki, bunday tizim o‘zi o‘rganish uchun ko‘proq hisoblash kuchi talab qiladi.

Neyron mashina tarjimasi hali ham tarjimon tizimi bilimlar bazasi yaqin boshlab rivojlanayotgan bo‘lsada, statistik ma’lumotlarga ega bo‘lgan kompyuter tarjimasi sifatida rekkurent o‘zi o‘rganish va sun’iy neyron tarmog‘ini yaratish uchun ko‘proq ma’lumot to‘plash imkonini bermoqda. Protsessi tezlashtirish uchun grafik protsessor ishlab chiquvchi NVIDIA kompaniyasi tomonidan GPU, shuningdek, Google tomonidan Tensor (*TPU*) - o‘z grafik chiplarini avtomatlashtirilgan tarjima texnologiyalari uchun moslashtirgan. Grafika chiplari dastlab matritsalarini hisoblash algoritmlari uchun optimallashtirilgan bo‘lib, ular

CPU bilan taqqoslaganda ishlash samaradorligi 7-15 barobarga ko'proqdir [16, 17].

Mashina o'zi o'rganish protsessida, bitta neyron modelni o'qitish 1 dan 3 haftagacha davom qiladi, biroq taxminan bir xil o'lchamdagi statistik model 1-3 kun ichida o'rnatiladi va bu tafovut kattalashuvi oshib boradi.

Neyron mashina tarjimasini ko'plab muammolarga duch kelmoqda, misol uchun, yuqori texnologik kontentni tarjima qilishda. Biroq, noma'lum texnik qisqartirishlarni o'z ichiga olgan manba materiallari, neyron mashina tarjima tizimi singari biron-bir kompyuter tarjima tizimi tomonidan yaxshi tarjima qilinmaydi. Tizim o'zi o'rganish uchun ko'p ma'lumotlarga ega bo'lmagan tillar uchun, misol uchun, qoraqalpoq va o'zbek tillari uchun NMT foydalangan holda boshqa tillarning manba materiallaridan foydalanish imkoniyatini ochadi.

NMT va SMT o'rtasidagi asosiy tafovut rekkurent o'rganish algoritmlari uchun manbalarni taqdim etganinganda, ularga nima izlash kerakligini ko'rsatib bo'lmaydi. Tizim mustaqil tarzda manba jumlasini atrofidagi kontekstli maslahatlar singari andozalarni topishga imkon beradi. Shu bilan birga, jarayonning o'ziga xosligi ko'p jihatdan mavhum bo'lib qolmoqda.

Marchiano, *“rekurrent neyron tarmoq va katta ma'lumotlar, bizga dunyoni idrok qilish va tahlil qilishimizga imkon beradi”* deb ta'kidlaydi.

Shuning uchun biz faqatgina tarjimon tizimi uchun o'zi o'rganish materiallarini taqdim etamiz, agar algoritmlar o'zlarining ishlarini bajarayotgan bo'lsa va tarjimalar aniq bo'lmasa, materialini moslashtiramiz. NMT tizimlari oldindan belgilangan so'z vektorlarini ishlatgani sababli, ular faqat yopiq so'z birikmalarida hosil bo'lishi mumkin (odatda faqat 30 k - 100 k so'zni kengroq ishlatish uchun ishlatiladi).

Hozirda foydalanish mumkin bo'lgan ko'p neyron mashina tarjima tizimlari mavjud: Google Tarjimon, Microsoft Translator, Yandex Translator va Systran Pure Neural Machine Translation ko'rsatish mumkin. Shunga qaramasdan, ishlab chiqarishga tayyor tizimlar nuqtai nazaridan hali ham bir oz oldinga siljish mavjud.

Microsoft, Google, Systran, Baidu, Facebook, Amazon va boshqalar NMT tizimlarining kelgusidagi rivojlanishiga katta ulush qo‘shishmoqda [16, 17].

Neural MT ning chiqarilishi, birinchi navbatda, SMT tizimlariga nisbatan eng yuqori darajani ko‘rsatadigan til yo‘nalishlarida paydo bo‘ladi.

Biroq, bir narsa aniq: Neyron MT – bu tarjimon tizimida katta burilish. Ushbu model qanchalik yoshligini hisobga olsak, tarjima sifatining yaxshilanishi so‘nggi 10 yilga nisbatan juda katta ko‘rsatkichga erishdi. An’anaviy tarjima va mashina tarjima yondashuvlari orasidagi tafovut kengayib boraveradi va biz buning qanchalik uzoq bo‘lishini aniq ayta olmaymiz.

2.4. Tarjima va lug'at dasturlarida ma'lumotlar bazasi va tashqi ma'lumotlar bilan ishlash

Statistik va neyron avtomatlashtirilgan tarjima tizimlari uchun parallel ma'lumotlar katta ro'l o'ynaydi. Tarjima amalga oshirilayotgan juftlik tillar tarjima natijasining sifati eng yuqori o'rinda talab qilinadigan unsurdir. Bizga neyron tarmoqlari o'zi o'rganish ma'lumotlarining miqdori uchun talab qilinishi ma'lum. Tarjima qilinadigan tekst natijasining sifat ko'rsatkichi uchun juda katta miqdorda parallel ma'lumot talab qilinadi, biroq har doim ham yetarlicha bo'lmaydi.

Eng mashhur tillar juftlariga ingliz tili va asosiy Yevropa tillari (*inglizcha* \Leftrightarrow *xitoycha*, *inglizcha* \Leftrightarrow *ruscha*) kompyuter tarjimasida ma'lumot yetarli va bu singari juftliklar uchun neyron tarmoq tuzimlari juda yaxshi natijalarga erishdi. Ammo o'zbek, qoraqalpoq, qozoq va boshqa turkiy tillar (*o'zbekcha* \Leftrightarrow *inglizcha*, *o'zbekcha* \Leftrightarrow *Ruscha*) da parallel ma'lumotlarning ozligi, ushbu tillarda hozirgacha tarjima sifati oqsayotganidan darak beradi.

Biroq, parallel ma'lumotlarning bir necha milliondan ortiq jumlar mavjud bo'lmaganda, neyron tarmoqlar foydasizdir. Parallel ma'lumotlarga boy bo'lgan bunday tillar juda oz, biroq ko'p tillarda tarkiblanmagan ko'p tilli parallel tekstlar mavjud bo'lib, ular asosan yangiliklar, bloglar, ijtimoiy tarmoqlar va davlat tashkilotlari hujjat almashinuvida har doim yangi tarkib hosil bo'ladi. Ushbu tekstlarning barchasi statistik tizimlarni takomillashtirishga yordam berganidek, neyron tarmog'i tarjimasini sifatini yaxshilash uchun ishlatilishi mumkin, ammo afsuski, bunday usullar hali ishlab chiqilmagan.

Biz o'zbek va rus tillari uchun tarjimon tizimini ishlab chiqarishda o'zbek va rus tillari uchun tarjima qilingan adabiyotlar, hujjatlar va boshqa materiallardan foydalanishimiz mumkin.

Ayrim foydalanuvchilar nazarida O'zbekiston doirasida tarjimon tizimini ishlab chiqish kerak emas deb hisoblashadi. Sababi Google, Microsoft va Yandex singari katta kompaniyalar tizimlariga raqobatdosh bo'lishga xavfsirashadi yoki ularning tizimlarini rivojlanishini kutishadi. Ammo biz lokalizatsiya qilinmagan

tarjimon tizimlarini o‘zlarimiz rivojlantirmasak, boshqa kompaniyalar tijoriy maqsadda ishlab chiqargan bunday tizimlari bizning ehtiyojimizni qondira olmaydi deb hisoblaymiz. Sababi bu kompaniyalarda o‘zbek va unga juft bo‘ladigan boshqa tillar tarjimon tizimlarini rivojlantirishda ularda parallel ma’lumotlarning kamligi, tizim sifatini yaxshilashga ko‘maklashuvchi auditoriya singari muammolar mavjud [16, 17].

BOBGA OID XULOSA

Ushbu bobda tekstlarni avtomatlashtirilgan tarjima qilish modellari, usullari va algoritmlari tahlil qilindi. Biz turkiy tillar avtomatlashtirilgan avtomatlashtirilgan tarjima tizimini ishlab chiqarish uchun kerak bo'lgan modellarni tahlil qildik va neyron tarmoq asosidagi yondashuv eng optimal yechim deb topdik.

Bobda neyron avtomatlashtirilgan tarjima va lug'at tizimlari arxitekturasi o'rganilib chiqildi, o'zbek-qoraqalpoq tarjima tizimi uchun algoritmlar taklif qilindi.

Avtomatlashtirilgan tarjima tizimlarini qurilishida enkoder va dekoder etaplarining vazifalar muhim ahamiyatga ega. Yuqorida enkoder va dekoder ish printsplariga alohida toxtalindi [1, 31].

Neyron tarmoq modellari fanga 1950-yillaridan kirib kelganiga qaramay, ba'zi texnik va dasturiy yechimlar bilan bog'liq holda avtomatlashtirilgan tarjima tizimlarida 2015-yildan boshlab qo'llanila boshladi. Neyron mashina tarjimasi qisqa vaqt ichida o'z sifat ko'rsatkichlari bilan boshqa tarjima yondashuvlaridan ajralib chiqdi va tarjima sohasida katta burilish yasadi yohud sifatli tarjima muammosining yechimi bo'ldi.

Modeli yaratish uchun bizga ozida 100 million jilddan (foydalanishdan) va ozroq sifatli tarjima qilish uchun 500 million belgidan iborat ma'lumotlar bazasi kerak bo'ladi. Afsuski hali hech bir kompaniyada bunday materiallar mavjud emas. Ammo bu sifatli tarjima imkonsizligini dalili hisoblanmaydi. Sababi 2015-2016-yildan hozirgacha tarjimon tizimlari sifat ko'rsatkichi sezilarli burilish bilan o'sdi. Yangi texnologiyalar va yangi yondashuvlar yordami bilan yaqin kelajakda inson tarjimasi sifatiga raqobatlasha oladigan tarjimon tizimlari paydo bo'ladi.

III. AVTOMATLASHTIRILGAN TARJIMA TIZIMI NATIJALARI TAHLILI

Ushbu bobda biz avtomatlashtirilgan tarjima modellari asosida ishlab chiqilgan tizim natijalarini tahlil qilamiz. Bitiruv malakaviy ishining ushbu bobida bir qancha yondashuvlar ustida sinovlar o‘tkazildi. Avtomatlashtirilgan tarjima tizimlari modellari natijalari Matlab va joriy tizim algoritmlari ko‘magida olinadi. Shuningdek, ishlab tarjimon tizimi sifat natijalari boshqa tarjimon tizimlari natijalari bilan taqqoslanadi. Sifat ko‘rsatkichlarini taqqoslash ikki hil usulda to‘g‘ridan-tog‘ri tizim foydalanuvchisidan tarjima qilingan tekstga sifat darajasini belgilash va tizim ichki avtomatik usulda ishlovchi algortmlari bilan taqqoslash imkoniyati orqali amalga oshiriladi.

3.1. Mavjud tarjimon va lug‘at tizimlarining afzalliklari va kamchiliklari.

Tarjima sifatini baholash usullari

Shunday qilib, taqqoslangan ma‘lumotlarga asoslanib, biz mashinani tarjima qilish tizimining quyidagi afzalliklari va nuqsonlarini ajratib olishimiz mumkin.

Tarjima tizimlarining afzalliklari:

- mutaxassisliklar bo‘yicha lug‘atlar mavjudligi, tezkor qidiruv, bir necha oynada joylashish, bir necha tarjima variantlarini bir vaqtda ko‘rib chiqish imkoniyati;
- shaxsiy foydalanuvchi lug‘atini yaratish imkoniyati;
- Microsoft Office va boshqa muharrirlar bilan bog‘lanish, asl nusxadan yoki uning tarjimasidan foydalanmasdan, tarjima uchun tegishli yozishmalarni tanlash;
- turli tillardan tarjima qilish imkoniyati;
- kompaktlik, kompyuterda deyarli cheksiz ko‘p ma‘lumot mavjudligi;
- izlash tezligi.

Tarjima va lug‘at tizimlarining kamchiliklari:

Til juftliklari	Qoidalariga asoslangan yondashuv	GNMT	Inson tarjimasi	Tarjima sifat natijalari yaxshilinish nisbati
-----------------	----------------------------------	------	-----------------	---

- ekran o'lchamlari tufayli paydo bo'lgan cheklangan ko'rinish, katta bir lug'at hosil qilish sizni birdaniga ko'proq so'z ma'nosini ko'rsatadi, ayniqsa tarjima qilinayotgan so'zlar juda ko'p sonli so'z ma'nosi ko'p bo'lsa;
- lug'at-tarjimonlar grammatikaga, stilistikaga va so'z birikmalariga rioya qilmaydilar, so'zlarni, badiiy usullarni hisobga olmaydilar;
- ko'pincha tekst tarjimonlari tarjima qilgan tekst varianti kontekstga mos kelmasligi mumkin, chunki u mavjud variantlardan bittasini tanlaydi;
- bazada hech qanday variant bo'lmasa, uni tarjima qilmaydi;
- elektron lug'at-tarjimonlari ko'pincha so'zni boshqa tilga tarjima qilishning bir nechta variantlarini beradi, bu tekst lug'atdagi yozishmalardagi tafovutlarni tushuntirmasdan, bu esa bu erda to'g'ri yozishmalarni tanlashni murakkablashtiradi.

BLUE algoritmlari tarjima sifatini baholashning mashhur usullaridan bo'lib, bu usul tarjimon tizimi sifat ko'rsatkichlarini belgilash uchun ishlatiladi.

Google tadqiqotida neyron kompyuter tarjimasi natijasi klassik statistik yondoshuv (yoki PHM asosidagi mashinalar tarjimasi, PBMT) natijasi bilan taqqoslaganda, til juftligiga qarab, 58% dan 87% gacha bo'lgan relativ o'sishni ko'rsatgan [16, 17].

Ingiliz-Ispan	4.885	5.428	5.55	87%
Ingiliz- Fransuz	4.932	5.295	5.496	64%
Ingiliz-Xitoy	4.035	4.594	4.987	58%
Ispan-Ingiliz	4.872	5.187	5.372	63%
Fransuz- ingiliz	5.046	5.343	5.404	83%
Xitoy-Ingiliz	3.694	4.263	4.636	60%

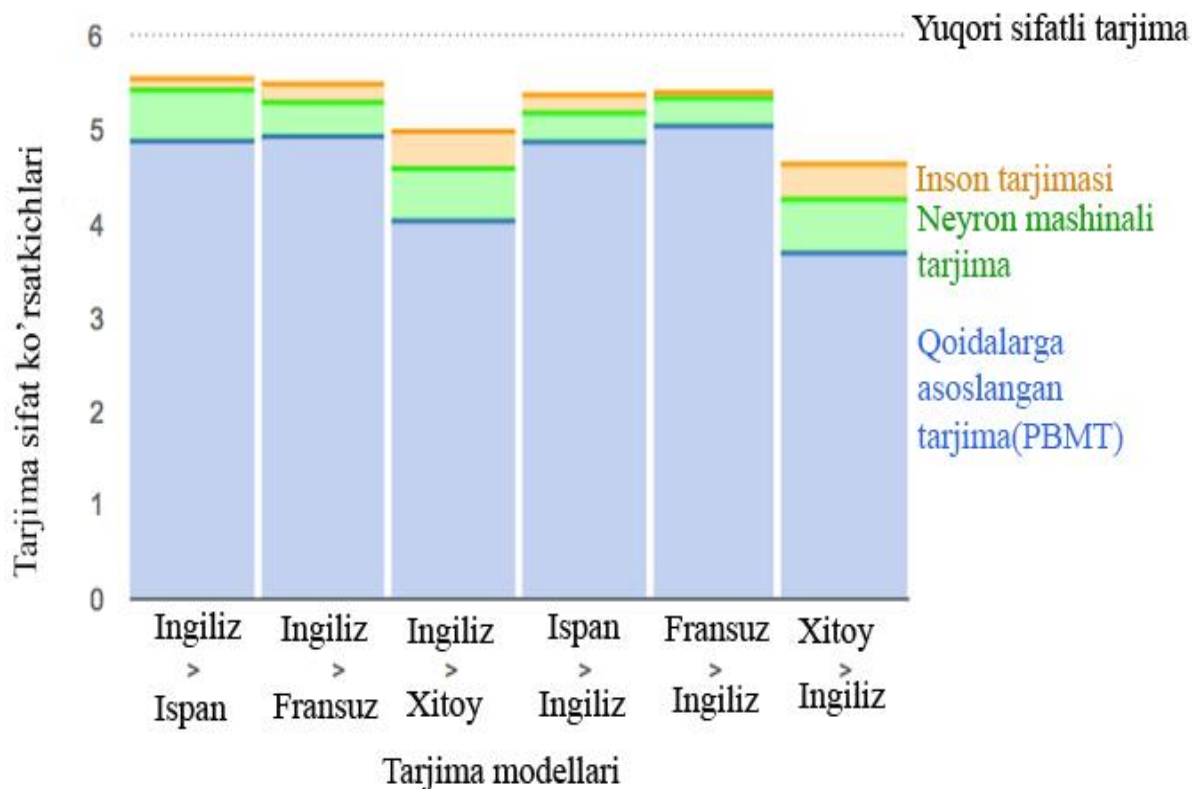
3.1-jadval. BLUE algoritmlari ko'magida olingan tarjimon tizimlarining sifat ko'rsatkichlari tahlili

SYSTRAN, tarjima sifatini, turli tizimlar tomonidan ko'rsatib o'tilgan bir qancha ko'rsatib o'tilgan versiyalar va "inson" tarjimasi bilan taqqoslanib baholanadi. Neyron tarmoq tarjimasi 46% hollarda inson tomonidan qilingan tarjimaga ustunligini bildirgan.

A>=B	RE	NM	BI	GO	NA	V8
RE		61.8	81.2	70	74.7	72.3
NM	46.4		74.5	73.9	72	63.1
BI	29	39		44.4	54.6	44
GO	38.2	40.2	79.3		72.1	56.1
NA	34.7	40.6	73.4	51.6		44.5
V8	35.6	49.9	77.7	68.4	79.2	

3.2-rasm. Systran tarjimon tizimi sifat ko'rsatkichi

Google kompaniyaning vakillari “Nisbiy takomillashtirish”, ya'ni neyron yondoshuv bilan mumtoz statistik translatsiyada bo'lgan inson tarjimasi sifatiga qanday yaqinlashish haqida tadqiqot olib borishmoqda [17, 16].



3.3-rasm. Inson tarjimasi, neyron mashina tarjimasi va qoidalarga asoslangan tarjima modellari sifat ko'rsatkichlari

Google tomonidan ko'rsatib o'tilgan natijalarni tahlil qiladigan soha mutaxassislari "GNMT tizimi: Inson va Mashinani Tarjima qilish o'rtasidagi aloqani ko'paytirish" maqolasida ko'rsatib o'tilgan natijalar haqida shubha bilan qarashadi [16, 17].

PROMT tarjimon tizimi boshqa raqobatchilar bilan muntazam tarzda tarjima sifatini taqqoslab boradi, ishlab chiqaruvchilar, neyron tarmoq tarjimasi avvalgi yondashuvlarga nisbatan qanchalik ustunligini tekshirishimiz mumkin.

3.3. Tarjima va lug'at dasturlarini qurishda neyron tarmoqlaridan foydalanish

Yuqorida o'tganimizdek, "universal" tarjimon doimo maqbul sifat ko'rsatkichini bermaydi va muayyan atamalarni qo'llab-quvvatlamaydi. Neyron tarmoq ko'magida tarjima qilish uchun tizim asosiy talablarga javob berishi va shu jarayonlarni integratsiya qilishi kerak:

- neyron tarmoqni o'qitish uchun yetarli miqdordagi parallel tekstlar kerakligi;
Ko'p hollarda, tarjima qilinayotgan mavzu bo'yicha parallel tekstlar oz yoki umuman tizimlarda mavjud bo'lmasligi mumkin. Bor bo'lgan taqdirda ham ular avtomatik tarzda ishlov berish uchun tarkiblanmagan yoki tasniflangan holatda bo'lishi mumkin.

Modeli yaratish uchun esa bizga ozida 100 million jilddan (foydalanishdan) va ozroq sifatli tarjima qilish uchun 500 million belgidan iborat ma'lumotlar bazasi kerak bo'ladi. Afsuski hali hech bir kompaniyada bunday materiallar mavjud emas.

- natijaning sifatini avtomatik baholash uchun mexanizm yoki algoritmlarning mavjudligi;
- yetarli hisoblash kuchi;

"Universal" neyron tarjimon natijasi ko'pincha sifat uchun mos kelmaydi va uning sifatini va ishlash tezligini ta'minlaydigan maxsus neyron tarmog'ini foydalanish uchun "kichik bulut" talab qilinadi.

- maxfiylik masalalari bilan qanday ishlashi noma'lum;

Har bir mijoz o'ziga tegishli ma'lumotlarini xavfsizligini ta'minlagan holda bulutga yetkazib berishga tayyor emas, neyron mashina tarjimasi bo'lsa birinchi bulutli usul deb biladilar.

3.4. Ishlab chiqilgan tarjimon va lug'at tizimi natijalari

Tarjimon tizimi tekstlarni o'zbek-qoraqalpoq tillarida tarjima qiluvchi tarjimon tizimidir. Biz bu avtomatlashtirilgan tizimni ishlab chiqishda avtomatlashtirilgan tarjima yondashuvlaridan foydalandik.



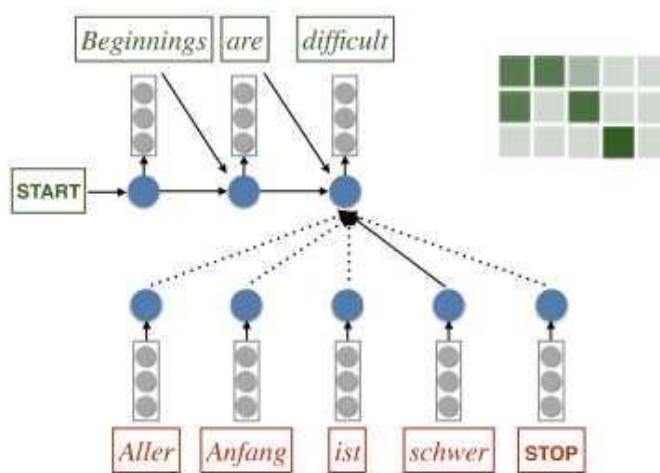
3.4-rasm. Tarjimon tizimi interfeysi

Ishlab chiqilgan tarjimon tizimi asosan statistik tarjima yondashuvi asosiga qurilgan bo'lib, tizim ko'magida o'zbek va qoraqalpoq tillardagi tekstlarni qiyinchiliksiz bir tildan ikkinchisiga tarjima qilish mumkin. Avtomatlashtirilgan tarjimon tizimlari tarjima jaroyiniga yangicha yondashuv bo'lib, tizim ko'magida tekstlarni bir tildan ikkinchi tilga qisqa vaqt ichida tarjima qilish imkoniyati yaratildi.

Quyida tarjimon tizimi ishga tushirilgandan keyingi 3 oylik tahlil natijalarini keltiramiz. Ushbu qismda ko'rsatiladigan natijalar Google Analitics servisi orqali olingan. Natijalar 3 oy davomida yi'g'ilgan ma'lumotlarga asoslanadi. Statistik ma'lumot sifatida foydalanuvchilarning tarjimondan foydalanish darajasi va boshqa tahliliy ko'rsatkichlar taqdim etiladi.

Biz tarjimon tizimi ishlab chiqarishda foydalanayotgan model sodda bo'lib, an'anaviy modellarga kiritilgan bir nechta induktiv nuqsonlardan tashqarida emas.

Oxirgi yillarda neyron tarmoq yondashuvi deb ataladigan uchinchi avlod avtomatlashtirilgan tarjima va lug'at tizimlari modellari an'anaviy statistik tarjima tizimlaridan ustun turadigan yoki ayrim hollarda raqobatlashadigan yuqori ko'rsatkichli tarjimalarni taqdim etmoqda.



3.15-rasm: Sun'iy intellekt tarjimasining taxminiy modeli.

Enkoder dekoderning pastki qismida ko'rsatiladi va qirralarning diqqat mexanizmiga mos keladigan ikkita ulanadi. Katta qovurg'alar ko'proq e'tiborni tortadi va bu qiymatlar har bir maqsadli so'z uchun bitta qator bilan matritsa shaklida ham ko'rsatiladi.

OpenNMT - bu universal o'zi o'rganuvchi tizim bo'lib, asosan avtomatlashtirilgan tarjima va lug'at tizimlari, umumlashtiruv, rasimga va nutqni tanib olish singari turli vazifalarni qamrab oluvchi modellar ketma-ketligiga ixtisoslashgan. Ushbu frameworkda shuningdek, tilni modellashtirish va ketma-ket etiketlash singari boshqa tartibga solinmaydigan vazifalarga ham yechimini topgan.

Bu ilovalar barcha ishlatiladigan modullar enkoder, dekode, biriktiruvchi qatlam, diqqat darajalari muntazam qayta ishlanadi va kengaytiriladi. Qo'llanish uchun qulay bo'lib, imkon qadar keng qamrovda qo'llaniladi va ilovalar, mijoz-server yoki kutubxonalar orqali ishlatilishi mumkin.

OpenNMT tarjimon tizimi tarkibi:

Ma'lumot formati:

Ma'lumotlar fayllarining formati ochiq modellar tomonidan foydalaniladigan *opennmt.inputters.Inputter* tomonidan aniqlanadi.

Tekst:

Barcha *opennmt.inputters.TextInputters* tekstli faylni quyidagi joylarda kiritishini kutadi:

- jumlar yangi satr bilan ajratiladi
- tokenlar bo'sh joy bilan;

Vektorlar:

Opennmt.inputters.SequenceRecordInputter seriyali *TFRecords* faylini kutadi.

Parallel kirishlar

Opennmt.inputters.ParallelInputter dan foydalanilganda, ko'pgina kirish fayllari kirish fayllari singari kutiladi.

O'zi o'rganuvchi tizim materiallari ko'rinishi:

data:

train_features_file:

- *train_source_1.records*
- *train_source_2.txt*
- *train_source_3.txt*

BOBGA OID XULOSA

Bitiruv malakaviy ishining yakunlovchi bobida bitiruv malakaviy ishi davomida olingan natijalar tahlil qilindi. Ishlab chiqilgan model, algoritmlar va tizim ustida eksperiment o'tkazildi. Statistik ma'lumotlar olindi.

Ishlab chiqilgan tizim natijalarini olishda to'g'ridan-to'g'ri foydalanuvchilar fikrlari tahlil qilindi. Tizimda tarjima qilinga so'zlar o'zi o'rganuvchi tizim faoliyatiga katta ta'sir ko'rsatishi ma'lum bo'ldi. Avtomatlashtirilgan tarjima qilish tizimidan foydalangan holda tarjima qilish borgan sari inson mexnatini yengillashtirishi aniqlandi. O'zi o'rganuvchi tizim yuqori sifat ko'rsatkichini berishi uchun kerak texnik talablar va ma'lumotlar qayta ishlash uchun vaqt talab qilinadi.

XULOSA

Ushbu bitiruv malakaviy ishida, tekstlarni avtomatlashtirilgan turda kompyuter yordamida tarjima qilish modellari va usullari tadqiq qilindi va kerak natijalar olindi. Bitiruv malakaviy ishi davomida hozirgi kun dolzarb masalasi bo'lgan sifatli tarjima muammosiga yangicha yechim hisoblanmish, tarjimon tizimi ishlab chiqildi.

Ishlab chiqarilgan tizim sifatini tekshirish maqsadida eksperiment o'tkazildi. Avtomatlashtirilgan tarjimon tizimi natijalari, tarjimon ekspert natijalari bilan taqqoslandi.

Ushbu bitiruv malakaviy ishi natijalariga ko'ra quyidagi xulosalarga kelindi. Sun'iy intellekt tizimlari tarjima borasidagi qarashlarni tubdan o'zgartirdi. Neyron tarmoqlari va o'zi o'rganuvchi *avtomatlashtirilgan tarjima va lug'at tizimlari* algoritmlari ko'magida biz eng optimal tarjima sifatiga erishishimiz mumkin. Ushbu xulosalar bizning 3 yillik tajribalarimiz va Google, Yandex, Microsoft kompaniyalari tomonidan ishlab chiqarilgan va rivojlantirilayotgan tarjimon tizimlarini taxlil qilish natijasida olindi[16, 17].

Tadqiq natijalariga ko'ra kompyuter lingvistikasini algoritmlashtirish va faqat lingvistikaga tayangan holda tarjimoni amalga oshirish kerak natijalarni bermadi. Agarda neyron tarmoqlari orqali tarjima amalga oshirilsa, lingvistikadan foydalanish shart emas. Sababi tabiiy tildagi tekstlarni kompyuter ko'magida tarjima qilishda hech qanday algoritim va qonuniyatlar amal qilmaydi. Til tekstlarini taxlil qilish chiziqli bo'lmagan algoritmlarga mos keladi. Shu sababdan optimal natijalarni olish uchun biz neyron tarmoq modellaridan foydalanishimizga to'g'ri keladi.

Biz tarjimon tizimining sifatini yaxshilash uchun, tizimning o'zi o'rganuvchi *avtomatlashtirilgan tarjima va lug'at tizimlari* algoritmlari modullarini doimiy tarzda ishlatib boramiz, bu yuqori sifat natijalariga erishishning birdan bir kalitidir.

Bitiruv malakaviy ishining ilmiy nazariy ahamiyati katta bo'lib, ushbu bitiruv malakaviy ishidagi sun'iy intellekt tizimlari bilan bog'liq nazariy ma'lumotlardan dars protsessida foydalanish mumkin.

Bitiruv malakaviy ishi asosida ishlab chiqilgan model, algoritm va dasturlar *OpenSource* shaklida ochiq taklif etiladi. Ushbu tizimdan O‘zbek va boshqa turkiy tillardagi tekstlarni tarjima qilishda foydalanish mumkin. Shu tizim asosida ishlab chiqarilgan tarjimon tizimlari avtomatlashtirilgan tarjima tizimlari kerak bo‘lgan tashkilotlarga foydalanish uchun taqdim etiladi.

Modeli yaratish uchun bizga ozida 100 million jilddan (foydalanishdan) va ozroq sifatli tarjima qilish uchun 500 million belgidan iborat ma’lumotlar bazasi kerak bo‘ladi. Afsuski hozircha hali hech bir kompaniyada foydalanishga tayyor shu singari materiallar mavjud emas.

Bitiruv malakaviy ishida neyron tarmoq tarjima modellarini o‘rganish bo‘yicha quyidagi xulosalarga kelindi:

- umuman, neyron avtomatlashtirilgan tarjima va lug‘at tizimlari “toza” statistik yondashuvga qaraganda yuqori sifatli natijalar beradi;
- neyron tarmoq orqali avtomatik tarjima qilish – “*universal tarjima*” muammosini yechish uchun yanada qulayroqdir;
- mashinani tarjima qilishga qaratilgan yondashuvlarning hech biri tarjima vazifasini yechish uchun ideal universal vosita emas;
- biznesni tarjima qilish muammolarini yechish uchun faqatgina ixtisoslashgan echimlar barcha talablarga javob berishni ta’minlaydi;

Tarjimon tizimlarini ishlab chiqish uchun bu uchun eng mos bo‘lgan ko‘p yondashuvli usullardan foydalanishimiz kerak. Foydalanuvchi uchun ichida neyron tarmoq bor yoki yo‘qligi ahamiyatga ega bo‘lmay, vazifani tushunish va sifat ko‘rsatkichi eng muhimidir.

ADABIYOTLAR

1. Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv: 1406.1078, 2014.
2. О. С. Кулагина. О современном состоянии машинного перевода // Математические вопросы кибернетики, вып. 3, М.: Наука, 2013, стр. 5—50. Библиография из 140 названий. ISBN 5-02-014323-5.
3. Uteuliyev N.U. Abdaliev G. R. Achilov D.S. Sernazarov R. A. Web-dasturlash tillari ko'magida tekst tarjimalovchi dastur ishlab chiqish // Hisoblash va amaliy matematika muammolari. Toshkent-2016. 100-106 b.
4. Uteuliev N.U., Achilov D.S. Sun'iy neyron tarmoqlari modellarini tekstlarni tarjima qilish protsessida qo'llash. Значение информационно-коммуникационных технологий в инновационном развитии реальных отраслей экономики. Сборник докладов С. 207-209. Toshkent-2017.
5. Uteuliev N.U., Achilov D.S. Tekstlarni tarjima qilish protsessida axborot texnologiyalarining o'rni. Kadrlar sifatini oshirishda axborot texnologiyalarining o'rni mavzusidagi, TATU professor o'qituvchilarining ilmiy-uslubiy konferentsiyasi. Maruzalar to'plami. 198-199 b. Toshkent-2017
6. Гращенко Л. А., Клышинский Э. С., Тумковский С. Р., Усманов З. Д. Концептуальная модель системы русско-таджикского машинного перевода // Доклады Академии наук Республики Таджикистан.- 2011. - том 54, № 4. - С. 279-285.
7. Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages

- 1412–1421, Lisbon, Portugal, September. Association for Computational Linguistics.
8. Nazirov Sh.A. Turkiy tillar uchun elektron tarjimaning dasturiy ta'minoti yaratish muammolari // TATU xabarlari №4/2011 Toshkent. 57-60 bet.
 9. Nazirov Sh.A., Raxmanov Q.S., Xojiev S.N., To'xtamurodova Sh.Sh. Tabiiy tillarda lug'atlarni qurish usullari // "Aniq fanlarni o'qitishning dolzarb muammolari" mavzusidagi respublika ilmiy-amaliy anjumani materiallari. 2013 yil 22-23 noyabr, Guliston. 114-117b.
 10. Nazirov Sh.A., Raxmanov Q.S., To'xtamurodova Sh.Sh. Turkiy tillar uchun elektron lug'atlarni qurish // "Aniq fanlarni o'qitishning dolzarb muammolari" mavzusidagi respublika ilmiy-amaliy anjumani materiallari. 2013 yil 22-23 noyabr, Guliston. 121-123b.
 11. Turkiy tillar uchun elektron lug'at yaratish muammolari. Abidova Sh.B. TATU, Respublika ilmiy-texnik anjumanining Ma'ruzalar to'plami 1-qism, Toshkent – 2017
 12. Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL), pages 104–111, New York, NY.
 13. O'zbekcha – qoraqalpoqcha va qoraqalpoqcha - o'zbekcha lug'at. Akad. J. Bazarbev tahriri ostida. T - 2011
 14. Incorporating Structural Alignment Biases into an Attentional Neural Translation Model. Trevor Cohn and Cong Duy Vu Hoang and katerina Vymolova. University of Melbourne Melbourne, VIC, Australia
 15. Гращенко Л. А., Клышинский Э. С., Тумковский С. Р., Усманов З. Д. Концептуальная модель системы русско-таджикского машинного перевода // Доклады Академии наук Республики Таджикистан. — 2011. — том 54, № 4. — С. 279—285.

16. Chitu, Alex (22 October 2007). "Google Switches to Its Own Translation System". [Googlesystem.blogspot.com](http://googlesystem.blogspot.com). Retrieved 2012-08-13.
17. "Google Translator: The Universal Language". Blog.outer-court.com. 25 January 2007. Retrieved 2012-06-12.
18. Nagao, M. 1981. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle, in *Artificial and Human Intelligence*, A. Elithorn and R. Banerji (eds.) North- Holland, pp. 173–180, 1984.
19. "The Association for Computational Linguistics – 2003 ACL Lifetime Achievement Award". Association for Computational Linguistics. Retrieved 2010-03-10.
20. Mohammadi, Mehdi (2011). A Semantic-Tended Approach to Recombination in EBMT Systems. 7th International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE), 2011. IEEE. pp. 260–264. doi:10.1109/NLPKE.2011.6138205.
21. H. Somers, "Review article: Example-based Machine Translation," *Machine Translation*, no. 14, pp. 113-157, 1999.
22. Adam Boretz. "Boretz, Adam, "AppTek Launches Hybrid Machine Translation Software" SpeechTechMag.com (posted 2 MAR 2009)". Speechtechmag.com. Retrieved 2012-06-12.
23. "KantanMT Users Can Now Customise and Deploy Neural Machine Translation Engines | Slator". Slator. 2017-03-13. Retrieved 2017-06-23.
24. "Omniscien Technologies Announces Release of Language Studio™ with Next-Generation NMT Technology | Slator". Slator. 2017-04-21. Retrieved 2017-06-23.
25. Rowe, Sam Del (2017-06-12). "SDL Adds Neural Machine Translation to Its Enterprise Translation Server". *CRM Magazine*. Retrieved 2017-06-23.
26. Milestones in machine translation – No.6: Bar-Hillel and the nonfeasibility of FAHQT Archived 12 March 2007 at the Wayback Machine. by John Hutchins

27. Hybrid approaches to machine translation. Costa-jussà, Marta R., Rapp, Reinhard, Lambert, Patrik, Eberle, Kurt, Banchs, Rafael E., Babych, Bogdan, Switzerland. ISBN 9783319213101. OCLC 953581497.
28. Обзор систем машинного перевода. Колесник А. С. II Всеукраїнська науково-практична конференція "Інтелектуальні системи та прикладна лінгвістика", 28 березня 2013р., м. Харків

ILOVA

Dictionary.cc

```
#include "onmt/Dictionary.h"

#ifdef ANDROID_GNUSTL_COMPAT
    # include "onmt/android_gnustl_compat.h"
#endif

#include <fstream>

#include "onmt/th/Utils.h"

namespace onmt
{

    const size_t Dictionary::pad_id = 0;
    const size_t Dictionary::unk_id = 1;
    const size_t Dictionary::bos_id = 2;
    const size_t Dictionary::eos_id = 3;

    Dictionary::Dictionary()
    {
    }

    Dictionary::Dictionary(th::Class* dict)
    {
        load(dict);
    }

    void Dictionary::load(th::Class* dict)
    {
        auto dict_data = dynamic_cast<th::Table*>(dict->get_data());
        auto id2word = th::get_field<th::Table*>(dict_data, "idxToLabel");

        auto array = id2word->get_array();

        for (size_t i = 0; i < array.size(); ++i)
        {
            const std::string& word =
dynamic_cast<th::String*>(array[i])->get_value();
            _id2word.push_back(word);
            _word2id[word] = i;
        }
    }
}
```

```

size_t Dictionary::get_size() const
{
    return _id2word.size();
}

size_t Dictionary::get_word_id(const std::string& word)
const
{
    auto it = _word2id.find(word);

    if (it == _word2id.cend())
        return unk_id;

    return it->second;
}

const std::string& Dictionary::get_id_word(size_t id) const
{
    return _id2word[id];
}
}

```

ITranslator.cc

```

#include
"onmt/ITranslator.h"

#include "onmt/SpaceTokenizer.h"
#include <utility>

namespace onmt
{

    std::string
    ITranslator::translate(const std::string& text)
    {
        return translate(text, SpaceTokenizer::get_instance());
    }

    std::string
    ITranslator::translate(const std::string& text,
                           float& score,
                           size_t& count_tgt_words,
                           size_t& count_tgt_unk_words,
                           size_t& count_src_words,
                           size_t& count_src_unk_words)
    {

```

```

        return translate(text, SpaceTokenizer::get_instance(), score,
count_tgt_words, count_tgt_unk_words, count_src_words,
count_src_unk_words);
    }

    std::string
    ITranslator::translate(const std::string& text,
                          ITokenizer& tokenizer)
    {
        float score;
        size_t count_tgt_words, count_tgt_unk_words, count_src_words,
count_src_unk_words;
        return translate(text, tokenizer, score, count_tgt_words,
count_tgt_unk_words, count_src_words, count_src_unk_words);
    }

    std::string
    ITranslator::translate(const std::string& text,
                          ITokenizer& tokenizer,
                          float& score,
                          size_t& count_tgt_words,
                          size_t& count_tgt_unk_words,
                          size_t& count_src_words,
                          size_t& count_src_unk_words)
    {
        std::vector<float> best_scores;
        std::vector<size_t> best_count_tgt_words, best_count_tgt_unk_words;
        auto res = get_translations(text, tokenizer, best_scores,
best_count_tgt_words, best_count_tgt_unk_words, count_src_words,
count_src_unk_words);
        score = best_scores.at(0);
        count_tgt_words = best_count_tgt_words.at(0);
        count_tgt_unk_words = best_count_tgt_unk_words.at(0);
        return res.at(0);
    }

    std::vector<std::string>
    ITranslator::get_translations(const std::string& text)
    {
        return get_translations(text, SpaceTokenizer::get_instance());
    }

    std::vector<std::string>
    ITranslator::get_translations(const std::string& text,
                                  std::vector<float>& scores,
                                  std::vector<size_t>& count_tgt_words,
                                  std::vector<size_t>&
count_tgt_unk_words,

```

```

        size_t& count_src_words,
        size_t& count_src_unk_words)
    {
        return get_translations(text, SpaceTokenizer::get_instance(),
scores, count_tgt_words, count_tgt_unk_words, count_src_words,
count_src_unk_words);
    }

    std::vector<std::string>
    ITranslator::get_translations(const std::string& text,
                                ITokenizer& tokenizer)
    {
        std::vector<float> scores;
        std::vector<size_t> count_tgt_words, count_tgt_unk_words;
        size_t count_src_words, count_src_unk_words;
        return get_translations(text, tokenizer, scores, count_tgt_words,
count_tgt_unk_words, count_src_words, count_src_unk_words);
    }

    TranslationResult
    ITranslator::translate(const std::vector<std::string>& tokens,
                          const std::vector<std::vector<std::string> >&
features)
    {
        size_t count_src_unk_words;
        return translate(tokens, features, count_src_unk_words);
    }

    std::vector<std::string>
    ITranslator::translate_batch(const std::vector<std::string>& texts)
    {
        return translate_batch(texts, SpaceTokenizer::get_instance());
    }

    std::vector<std::string>
    ITranslator::translate_batch(const std::vector<std::string>& texts,
                                std::vector<float>& scores,
                                std::vector<size_t>& count_tgt_words,
                                std::vector<size_t>&
count_tgt_unk_words,
                                std::vector<size_t>& count_src_words,
                                std::vector<size_t>&
count_src_unk_words)
    {
        return translate_batch(texts, SpaceTokenizer::get_instance(),
scores, count_tgt_words, count_tgt_unk_words, count_src_words,
count_src_unk_words);
    }

```

```

std::vector<std::string>
ITranslator::translate_batch(const std::vector<std::string>& texts,
                             ITokenizer& tokenizer)
{
    std::vector<float> scores;
    std::vector<size_t> count_tgt_words, count_tgt_unk_words;
    std::vector<size_t> count_src_words, count_src_unk_words;
    return translate_batch(texts, tokenizer, scores, count_tgt_words,
count_tgt_unk_words, count_src_words, count_src_unk_words);
}

std::vector<std::string>
ITranslator::translate_batch(const std::vector<std::string>& texts,
                             ITokenizer& tokenizer,
                             std::vector<float>& scores,
                             std::vector<size_t>& count_tgt_words,
                             std::vector<size_t>&
count_tgt_unk_words,
                             std::vector<size_t>& count_src_words,
                             std::vector<size_t>&
count_src_unk_words)
{
    std::vector<std::string> translations;
    scores.clear();
    count_tgt_words.clear();
    count_tgt_unk_words.clear();
    std::vector<std::vector<float> > batch_scores;
    std::vector<std::vector<size_t> > batch_count_tgt_words,
batch_count_tgt_unk_words;
    auto res = get_translations_batch(texts, tokenizer, batch_scores,
batch_count_tgt_words, batch_count_tgt_unk_words, count_src_words,
count_src_unk_words);
    for (size_t i = 0; i < res.size(); ++i)
    {
        translations.push_back(std::move(res[i].at(0)));
        scores.push_back(batch_scores[i].at(0));
        count_tgt_words.push_back(batch_count_tgt_words[i].at(0));

count_tgt_unk_words.push_back(batch_count_tgt_unk_words[i].at(0));
    }

    return translations;
}

std::vector<std::vector<std::string> >
ITranslator::get_translations_batch(const std::vector<std::string>&
texts)

```



```

    {
        return get_translations_batch(texts,
SpaceTokenizer::get_instance());
    }

    std::vector<std::vector<std::string> >
    ITranslator::get_translations_batch(const std::vector<std::string>&
texts,
                                        std::vector<std::vector<float> >&
scores,
                                        std::vector<std::vector<size_t>
>& count_tgt_words,
                                        std::vector<std::vector<size_t>
>& count_tgt_unk_words,
                                        std::vector<size_t>&
count_src_words,
                                        std::vector<size_t>&
count_src_unk_words)
    {
        return get_translations_batch(texts,
SpaceTokenizer::get_instance(), scores, count_tgt_words,
count_tgt_unk_words, count_src_words, count_src_unk_words);
    }

    std::vector<std::vector<std::string> >
    ITranslator::get_translations_batch(const std::vector<std::string>&
texts,
                                        ITokenizer& tokenizer)
    {
        std::vector<std::vector<float> > scores;
        std::vector<std::vector<size_t> > count_tgt_words,
count_tgt_unk_words;
        std::vector<size_t> count_src_words, count_src_unk_words;
        return get_translations_batch(texts, tokenizer, scores,
count_tgt_words, count_tgt_unk_words, count_src_words,
count_src_unk_words);
    }

    TranslationResult
    ITranslator::translate_batch(const
std::vector<std::vector<std::string> >& batch_tokens,
                                const
std::vector<std::vector<std::vector<std::string> > >& batch_features)
    {
        std::vector<size_t> batch_count_src_unk_words;
        return translate_batch(batch_tokens, batch_features,
batch_count_src_unk_words);
    }

```

```
}
```

Translationresult.cc

```
#include
```

```
"onmt/TranslationResult.h"
```

```
namespace onmt
```

```
{
```

```
    TranslationResult::TranslationResult(const  
std::vector<std::vector<std::vector<std::string> > >& words,  
                                         const  
std::vector<std::vector<std::vector<std::vector<std::string> > >  
>& features,  
                                         const  
std::vector<std::vector<std::vector<std::vector<float> > > >&  
attention,  
                                         const  
std::vector<std::vector<float> >& score,  
                                         const  
std::vector<std::vector<size_t> >& count_unk_words)  
    : _words(words)  
    , _features(features)  
    , _attention(attention)  
    , _score(score)  
    , _count_unk_words(count_unk_words)  
    {  
    }
```

```
    const std::vector<std::string>&  
TranslationResult::get_words(size_t job_index, size_t  
translation_index) const  
    {  
        return _words[job_index][translation_index];  
    }
```

```
    const std::vector<std::vector<std::string> >&  
TranslationResult::get_features(size_t job_index, size_t  
translation_index) const  
    {  
        return _features[job_index][translation_index];  
    }
```

```
    const std::vector<std::vector<float> >&  
TranslationResult::get_attention(size_t job_index, size_t
```

```

translation_index) const
{
    return _attention[job_index][translation_index];
}

float TranslationResult::get_score(size_t job_index, size_t
translation_index) const
{
    return _score[job_index][translation_index];
}

size_t TranslationResult::get_count_unk_words(size_t job_index,
size_t translation_index) const
{
    return _count_unk_words[job_index][translation_index];
}

const std::vector<std::vector<std::string> >&
TranslationResult::get_words_job(size_t job_index) const
{
    return _words[job_index];
}

const std::vector<std::vector<std::vector<std::string> > >&
TranslationResult::get_features_job(size_t job_index) const
{
    return _features[job_index];
}

const std::vector<std::vector<std::vector<float> > >&
TranslationResult::get_attention_job(size_t job_index) const
{
    return _attention[job_index];
}

const std::vector<float>&
TranslationResult::get_score_job(size_t job_index) const
{
    return _score[job_index];
}

const std::vector<size_t>&
TranslationResult::get_count_unk_words_job(size_t job_index)
const
{
    return _count_unk_words[job_index];
}

```

```

size_t TranslationResult::count_job(size_t job_index) const
{
    return _words[job_index].size();
}

const std::vector<std::vector<std::vector<std::string> > >&
TranslationResult::get_words_batch() const
{
    return _words;
}

const
std::vector<std::vector<std::vector<std::vector<std::string> > >
>& TranslationResult::get_features_batch() const
{
    return _features;
}

const std::vector<std::vector<std::vector<std::vector<float> >
> >& TranslationResult::get_attention_batch() const
{
    return _attention;
}

const std::vector<std::vector<float> >&
TranslationResult::get_score_batch() const
{
    return _score;
}

const std::vector<std::vector<size_t> >&
TranslationResult::get_count_unk_words_batch() const
{
    return _count_unk_words;
}

size_t TranslationResult::count() const
{
    return _words.size();
}

bool TranslationResult::has_features() const
{
    return !_features.empty();
}
}

```

TranslatorFsctory.cc

```
#include
"onmt/TranslatorFactory.h"
```

```
namespace onmt
{
    std::unique_ptr<ITranslator> TranslatorFactory::build(const
std::string& model,
                                                    const
std::string& phrase_table,
                                                    const
std::string& vocab_mapping,
                                                    bool
replace_unk,
                                                    bool
replace_unk_tagged,
                                                    size_t
max_sent_length,
                                                    size_t
beam_size,
                                                    size_t
n_best,
                                                    bool
cuda,
                                                    bool
qlinear,
                                                    bool
profiling)
    {
        ITranslator* t = nullptr;

        t = new DefaultTranslator<float>(model,
                                        phrase_table,
                                        vocab_mapping,
                                        replace_unk,
                                        replace_unk_tagged,
                                        max_sent_length,
                                        beam_size,
                                        n_best,
                                        cuda,
                                        qlinear,
                                        profiling);

        return std::unique_ptr<ITranslator>(t);
    }

    std::unique_ptr<ITranslator>
    TranslatorFactory::clone(const std::unique_ptr<ITranslator>&
```

```
translator)
{
    ITranslator* t = new DefaultTranslator<float>(
        dynamic_cast<const
DefaultTranslator<float>&>(*translator));
    return std::unique_ptr<ITranslator>(t);
}

}
```