Mongolian Journal of Engineering and Applied Sciences, 2021, 3(1)

Компьютерын ухаан

ГАРАЛ НЭГ ҮГИЙГ ҮСГИЙН ДАРААЛАЛД ТУЛГУУРЛАСАН seq2seq ЗАГВАРААР ҮҮСГЭХ НЬ

З.Цолмон, Б.Хуягбаатар, Г.Амарсанаа*

МУИС, ХШУИС, Мэдээлэл компьютерын ухааны тэнхим, Машин оюуны лаборатори

Received on 2021.04.15; Revised on 2021.06.23; Accepted on 2021.06.27

*Холбоо баригч зохиогч: amarsanaag@num.edu.mn

Хураангуй

Аливаа хэл хооронд оршдог бичлэг болон дуудлага төстэй, ижил утгатай гарал нэг үгсийг тодорхойлох нь компьютер хэл шинжлэлийн даалгаварт хэрэглэх хэлний шинэ нөөцийг үүсгэх боломжийг олгож байна. Энэ ажлаар үгийн үсгийн дараалалд тулгуурлан гарал нэг үгийг автоматаар үүсгэх аргыг боловсруулахыг зорьсон юм. Бид төстэй болон өөр үсэгтэй таван хос хэлний хувьд гарал нэг үгсийг үүсгэх seq2seq гүн сургалтын загварыг гаргалаа. Сургасан загварыг үүсгэсэн үгийн тэмдэгтийн зөрүүгээр үнэлэхэд дунджаар 0.73 магадлалтайгаар гарал нэг үгсийг зөв үүсгэж чадсан.

Түлхүүр үг: гарал нэгтэй үг, sequence-to-sequence, LSTM, гүн сургалт

1 Удиртгал

Компьютер хэл шинжлэлийн зарим даалгаварт, жишээ нь, машин орчуулгад гарал нэг үгсийг ашиглах нь орчуулгын чанарыг сайжруулах боломжтой байна [1]. Үүний тулд аливаа хос хэлний хооронд тохиолддог нэг гарал үүсэлтэй (etymology), ижил утгатай үгс - гарал нэг үгсийг ашиглах нь үгийг оновчтой сонгох боломжийг олгоно. Бусад даалгаврын хувьд энэ нь хэлний нэмэлт нөөц болж ашигладах юм. Ийм үгсийн санг олон хос хэлний хооронд тодорхойлох нь маш их хүчин чармайлт, хугацаа шаардсан, ойрын хугацаанд бараг бүтэшгүй даалгавар юм.

Хэрэв өөр хэлний тухайн хоёр үг нь нэг гарал үүсэлтэй, утга ижил бол генетик [2], ямарч өөрчлөлтгүй шууд танигдахуйц бичигддэг, нэг хэлээс нөгөөд шууд утгаараа орж ирсэн бол зээлсэн гарал үг болдог. Зээлсэн гарал нэг үгс нь гарал нэг үгсийн тодорхойлолтод хамаарахгүй боловч хэрэглээнд гарал нэг үг шиг хэрэглэгдсээр ирсэн байдаг. Эндээс өөр өөр хэлний үгс нь яг ижил болон ойролцоо утга илэрхийлж мөн ойролцоо бичигддэг бол үнэн гарал нэг үгс гэдэг бол харин утга ондоо бол худал гарал нэг үгс, өөрөөр худал найзууд ч гэдэг [3, 4].

Хүн аливаа гадаад хэл дээрх өгүүлбэрт байгаа мэдэхгүй үгийн утгыг өмнө нь эзэмшсэн хэлний мэдлэгээ ашиглан тааж ойлгох боломжтой байдаг [5]. Жишээлбэл,

> *La vacuna experimental contra el coronavirus es segura y produce una respuesta inmune*

өгүүлбэрээс Англи хэл мэддэг хүн *experimental-experimental, coronavirus-coronavirus, produce-produces, inmune-immune* (Испани-Англи үг харгалзан) зэрэг үгсийг хялбархан таньж чадна.

Аливаа нэг хэлний үг нь өөр хэлний үгтэй гарал нэг болохыг тэдгээрийн тэмдэгтийн ижил байдлаар дүгнэж болох [6] ч үсэг ондоо хэлнүүдийн хувьд түвэгтэй юм. Энэ тохиолдолд тухайн хос үгийн дуудлагын тэмдэглэгээ хэрэгтэй болно.

Гарал нэг үгийн санг олон хэл хооронд үүсгэх нь компьютер хэл шинжлэлд хэрэглэх хэлний нөөцийг бүрдүүлэх асуудлын нэг юм. Ийм нөөцийг бүрдүүлэхэд тухайн хос үгийн гарал үүсэл, мөн хэлний-хээ хувьд илэрхийлэх үгийн утга, утгазүйн холбоо хамаарал, авиазүйн дуудлага зэрэг тал бүрийн мэдээллийг ашиглан үүсгэж болно.

Гарал нэг үгсийн хослол, түүний бичиглэлд тулгуурлан гарал нэг үгсийн санг өргөтгөн баяжуулах боломжтой эсэхийг энэ судалгаагаар туршиж үзлээ. Энэ ажлаар гарал нэг үгсийн хослолыг автоматаар, үнэн болон худал гарал нэг үгсийг ялгалгүй үүсгэхийг зорьсон юм. Үүний тулд эх хэл дээр байгаа өгүүлбэр - үгсийн цуваг зорилтот хэлний үгсийн цуваанд буулгах машин орчуулгын аргыг үгийн тэмдэгтийн түвшинд ашиглаж эх хэлний үг өгөгдөхөд зорилтот хэлний гарал нэг үгийг үүсгэн гаргасан болно. Үр дүнд нь 5 хос хэлний seq2seq загварыг боловсруулж үнэлэхэд гарал нэг үгсийг багадаа 55, ихдээ 86 хувийн магадлалтайгаар зөв үүсгэж чадсан.

Энэ өгүүллийн хоёрдугаар бүлэгт гарал нэг үгсийг таних, олж тодорхойлох, үгийн сан үүсгэх ижил төстэй зарим ажлын талаар тайлбарласан. Гуравдугаар бүлэгт гарал нэг үгсийг үүсгэх аргачлалыг, дөрөв болон тавдугаар бүлэгт туршилт, үр дүнгийн талаар бичсэн болно.

2 Гарал нэг үгсийг тодорхойлох асуудал

Компьютер хэл шинжлэлд, гарал нэг үг автоматаар үүсгэх судалгааг хийсээр байна. Хамгийн сүүлийн үеийн судалгаанууд [3], [7] тэмдэгтэд тулгуурласан машин орчуулгын аргыг гарал нэг хоршоо үгсийн сан ашиглан үүсгэсэн байдаг. Мөн машин орчуулгын аргаар [1] их хэмжээний, олон хэлний гарал нэг үгсийн сангийн бүлгийг үүсгэсэн байна. Ийм аргуудаар гарал нэг үг үүсгэх нь үндсэндээ цахим хэл шинжлэлд тодорхой хэл боловсруулах чадварыг сайжруулах сайн эх сурвалж болж байгаа хэдий ч эдгээр аргуудын чанараас шалтгаалан хэл шинжлэлийн зарим асуудлыг шийдвэрлэж чадахгүй хэвээр байна. Жишээ нь, Араб болон Еврэй хэлэнд эгшиггүй бичигддэг боловч машин орчуулгаар ийм үгийг эгшиггүй бичиж чаддагүй байна [8]. Дээрх ажлуудаас Бэйнборн нар [3] тэмдэгтэд тулгуурласан статистик машин орчуулгын аргаар боломжит орчуулгын хэд хэдэн хувилбарыг үүсгэж тэдгээрээс олох гэж байгаа үг нь хаана орсоноос нь хамаарч загвараа үнэлсэн байдаг. Харин Ву нарын [1] гарал нэг үгсийн их хэмжээний санг хүснэгтлэн үүсгэх судалгаа нь олон хэлний үгсийн сангуудаас тэмдэгтийн ижил байдлаар бүлэглэн цуглуулж үгсийн сангаас олдоогүй нөхөх шаардлагатай үгсийг тодорхойлж ашигласан байдаг.

Гарал нэг үгсийн сан CogNet[1] [9] нь нийт 338 хэлний 8.1 сая гарал нэг үгсийн хослол агуулсан том хэмжээний сан юм. Энэ санг үгийн утга, гарал үүсэл болон утга зүйн холбоо хамаарал зэрэг гурван мэдээллийг тооцож үүсгэжээ. Үүнд, үгийн утга, үгсийн утга зүйн хамаарлыг агуулсан үг зүйн сан - Universal Knowledge Core[2] [10] ашигласан ба энэхүү сан нь Princeton Wordnet [11], Open Multilingual Wordnet [12], PanLex [13], Wiktionary [14] сангуудыг нэгтгэж үүсгэсэн байна. Бид CogNet санг судалгаандаа ашигласан болно.

Энэ ажил нь олон хэлний олон үгсийн хослолтой томоохон хэмжээний сан дээр seq2seq [15] гүн сургалтын аргаар сургалт хийснээрээ өмнөх ажлуудаас ялгаатай. Хэдийгээр хос хэл бүрийн хувьд тус тусад нь загвар боловсруулж байгаа хэдий ч нэг аргачлал ашиглаж байгаа учир ямар ч хос хэлний хооронд ашиглах боломжтой загвар үүсгэж чадах юм. Харин хэлний нөөцийг баяжуулах асуудлын хувьд гүн сургалтын загвараар гарал нэг үгсийг автоматаар үүсгэх шинэ аргыг танилцуулж байгаа болно.

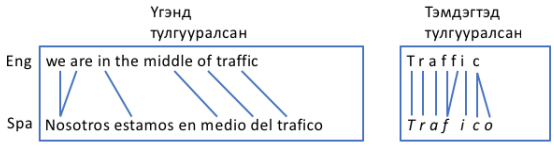
3 Аргачлал

Энэ бүлэгт гарал нэг үгсийг олох машин орчуулгын аргачлалын зохиомж болон машин сургалтын аргыг судалгааны асуудалд нийцүүлэн тайлбарласан болно. Мөн үнэлгээний аргачлалыг үзүүлсэн.

[1]https://github.com/kbatsuren/CogNet
[2]http://ukc.datascientia.eu/

3.1 Тэмдэгтэд тулгуурласан машин орчуулга

Энэ ажлын үндсэн даалгавар бол эх хэлээр өгөгдсөн үгийн хувьд зорилтот хэлэнд тохирох гарал нэг үгийн зөв олох юм. Үүний тулд өгөгдсөн үгийг тэмдэгтийн цуваа (үсэг) гэж үзээд машин орчуулгаар зорилтот хэлний хувьд тохирох тэмдэгтийн цуваг үүгэх даалгавар болгон тодорхойлсон. Өөрөөр хэлбэл, тэмдэгтийн цуваа болон үгсийн хослолуудад гарал нэг үгсийг илэрхийлэх ямарваа мэдлэг байгаа гэж таамаглан тэмдэгтийн дарааллыг орчуулах ажил болон хувиргаж буй хэрэг. Тэмдэгтэд тулгуурласан арга нь үгийн бичигдэж байгаа тэмдэгтийн цуваг оролтод оруулж орчуулах хэлний үгийг мөн тэмдэгтийн дарааллаар үүсгэх (Зураг 1) юм. Зураг 1-т харуулсан жишээ дээр Англи хэлний



Зураг 1: Үг болон тэмдэгтэд тулгуурласан машин орчуулгын дүрслэл

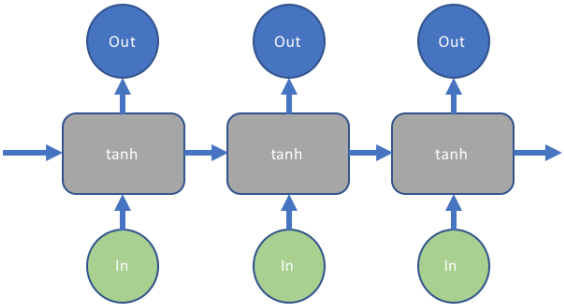
Traffic гэдэг үгийг Испани хэлний харгалзах үгэнд буулгахад Англи үгийн *ff* хос үсэг нь *f*, *c* үсэг нь *co* үсгүүд болж харгалзан хувирна. Гэхдээ нийт хэлний хооронд ийм хувирал нь тодорхой дүрэмд захирагдахгүй. Seq2seq гүн сургалтын архитектур нь Long short-term memory (LSTM) эсийг ашигласнаар оролтод өгч байгаа үсгийн дарааллаас ямар үсгүүд харгалзах хэлний үсгүүд болон тэмдэгт болж хувирах эсэхийг оролт гаралтын үсгүүдийн дарааллын зүй тогтлоос тооцоолж сурадцдаг. Тооцоолол хийхдээ дарааллын үсэг бүрийг харгалзах хэлний хооронд хувиргахад жин оноож тооцоолдог. Энд зөвхөн үсэг бүрийн хувьд тооцоолохгүй ойр хамт орж байгаа үсэг бүрийн дарааллаас хамаарсан жинг тооцоолж олдог. Учир нь LSTM нь дарааллын бүх элементүүдийг хамруулан тооцоолдгоороо давуу талтай байдаг. Дараалал хэтэрхий урт болсон тохиолдолд градиент замхрах асуудал байдаг хэдий ч үсгийн түвшинд энэ дутагдал нь нөлөөлөхгүй юм. Тиймээс үндсэн зорилго болох аливаа хэлний гарал нэг үгнээс өөр хэлний гарал нэг үг үүсгэх ажилд тохирох юм. Тэмдэгтэд тулгуурласан аргын нэг давуу тал нь ямарч үсгийн дараалал үүсгэх боломжийг олгодгоороо үгэнд тулгуурласан машин орчуулгын аргуудаас давуутай байдаг. Энэхүү аргаар товчилсон үгнээс зөв үг бүтээх боломжтой бөгөөд дараа дараагийн судалгаанд ашиглагдах бүрэн боломжтой юм. Өмнөх судалгаанд ашигласан арга нь хэд хэдэн боломжтой үгсийн хувирал үүсгэн түүн дотроос зөв хувирсан үг байгаа эсэх, тухайн зөв үг нь хэдэд эрэмблэгдэж байгаагаар үнэлгээ [3] өгч байсан бол манай загвар хамгийн сайн тохирох ганц үгийн хувирал санал болгодог загвар тул мөн өмнөх ажлуудтай үр дүнгээ харьцуулах боломжгүй юм.

3.2 Seq2seq гүн сургалтын арга

Seq2seq машин сургалтын загвар нь статистикт биш Recurrent Neural Network (RNN) гэгдэх гүн сургалтын архитектурыг ашигладаг. Энэхүү арга нь цуваа оролтыг боловсруулан өөр цуваа оролтод хувирган гаргах үндсэн үйлдэлтэй, анх Google компани машин орчуулгад ашигласан бөгөөд машин орчуулгын хөгжлийг шинэ шатанд гаргасан байдаг. Seq2seq загвар нь цахим хэл боловсруулах маш олон хэрэглээнд одоо ашиглагдаж байна. Тухайлбал, зураг гарчиг оноох, хугацааны цуваа өгөгдөл дээрх таамаглах хийхэд зэрэг болно. Seq2seq машин сургалтын загвар нь RNN-ийн үндсэн архитектурыг ашигладаг бөгөөд LSTM [16] болон Gated Recurrent units (GRU) гэсэн хоёр эсийн архитектурыг нейроны эс байдлаар ашигладаг. LSTM нь RNN архитектурын гол дутагдал болох оролтын цуваа хэтэрхий урт бол градиент замхрах асуудлыг тодорхой хэмжээгээр шийдэж өгснөөрөө давуу талтай болсон байдаг боловч хэтэрхий урт цувааны хувьд бүрэн шийдэж чадаагүй.

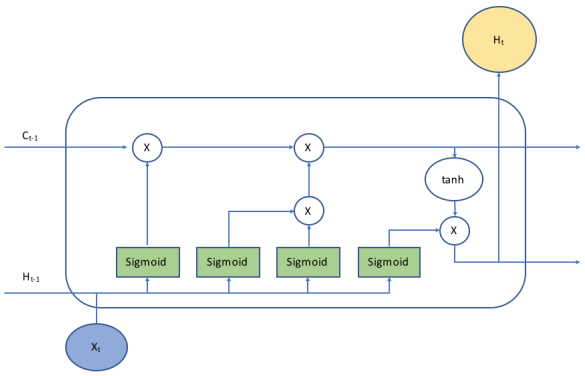
Seq2seq архитектурын үндсэн загвар нь энкодер, декодер гэсэн хоёр хэсгээс бүрдэх бөгөөд оролтын цуваа тэмдэгт энкодероор боловсруулагдан декодероор зорилтод цувааны хувиргалтаар гарна. Ингэхдээ энкодер болон декодерын үндсэн эсэндээ LSTM болон GRU ашигласан байдлаараа ялгардаг.

LSTM архитектурын бүтэц нь стандарт RNN архитектуртай төстэй боловч RNN дутагдлыг нөхсөн байдлаараа ялгардаг. Хэдийгээр LSTM архитектур



Зураг 2: RNN архитектур

нь RNN-тэй төстэй архитектуртай боловч RNN архитектур шиг ганц давхрага (зураг 2 харуулсан *tanh*) ашиглахгүй харин бүр төвөгтэй гэж хэлж болох 4 давхрага ашигладаг. Зураг 3-т харуулсан шиг LSTM эсийн архитектур нь *tahn* функцээс гадна *sigmoid* 3 функц ашиглаж байна. Үүнээс гадна векторын үржвэрийг ашигласан байдаг. Энд C үсгээр тэмдэглэсэн тэмдэглэгээ нь RNN сүлжээний эсийн төлөвийг илэрхийлдэг. Эсийн төлөвийг дараа дараагийн эсийн төлөвтөө өөрчлөх төлөвийн мэдээллийг нэмэх байдлаар дамжуулж нийт цувааны хувьд тооцоолол хийхэд ашигладаг. LSTM эс доторх үндсэн ажиллагаа нь нэгдүгээрт эсийн төлөвөөс шаардлагагүй мэдээллийг мартах, хоёрдугаарт эсийн төлөвт мэдээлэл нэмэх, гуравдугаарт гаралтыг тооцоолох гэсэн гурван үндсэн үйлдлээс бүрддэг. LSTM эсийн



Зураг 3: LSTM эсийн архитектур

эхний *sigmoid* функц нь мартах буюу "forget gate level"дууддаг. Энэ нь өмнөх эсийн төлөв нь хэр чухал болохыг шиддэг. *Sigmoid* функц нь 0 ... 1 хооронд утга авах тул өмнөх эсийн төлөвийг хадгалах эсвэл алга болгох эсэхийг өмнөх эсийн төлөв болох H_{t-1} болон оролтын утга болох X_t-ийн утгуудаар тооцоолж шийднэ.

$$f_t = \delta(W_f \cdot [H_{t-1}, x_t] + b_f) \tag{1}$$

f_t = forget gate-ийн вектор
W_f = forget gate дээрх хувьсагчдын жингийн матриц
b_f = forget gate дээрх загварын bias вектор
H_{t-1} = өмнөх эсийн далд төлөвийн вектор
x_t = оролтын вектор

Шаардлагагүй мэдээллээ алга болгосны дараа LSTM эс нь ямар өгөгдөл эсийг төлөвт нэмэгдэхийг шийдвэрлэдэг. Энэ үйлдлийг хоёрдох *sigmoid* болон *tanh* функцийн тусламжтайгаар гүйцэтгэдэг. Нэг болон хоёрдох *sigmoid* функцийн түвшнийг оролтын хаалганы давхрага гэдэг бөгөөд эсийн төлөвт ямар утга нэмэгдэхийг шийднэ. Харин *tanh* давхрага нь гаралтын төлөвт нэмэгдэх утгуудын векторыг үүсгэж өгдөг.

$$i_t = \delta(W_i \cdot [H_{t-1}, x_t] + b_i) \tag{2}$$

i_t = оролтын/засварлалтын вектор
W_i = оролтын хувьсагчдын жингийн матриц
b_i = оролтын bias вектор
H_{t-1} = өмнөх эсийн далд төлөвийн вектор
x_t = оролтын вектор

$$C_t = \tanh(W_C \cdot [H_{t-1}, x_t] + b_C) \tag{3}$$

C_t = Эсийн төлөвийн вектор
W_C = Эсийн төлөвийн жингийн матриц
b_C = Эсийн төлөвийн bias вектор
H_{t-1} = өмнөх эсийн далд төлөвийн вектор
x_t = оролтын вектор

Дээрх бүх шинэ төлөвийн мэдээллийг ашиглан гаралтын шинэ төлөвийг дараах томьёогоор тооцоолдог.

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{4}$$

f_t = forget gate вектор

C_{t-1} = Эсийн өмнөх төлөвийн жингийн матриц

i_t = Оролтын вектор

C_t = Эсийн төлөвийн вектор

LSTM эсийн гаралтын утгыг тооцоолох алхам нь хамгийн сүүлд шаардлагатай болдог. Үүнд гурав дахь *sigmoid* болон нэмэлт *tanh* шүүлтүүрийг ашигладаг. Гаралтын утга нь эсийн төлөвийн утгаас хамаарах боловч энэхүү утга нь *sigmoid* шүүлтүүрээр ордог. Үндсэндээ *sigmoid* функц нь эсийн төлөвийн аль хэсэг нь гаралтын утгад нөлөөтэй эсэхийг шийддэг гэсэн үг юм. Өөрөөр хэлбэл эсийн төлөвийн утгыг *tanh* шүүлтүүрээр оруулж гарсан үр дүнг гуравдах *sigmoid* функцийн гаралтаар үржүүлнэ гэсэн үг юм.

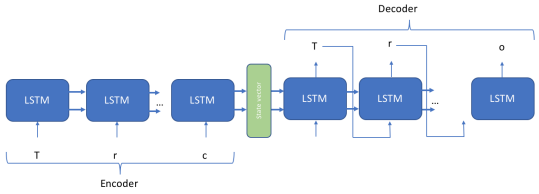
$$o_t = \delta(W_o \cdot [H_{t-1}, x_t] + b_o) \tag{5}$$

o_t = эсийн гаралтын вектор

$$h_t = o_t * \tanh(C_t) \tag{6}$$

h_t = эсийн далд төлөвийн вектор

Тэмдэгтэд суурилсан seq2seq загварыг доорх зураг 4-т харуулав. Гарал нэг үгс үүсгэх гүн сургал-



Зураг 4: Тэмдэгтэд тулгуурласан seq2seq загварын архитектур

тын загвар хангалттай сайн ажиллаж байгаа эсэхийг үнэлэхдээ үг хоорондоо хэр зөрүүтэй бичигдсэн эсэхийг хэмждэг Левенштэйн зайг шууд ашиглах боломжгүй юм. Учир нь Левенштэйн зай нь өгөгдсөн хоёр үгний нэгнээс нь нөгөөг үүсгэхэд хэдэн үсэг нэмэх, хасах тоог тооцдог бөгөөд олон үгнүүдийг шалгаж дундаж үзүүлэлт гаргах боломжгүй арга юм [17]. Тиймээс бид нормчилсон Левенштэйн зайг ашиглаж угсарсан загвар хэр зөв үг үүсгэж байгааг үнэлэв.

$$normLev = 1 - \frac{lev(word_{tar}, word_{prod})}{maxlen(word_{tar}, word_{prod})} \tag{7}$$

Энд $word_{tar}$ нь үгсийн санд буй жинхэнэ гарал нэг үг бол $word_{prod}$ нь сургасан загвараас үүсгэсэн гарал нэг үг юм. Нормчилсон Левенштэйн зайг олохдоо өгөгдсөн хоёр үгний Левенштэйн зайг тухайн үгнүүдийн хамгийн урт үгийн уртад хуваан 1-ээс хассанаар үгсийн тэмдэгтийн зөрүү 0 ... 1 нормчилогдож болохыг томьёо 7-д илэрхийлэв. Ингэснээр ямар ч урттай байсан үгнүүдийн үсгийн зөрүү нэг утгатай хэмжээсээр хэмжигдэх боломжтой болно. Нормчилсон Левенштэйн зай нь хоёр үг ижил бол 1, харин үсгийн зөрүү ихсэх тусам 0 рүү ойр утга авна.

4 Туршилт

Зураг 4 дээр тэмдэгтэд тулгуурласан seq2seq гүн сургалтын архитектурыг бүдүүвчлэн харуулав. Бид энэхүү загварыг машин сургалтын фрэймворк Tensorflow 2.0 болон машин сургалтын фрэймворктэй ажиллах хялбар интерфэйс болох Keras ашиглан python хэл дээр хэрэгжүүлсэн болно. Сургалтын өгөгдлийг CogNet 2.0 сангийн хэлний хослолуудын өгөгдлөөс (хүснэгт 1) сургалтын өгөгдлийг бэлдсэн болно. Сургалтад ашиглах хэлний сонголтыг хийхдээ түгээмэл хэлнүүд болон үсэг ондоо бичигддэг хэл гэсэн сонголтоор эдгээр хэлнүүдийг сонгосон болно. Сонгож авсан хэлнүүдээс Англи-Франц

Хүснэгт 1: Сургалтад ашигласан хэлнүүдийн хослол

|д|	Эх	Зорилтот	Хосын тоо
1	Англи	Франц	54,365
2	Англи	Испани	41,119
3	Англи	Герман	26,377
4	Англи	Орос	9,475
5	Англи	Грек	5,564

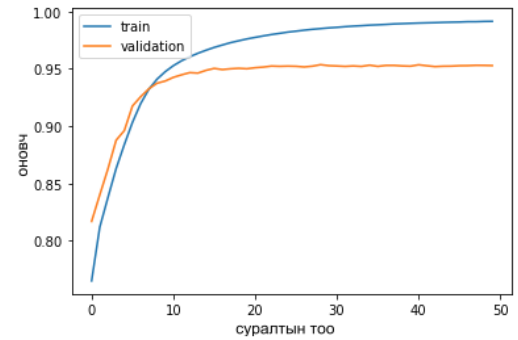
хэлний гарал нэг үгсийн хослол хамгийн их бөгөөд үсэг ижил хослол болно. Харин Англи-Орос, Англи-Грек хэлний хослолууд нь үсэг ондоо боловч гараг нэг үгсийн хослол бусад сонгож авсан хэлнүүдээс харьцангуй цөөн байгааг харж болох юм.

4.1 Сургалтын өгөгдөл

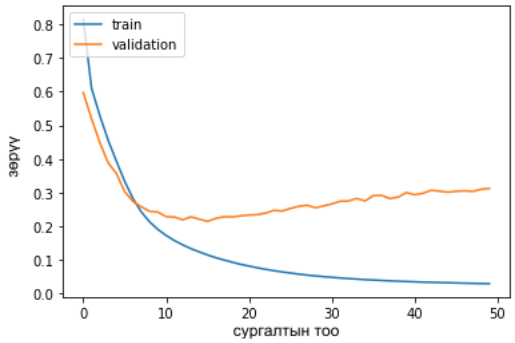
CogNet 2.0 үгсийн санд гарал нэг үгсийн хослол нь нэг нэг үг биш байсан бөгөөд зарим тохиолдолд 4-5 үгнүүдээс бүтсэн хослол байсан болно. Тиймээс сургалтын өгөгдлийг бэлдэхдээ хэлний хослол бүрийн хувьд 1 болон хоёр үгнээс бүтсэн гарал нэг үгсийн хослолыг үгсийн сангаас түүвэрлэн авсан. Түүвэрлэн авсан үгсээ оролтын болон гаралтын хэлний хувьд тэмдэгтээр салгаж ялгаатай тэмдэгтийн тоо, хамгийн урт үгийн тэмдэгтийн уртыг хэл бүрийн хувьд олж тэмдэгт бүрийг векторт хувирган seq2seq машин сургалтын оролт болон гаралтын хэмжээс болгон бэлдсэн. Сургалтад нийт өгөгдлийн 80% -ийг оруулж 20% -ийг шалгалтын өгөгдөл болгон ашигласан болно.

4.2 Загварыг сургах параметрын тохиргоо

Тэмдэгтэд тулгуурласан seq2seq загварыг өгөгдөл дээр сургахад хэд хэдэн загварын тохируулах пара-



(a) Загварын оновч



(b) Загварын алдаа

Зураг 5: Англи-Франц хэлний тэмдэгтэд тулгуурласан seq2seq загварын сургалтын үр дүн 50 epoch явагдахад (а) оновчлол (б) алдааны утга өөрчлөгдсөн байдал

метрийн Hyperparameter утгаар туршихад хамгийн боломжтой утга Batch size буюу сургалтын нэг алхам хийхэд ашиглах өгөгдлийн хэмжээг 64 байх, Latent dimensionality хэмжээ буюу энкодерийн гаралтын векторын утга 256, харин нийт сургалт хийх давтамжийн тоог 50 (epoch) гэсэн утга байв. Хэл бүрийн хослолын хувьд LSTM эсийн давхрагын тоо оролтын болон гаралтын тэмдэгтийн тооноос хамаарч encoder болон decoder дээр өөр өөр болно. Сургалтын загварын оптимайзераар RMSPROP, алдааг үнэлэх функцээр categorical cross entropy ашигласан болно. Зураг 5 дээр (а) график нь загварын оновч буюу загвар зөв үр дүн гаргаж байгаа магадлалыг сургалтын өгөгдөл болон баталгаажуулах өгөгдөл дээр нийт сургалтын хугацаанд 50 удаа явагдахад хэрхэн өөрчлөгдөж байгааг харуулсан байна. (b) графикт алдааны хэмжээ мөн сургалтын өгөгдөл болон баталгаажуулах өгөгдөл дээр 50 удаагийн алхамд хэрхэн өөрчлөгдөж байгааг харуулав. Сургалтыг Макинтош үйдлийн систем, интелийн 2.4GHz CPU, 8Gb санах ойнтой машин дээр хийсэн бөгөөд нэг загварын сургах дундаж хугацаа 4 цаг орчим болж байв.

5 Үр дүн

Бидний угсарсан тэмдэгтэд тулгуурласан seq2seq загварыг хэл бүрийн хувьд сургасан бөгөөд нийт 5 загвар үүссэн болно. Энэхүү загварын архитектур ямарч хэлний хослолын хувьд ажиллах боломжтой харуулав. Зураг 5-д Англи-Франц хэлний хослол дээр сургалтын явцын үнэлгээг харуулав. Сургасан загвар бүрийн хувьд бид нормчилсон Левенштэйн зайг тооцоолох замаар загвар гарал нэг үгсийг хэр сайн үүсгэж байгаа эсэхийг үнэлэж хүснэгт 2-т харуулав. Номрчилсон Левенштэйн зай нь үсгийн ялгаатай байдлаас хамаарахгүй үр дүн үзүүлсэн байгаа нь хүснэгт 2-оос харагдаж байна. Туршилтын үр дүнг харахад номрчилсон Левенштэйн зай Англи-Грек хэлнээс бусад хэлний хослол 0.72-гоос дээш, нийт хос хэлний хувьд дунджаар 0.73 оновчтой байна.

Хүснэгт 2: Загварыг үнэлэх хэлний хослол бүрийн нормчилсон Левенштэйн зайн утгууд

	Хослол	Үнэлгээ
Төстэй үсэгтэй	Англи-Франц	0.77
	Англи-Испани	0.72
	Англи-Герман	0.86
Өөр үсэгтэй	Англи-Орос	0.76
	Англи-Грек	0.55

6 Дүгнэлт

Энэ ажлаараа гарал нэг үгс үүсгэх тэмдэгтэд тулгуурласан машин сургалтын seq2seq архитектурыг танилцуулав. Бидний дэвшүүлсэн арга нь хэл болон үсгийн ялгаатай хослолд хангалттай сайн ажиллаж байгааг харуулж байна. Учир нь нэгээс бусад хэлний хослолын хувьд 0.72 болон 0.86-аас дээш номрчилсон Левенштэйн утгатай байгаа нь анхны загварын хувьд сайн үзүүлэлт гэж дүгнэж байна. Өгүүлбэрийн бүтэц дээр буюу үгийн дараалал дээр сайн ажиллаж танигдсан seq2seq аргыг тэмдэгтийн цувааны бүтэц дээр туршиг ажиллуулж гарал нэг үгсийг олж тодорхойлох боломжийг туршиж байгаа нь энэ ажлын нэг онцлог, шинэлэг юм. Цаашид загварыг сайжруулахын тулд сургалтын өгөгдлийг улам чанаржуулж мөн бусад архитектурыг угсарч турших шаардлагатай гэж үзэж байна. Тухайлбал, seq2seq загвар дээр attention давхрага оруулж ирэх мөн хамгийн сүүлийн үеийн гүн сургалтын нэг ололт болох transformer архитектурыг дараа дараагийн судалгаандаа ашиглах бүрэн боломжтой юм.

Зохиогчийн оролцоо

З. Цолмон алгоритм болон хөгжүүлэлтийг хийж гүйцэтгэн туршилтыг хийж өгүүлэл бичсэн. Б. Хуягбаатар, Г. Амарсанаа өгүүлэл бичихэд оролцож мөн зөвөлгөө өгч ажилласан болно.

Ашиг сонирхолгүйн баталгаа

Ашиг сонирхолын зөрчилгүй болохыг үүгээр баталж байна.

Ашигласан ном

[1] Wu W, Yarowsky D. Creating large-scale multilingual cognate tables. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018); 2018. .

[2] Crystal D. A dictionary of linguistics and phonetics. vol. 30. John Wiley & Sons; 2011.

[3] Beinborn L, Zesch T, Gurevych I. Cognate production using character-based machine translation. In: Proceedings of the Sixth International Joint Conference on Natural Language Processing; 2013. p. 883–891.

[4] Inkpen D, Frunza O, Kondrak G. Automatic identification of cognates and false friends in French and English. In: Proceedings of the International Conference Recent Advances in Natural Language Processing. vol. 9; 2005. p. 251–257.

[5] Ringbom H. On L1 transfer in L2 comprehension and L2 production. Language learning. 1992;42(1):85–112.

[6] Montalvo S, Pardo EG, Martinez R, Fresno V. Automatic cognate identification based on a fuzzy combination of string similarity measures. In: 2012 IEEE International Conference on Fuzzy Systems. IEEE; 2012. p. 1–8.

[7] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. arXiv preprint arXiv:150807909. 2015.

[8] Karimi S, Scholer F, Turpin A. Machine transliteration survey. ACM Computing Surveys (CSUR). 2011;43(3):1–46.

[9] Batsuren K, Bella G, Giunchiglia F. Cognet: A large-scale cognate database. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics; 2019. p. 3136–3145.

[10] Giunchiglia F, Batsuren K, Freihat AA. One world–seven thousand languages. In: Proceedings 19th International Conference on Computational Linguistics and Intelligent Text Processing, CiCling2018, 18-24 March 2018; 2018. .

[11] Miller GA. WordNet: a lexical database for English. Communications of the ACM. 1995;38(11):39–41.

[12] Bond F, Foster R. Linking and extending an open multilingual wordnet. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); 2013. p. 1352–1362.

[13] Kamholz D, Pool J, Colowick SM. PanLex: Building a Resource for Panlingual Lexical Translation. In: LREC. Citeseer; 2014. p. 3145–3150.

[14] Kirov C, Sylak-Glassman J, Que R, Yarowsky D. Very-large scale parsing and normalization of wiktionary morphological paradigms. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16); 2016. p. 3121–3126.

[15] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. arXiv preprint arXiv:14093215. 2014.

[16] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997;9(8):1735–1780.

[17] Schepens J, Dijkstra T, Grootjen F. Distributions of cognates in Europe as based on Levenshtein distance. Bilingualism: Language and cognition. 2012;15(1):157–166.