ISSN 1829-0043 ՀԱՅԱՍՏԱՆԻ ՃԱՐՏԱՐԱԳԻՏԱԿԱՆ ԱԿԱԴԵՄԻԱՅԻ ԼՐԱԲԵՐ (ՀՃԱԼ). 2018. Հ. 15, N 4

ՀՏԴ 621.396

ԻՆՖՈՐՄԱՏԻԿԱ, ԷԼԵԿՏՐՈՆԻԿԱ ԵՎ ԳԻՏԱԿԱՆ ՍԱՐՔԱՇԻՆՈՒԹՅՈՒՆ

Ս.Գ. Սարգսյան, Ա.Ա. Խառատյան, Ա.Ս. Հովակիմյան

ՀԱՅԵՐԵՆ ՏԵՔՍՏԻ ՏՈՆԱՅՆՈՒԹՅԱՆ ՎԵՐԼՈՒԾՈՒԹՅԱՆ ՄԵԹՈԴՆԵՐ

Մշակվել են մեխանիզմներ՝ որոշ սոցիալական ցանցերից հայերեն տեքստերի հավաքագրման, ֆորմատավորման և վեկտորային տեսքով ներկայացման համար: Դրանք օգտագործվել են տրամաբանական ռեգրեսիա և պատահական անտառ մոդելների ստեղծման և ծրագրորեն իրականացման համար: Կառուցվել են հայերեն լեզվով տեքստերի համար մինչ այժմ գոյություն չունեցող կորպուսներ (twitter-corpus.scv և facebook-corpus.scv): Օգտատերերի համար մշակվել են հայերեն լեզվով նոր գրառումներ ներբեռնելու և կառուցված մոդելները օգտագործելու ծրագրային միջոցներ, որոնք տեղադրվել են github.com կայքում https://github.com/hripman/Armenian-corpus:

Առանցքային բառեր. *տրամաբանական ռեգրեսիա, տեքստի տոնայնության վերլուծություն, պատահական անտառ, ուսուցչի կողմից ուսուցում, ուսուցանող բազմություն, սխալանքի մատրից:*

Ներածություն: Վերջին շրջանում, Internet տեղեկատվության հարստացման հետ կապված, արդիական է դարձել տեքստի (գրառում, փաստաթուղթ) տոնայնության վերլուծության խնդիրը (Sentiment Analysis) [1-3]: Այդ խնդրի լուծման տարածված եղանակներից է ուսուցչի կողմից մեքենայական ուսուցման մեթոդները, որոնցից աշխատանքում կիրառվել են տրամաբանական ռեգրեսիա (Logistic regression) և պատահական անտառ (Random Forest) մոդելները: ՈՒսումնասիրվել է որոշ սոցիալական կայքերի հայերեն մեկնաբանությունների էմոցիոնալ նկարագիրը, որոնք հիմք են հանդիսացել ստեղծված մոդելների աշխատանքի համար: Կատարվել է հայերեն տեքստերի տոնայնության վերլուծություն նշված մոդելների միջոցով:

Հոդվածում ներկայացվող ծրագրային միջոցների օգտագործմամբ օգտատերը հնարավորություն կունենա, առանց կարդալու սոցիալական ցանցերից հավաքված կարծիքները, իմանալ դրանց դրական կամ բացասական լինելը, խնայելով կարդալուն հատկացվող ժամանակը:

Ուսուցչի կողմից ուսուցման մեթոդները: Այս մեթոդներն ունեն աշխատանքի երկու հիմնական փուլ. սովորել ուսուցման համար նախատեսված տվյալների հիման վրա և կատարել դասակարգում արդեն նոր կամ թեստային տվյալներով: Սովորել ասելով նկատի ունենք այն գործընթացը, երբ կառուցվում է մոտարկող ֆունկցիա:

Փաստաթղթի դասակարգումը տեղեկատվական որոնիչի (Search Engine) հիմնական խնդիրներից է, որի նպատակն է փաստաթուղթը ներառել որևէ դասի մեջ՝ ելնելով դրա բովանդակությունից: Ի տարբերություն կլաստերացման խնդրի՝ այդ դասերը նախապես հայտնի պետք է լինեն: Դիցուք տրված է դասերի $C = \{c_1, \dots, c_{|C|}\}$ և փաստաթղթերի $D = \{d_1, \dots, d_{|D|}\}$ վերջավոր բազմությունները և անհայտ Φ ֆունկցիան, որը կամայական $< c, d > (c \in C, d \in D)$ զույգի համար որոշում է՝ համապատասխանում է արդյոք d փաստաթուղթը c դասին, թե՝ ոչ ($\Phi: D \times C \to \{0,1\}$): Դասակարգման խնդիրն է՝ գտնել Φ-ն մոտարկող $\Phi^{\prime}$ ֆունկցիա: Այդ $\Phi^{\prime}$ ֆունկցիան իրականացնում է դասակարգիչը**:**

Մեքենայական ուսուցման նախնական փուլում առանձնացվում է փաստաթղթերի ենթաբազմություն $Q = \{d_1, \dots, d_{|Q|}\} \subseteq D$: Ընդ որում, կամայական $< d_i, c_j > \in Q \times C$ զույգի համար Φ-ն հայտնի է: Q փաստաթղթերի ենթաբազմությունը բաժանվում է երկու չհատվող բազմության.

- $T_r = \{d_1, \dots, d_{|T_r|}\}$ ուսուցողական՝ այն բազմությունը, որի միջոցով որոշվում է $\Phi^{\prime}$ դասակարգիչը,
- $T_e = \{d_{|T_r|+1}, \dots, d_{|Q|}\}$ թեստային՝ այն փաստաթղթերի բազմությունը, որի վրա փորձարկվում է $\Phi^{\prime}$ դասակարգիչը:

Ամեն մի թեստային փաստաթուղթ տրվում է $\Phi^{\prime}$ դասակարգչին որպես մուտքային տվյալ, այնուհետև ստացված $\Phi^{\prime}(d_i, c_j)$ արդյունքը համեմատվում է իրական $\Phi(d_i, c_j)$ ֆունկցիայի արժեքի հետ: Որքան համընկնումները շատ լինեն, այնքան դասակարգիչը կհամարվի արդյունավետ աշխատող:

$\Phi^{\prime}$ դասակարգչի արժեքից կախված՝ տարբերում են երկու տեսակի դասակարգում՝ ճշգրիտ ($\Phi^{\prime}: D \times C \to \{0,1\}$) և հավանականային ($\Phi^{\prime}: D \times C \to [0,1]$): Դասակարգումը կոչվում է ճշգրիտ, եթե կամայական (d, c) զույգին համապատասխանում է բուլյան արժեք՝ 1 կամ 0, այսինքն՝ տվյալ փաստաթուղթը պատկանում է այդ դասին, թե՝ ոչ: Հավանականային դասակարգման դեպքում կամայական (d, c) զույգին համապատասխանացվում է [0,1] միջակայքից թիվ, որը ցույց է տալիս d փաստաթղթի c դասին

պատկանելու հավանականությունը [3, 4]:

Դասակարգման ժամանակ կատարվում է փաստաթղթերի ինդեքսավորում: Դա այն գործընթացն է, որի արդյունքում փաստաթղթերը բերվում են միևնույն ձևաչափի: Քանի որ փաստաթղթերի ծավալները սովորաբար բավականին մեծ են, անհրաժեշտ է լինում դրանցից հեռացնել այն բառերը (թերմերը), որոնք էմոցիոնալ նկարագիր չեն պարունակում (stop words)՝ օժանդակ բայեր, կապեր, շաղկապներ և այլն:

Հաջորդ փուլում փաստաթղթերը բերվում են այնպիսի ֆորմատի, որի հետ դասակարգման ալգորիթմները արդեն կարող են աշխատել, այսինքն՝ ներկայացվում են վեկտորային տեսքով: Կամայական փաստաթուղթ ներկայացվում է թերմերի (բառերի) բազմությամբ: Յուրաքանչյուր թերմի համար որոշվում է w_{ij} կշիռ $d_j \in D$ փաստաթղթի նկատմամբ: Հետևաբար ամեն մի փաստաթուղթ կարելի է ներկայացնել որպես կշիռների վեկտոր՝ $d_j \rightarrow =< w_{1j} \dots w_{|T|j} >$: Փաստաթղթերի կշիռները նորմավորվում են այնպես, որ $0 < w_{ij} < 1$, որտեղ $\forall i,j: 0 \leq i \leq |T|, 0 \leq j \leq |D|$:

Տեքստի տոնայնության դասակարգման մեթոդի գնահատման համար կա երկու հիմնական մոտեցում: Առաջինը դասակարգման տարբեր եղանակների միմյանց հետ համեմատումն է, երկրորդը՝ տվյալ ալգորիթմի գնահատումը որևէ չափողականությամբ: Ամենատարածվածը F1 չափողականությունն է, որը ներառում է երկու հիմնական հասկացություն՝ ճշգրտություն (precision) և լրիվություն (recall), որոնք ստացվում են սխալանքի մատրիցից (confusion matrix) [5, 6]:

Ճշգրտությունը դրական (բացասական) օրինակների համար ցույց է տալիս ճշգրիտ դրական (բացասական) տոնայնությամբ դասակարգված օրինակների քանակի և համակարգի կողմից դրական (բացասական) դասակարգված օրինակների հարաբերությունը: Լրիվությունը դրական (բացասական) օրինակների համար ցույց է տալիս ճշգրիտ դասակարգված դրական (բացասական) տոնայնություն ունեցող օրինակների քանակի և բոլոր դրական (բացասական) տոնայնությամբ օրինակների հարաբերությունը: Ճշգրտություն և լրիվություն հատկությունների հարմոնիկ միջինով որոշվում է F_β չափողականությունը [5].

$$F_\beta = (1+\beta^2) * \frac{precision * recall}{(\beta^2 * precision) + recall}:$$

β պարամետրն ընտրվում է՝ ճշգրտության և լրիվության գնահատականներից կախված: Եթե β-ն 1 է, ապա կունենանք F1 կախվածություն:

Տրամաբանական ռեգրեսիայի օգտագործումը տեքստի տոնայնության վերլուծության խնդրում: Տրամաբանական ռեգրեսիայի կիրառման դեպքում կախյալ y փոփոխականն ընդունում է երկու արժեք՝ 1 կամ 0 [4]: Անկախ փոփոխականների $\boldsymbol{x_1, x_2, \dots, x_n}$ բազմության հիման վրա հաշվարկվում է տվյալ պատահույթի հավանականությունը: Պարզության համար $\boldsymbol{x_0}$ հատկությունը վերցվում է հավասար 1-ի: Ենթադրվում է, որ $\boldsymbol{y = 1}$ պատահույթի հանդես գալու հավանականությունը՝

$$P\{y = 1|x\} = f(z),$$

որտեղ $z = w^T x = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n$:

x-ը և w-ն համապատասխանաբար $x_0, x_1, \dots, x_n$ անկախ փոփոխականների և $w_0, w_1, \dots, w_n$ ռեգրեսիայի պարամետրերի (կշիռների) վեկտոր սյուներն են, իսկ $f(z)$-ը՝ տրամաբանական ֆունկցիան (սիգմոիդ, logit-ֆունկցիան) [4] .

$$f(z) = \frac{1}{1 + e^{-z}};$$

Տեքստի տոնայնության վերլուծության խնդրում $\boldsymbol{w_1, w_2, \dots, w_n}$ ռեգրեսիայի պարամետրերը գտնելու համար անհրաժեշտ է կազմել ուսուցանող բազմություն (training set), որը կազմված է անկախ փոփոխականների և դրանց համապատասխան y կախյալ փոփոխականների հավաքածուից: Ֆորմալ դա $\left(\boldsymbol{x^{(1)}, y^{(1)}}\right), \dots, \left(\boldsymbol{x^{(m)}, y^{(m)}}\right)$ զույգերի բազմությունն է, որտեղ $\boldsymbol{x^{(i)} \in R^n}$-ը անկախ փոփոխականների արժեքների վեկտորն է, իսկ $\boldsymbol{y^{(i)} \in \{1, 0\}}$-ը՝ դրանց համապատասխան $\boldsymbol{y}$-ի արժեքը: Այդպիսի յուրաքանչյուր զույգ կոչվում է ուսուցանող նմուշ:

Մուտքային տվյալը (x փաստաթուղթը) դրական (+) և բացասական (-) դասերում դասակարգելու համար մեր կողմից օգտագործվել են, օրինակ, հետևյալ տիպի հատկություններ (ատրիբուտներ).

$f_1(c, x) = 1\ if$ "լավ" $\in$ x & $c = +$ otherwise 0,

$f_2(c, x) = 1\ if$ "վատ" $\in$ x & $c = -$ otherwise 0,

$f_3(c, x) = 1\ if$ "հիանալի" $\in$ x & $c = -$ otherwise 0:

Այս հատկություններից ամեն մեկին տրվում է կշիռ, որը կարող է լինել դրական կամ բացասական: $w_1(c, x)$-ը ցույց է տալիս <<լավ>> բառի կշիռը $c = +$ դասի համար, $w_2(c, x)$-ը և $w_3(c, x)$-ը՝ համապատասխանաբար <<վատ>> և <<հիանալի>> բառերի կշիռները $c = -$ դասի համար: $w_2(c, x)$-ը կլինի դրական, քանի որ տրամաբանորեն ժխտական բառերը ավելի շատ կհանդիպեն բացասական իմաստ ունեցող տեքստերում: Բայց, օրինակ, $w_3(c, x)$-ը կլինի բացասական, քանի որ <<հիանալի>> բառը հիմնականում հանդիպում է դրական տոնայնություն ունեցող տեքստերում: Ուսուցանող յուրաքանչյուր նմուշի համար որոշվել են դրանց հատկությունները և համապատասխան կշիռները:

Տրամաբանական ռեգրեսիայի ուսուցումը: Ուսուցման x օրինակների համար w կշիռներն ընտրվում են այնպես, որ դրանց համապատասխան y հավանականությունները լինեն մեծագույնը:

Տրված մուտքային x^j տվյալի համար w կշիռն ընտրվում է

$$\widehat{w} = arg_w \max \log P(y^j \,|\mathrm{x}^{\mathrm{j}})$$

բանաձևով [4] , իսկ բոլոր ուսուցման օրինակների համար կշիռն ընտրվում է հետևյալ բանաձևով.

$$\widehat{w} = arg_w max \sum_j logP(y^j|x^j):$$

Նպատակային $L(w) = \sum_j logP(y^j|x^j)$ ֆուկցիան անհրաժեշտ է մաքսիմալացնել:

P(c|x) հավանականությունը, կախված ատրիբուտներից, հաշվարկվում է հետևյալ բանաձևով.

$$P(c|\mathrm{x}) = \frac{\exp(\sum_{i=1}^{N} w_i f_i(c,x))}{\sum_{c' \in C} \exp(\sum_{i=1}^{N} w_i f_i(c',x))} ,$$

որտեղ $f_i(c,x)$-ը նշանակում է, որ դիտարկվում է x-ի f_i ատրիբուտը c դասի համար: Ուստի փնտրվում է այնպիսի w, որը մաքսիմալացնի նպատակային L ֆունկցիան՝ կախված ուսուցողական տեքստերում եղած ատրիբուտներից:

Այս մոդելում X օբյեկտը կարելի է դասել $y = 1$ դասին, եթե մոդելի կանխատեսած հավանականությունը $P\{y = 1 \mid x\} > 0{,}5$ և $y = 0$ դասին՝ հակառակ դեպքում:

Որոշման ծառեր (decision tree): Ծառը (որոշման ծառը) դասակարգման, ռեգրեսիայի խնդիրներում կիրառվող մեքենայական ուսուցման մեթոդներից է [7]: Ծառը փաստաթղթի համար կառուցելիս ընտրվում է թերմ (բառ, ատրիբուտ), որի միջոցով փաստաթղթերը բաժանվում են երկու չհատվող ենթաբազմության: Ամեն մի ենթաբազմությունից ընտրվում է մի նոր թերմ, և այդ գործողությունները կատարվում են այնքան ժամանակ, մինչև ստացվեն ենթաբազմություններ, որում բոլոր փաստաթղթերը պատկանում են միևնույն դասին կամ այդ դասի լրացմանը:

Դիցուք ունենք Q ուսուցողական փաստաթղթերի բազմությունը և $C = \{c_1, c_2, \dots, c_{|C|}\}$ դասերի բազմությունը: Դիցուք դասերի բազմությունը բաղկացած է m տարրերից՝ $||C|| = m$: Q ենթաբազմության ամեն մի փաստաթղթի համար հայտնի է, թե այն որ դասին է պատկանում, այսինքն՝ ուսուցողական տվյալների բոլոր փաստաթղթերի համար նպատակային F ֆունկցիայի արժեքը հայտնի է.

$$\forall d \in Q,\ F: D \times \mathrm{C} \rightarrow \{0{,}1\}:$$

Ծառի տերևներում պահվում են նպատակային ֆունկցիայի արժեքները, իսկ մնացած գագաթներում՝ անցումների պայմանները արմատից մինչև տերև տեղաշարժվելու համար: Ամեն մի փաստաթղթի դասակարգման համար անհրաժեշտ է շրջանցել ծառը արմատից մինչև տերև՝ այդպիսով գտնելով նպատակային ֆունկցիայի արժեքը, այսինքն՝ որոնվող դասը: Կան որոշման ծառերի կառուցման բազմաթիվ ալգորիթմներ [7]:

Պատահական անտառ (Random Forest): Պատահական անտառը (Random Forest) մեքենայական ուսուցման մեթոդներից է, որն օգտագործում է որոշման ծառերի անսամբլ (խմբեր): Ծառերի անսամբլի յուրաքանչյուր ծառ կառուցվում է առանձին, մուտքային ընտրանիից՝ օգտագործելով վերցնելը վերադարձով մեխանիզմը (բեգինգ) և պատահական ատրիբուտների ընտրման մեթոդը [8]:

Որոշման ծառերի հիմնական թերությունը դրանց գերուսուցման միտումն է [7]: Պատահական անտառը այդ խնդրի լուծման ուղիներից է: Պատահական անտառում յուրաքանչյուր ծառ տարբերվում է մնացածից: Պատահական անտառի գաղափարը այն է, որ յուրաքանչյուր ծառ կարող է բավականին լավ լուծել դասակարգման խնդիրը, բայց նա կարող է գերուսուցվել որոշ տվյալների կողմից: Եթե կառուցվում են շատ ծառեր, որոնք լավ են աշխատում, սովորաբար, դրանք գերուսուցվում են տարբեր աստիճաններով: Գերուսուցումը կարող ենք կրճատել դրանց արդյունքների միջինացումով:

Պատահական անտառի մոդելը կառուցելու համար որոշվում են մի շարք պարամետրեր, որոնց արժեքների փոփոխմամբ կարգավորվում է մոդելը: Պատահական անտառը լավ չի աշխատում շատ մեծ չափողականությամբ տվյալների, ինչպես նաև՝ հազվադեպ տվյալների, օրինակ, տեքստային տվյալների համար: Նշենք, որ մեր ստեղծած մոդելում հենց այդպիսի արդյունքներ էլ ստացվել են: Փորձարկումների արդյունքում համոզվել ենք, որ տեքստային տվյալների համար գերադասելի են գծային մոդելները: Պատահական անտառը լավ է աշխատում շատ մեծ տվյալների սահմաններում զուգահեռ պրոցեսորային ռեժիմում: Այնուամենայնիվ, պատահական անտառը պահանջում է ավելի շատ հիշողություն և ավելի դանդաղ է ուսուցանվում, քան գծային մոդելները:

Տրամաբանական ռեգրեսիա և պատահական անտառ մոդելների իրականացումները

Տրամաբանական ռեգրեսիա

Հայերեն տեքստը դասակարգելու նպատակով նախագծված տրամաբանական ռեգրեսիա և պատահական անտառ մոդելների իրականացման համար անհրաժեշտ ծրագրերը գրվել են python լեզվով: Մեքենայական ուսուցման համար՝ որպես ուսուցանվող տվյալներ, մեր կողմից օգտագործվել են twitter.com և facebook.com սոցիալական ցանցերի գրառումները և մեկնաբանությունները: Դրանցից կառուցել ենք <<դրական>> կամ <<բացասական>> տոնայնությամբ դասերին համապատասխան գրառումներ: Կատարել

ենք հայերեն տեքստային տվյալների (500 տեքստ twitter.com-ից, 200 տեքստ facebook.com-ից) մշակումներ (հղումների, թեգերի կետադրական նշանների հեռացում, բացատների նորմալիզացիա, մեծատառերի փոխարինում փոքրատառերով և այլն):

Մեկնաբանությունները կարդալու և facebook-corpus ֆայլը կառուցելու համար օգտագործել ենք Facebook Graph API: Տեքստային տվյալների մշակման և վեկտորային ներկայացման համար օգտագործվել են Bag Of Words և word2vec մեթոդները: Մեր կողմից 700 փաստաթուղթ բերվել է վեկտորային տեսքի և օգտագործվել մոդելների իրականացման համար: Ստեղծված մոդելների աշխատանքների արդյունքները գնահատվել են F1 չափողականությամբ:

Օգտատերերին ուսուցանված մոդելները տրամադրելու և օգտագործելու համար մշակել ենք twitter.com և facebook.com սոցիալական ցանցերից հայերեն տեքստ ներբեռման Web API-ներ: Դրանց միջոցով դասակարգման մուտքում ստանալով տեքստային տվյալ՝ վերադարձվում է նրանց դասը՝ դրական կամ բացասական: Դրանով օգտատերին հնարավորություն է տրվում հայերեն տեքստային մեկնաբանության դրական կամ բացասական լինելը իմանալ առանց տվյալ տեքստը կարդալու:

Sktlearn CountVectorizer դասի միջոցով կառուցվել է հատկությունների բառարանը: LogisticRegression–ի օգնությամբ ստեղծվել է դասակարգող օբյեկտը, որը, ըստ ուսուցանող գրառումների վեկտորական ներկայացման, ուսուցում է մոդելը: Ուսուցման ավարտից հետո պատրաստի մոդելը փորձարկվել է թեստային գրառումների վրա, բնական է, օգտագործելով դրանց վեկտորային ներկայացումը:

Թեստավորման արդյունքում (100-ից ավելի գրառում) տեքստերի ձևափոխման Bag of Words մեխանիզմը կիրառելիս ստացվել են հետևյալ արդյունքները.

Accuracy = 0.789, Precision = 0.784, Recall = 0.789, F1 = 0.785:

Ճշտությունը (accuracy) ցույց է տալիս, թե բոլոր դիտարկված օրինակներից քանի տոկոսն է ճիշտ դասակարգվել:

Երբ CountVectorizer դասը փոխարինել ենք TfidfVectorizer-ով, որն իրականացնում է մշակված TFIDF ալգորիթմը, ստացվել են ավելի թույլ արդյունքներ.

Accuracy = 0.773, Precision = 0.772, Recall = 0.773, F1 = 0.773:

Նույն մոդելում տեքստերի ձևափոխման word2vec մեխանիզմը կիրառելիս ստացվել են բավականին լավ արդյունքներ.

Accuracy = 0.812, Precision = 0.819, Recall = 0.812, F1 = 0.815:

Այսպիսով, տվյալների ներկայացման word2vec մեխանիզմի օգտագործումը լավացրել է մոդելի աշխատանքը:

Մոդելի իրականացման ժամանակ ստացվել է նաև բառերի կարևորության աստիճանի գնահատումը դրական կամ բացասական տոնայնություն ունեցող տեքստերում ըստ կշիռների: Մոդելը բավական լավ կարողացել է առանձնացնել դրական և բացասական իմաստ ունեցող բառերը: Դրական իմաստ ունեցող բառերից ամենամեծ կշիռն ունի <<գեղեցիկ>>-ը, այնուհետև <<հրաշալի>> և <<շատ>> բառերը: Իսկ բացասական իմաստ ունեցող բառերից ամենափոքր կշիռն ունեն <<վատ>>, <<չի>> բառերը: Սակայն որոշ բառերի տրվել են նաև բարձր (ցածր) կշիռներ, որոնք իրականում չունեն որևէ էմոցիանալ նկարագիր : Պատկերը փոխվում է ուսուցման տվյալների քանակն ավելացնելիս:

<u>Պատահական անտառ</u>

Ծրագրի աշխատանքի հիմնական փուլերն են.

- բեռնվում են տեքստային տվյալները և դրանց համապատասխան տոնայնությունները,
- կատարվում են տեքստային տվյալների մշակում և դրանց վեկտորային տեսքով ներկայացում, որն իրականացվել է sktlearn գրադարանի TfidfVectorizer դասի միջոցով (TFIDF ալգորիթմ),
- մշակված տեքստերը բաժանվում են ուսուցանման և թեստային տվյալների համապատասխանաբար 80 % և 20 % հարաբերությամբ:

Random Forest մոդելի իրականացման համար օգտագործել ենք sktlearn RandomForestClassifier դասը: Տվյալ ալգորիթմի պարամետրերի օպտիմալ արժեքների ընտրությունը և մոդելի աշխատանքի սխալանքի նվազեցումը կատարվել է փորձարկումների միջոցով (օրինակ, ավելացվել է օգտագործվող որոշման ծառերի քանակը՝ դարձնելով 100 և այլն): Մոդելի ուսուցումից հետո կատարել ենք փորձարկումներ թեստային տվյալների վրա:

Ներքոհիշյալ աղյուսակում ներկայացված են տրամաբանական ռեգրեսիա մոդելի իրականացման արդյունքները՝ տվյալների վեկտորացման տարբեր մեթոդների դեպքում և պատահական անտառ մոդելի իրականացման արդյունքը՝ տվյալների Bag of Words վեկտորացման դեպքում: Բերված է նաև յուրաքանչյուր մոդելի սկզբնարժեքավորման փուլի տևողությունը (մոդելի ուսուցման համար ծախսված ժամանակահատվածը):

Աղյուսակ

<<Տրամաբանական ռեգրեսիա>> և <<պատահական անտառ>> մոդելների կիրառման արդյունքները

Դասակարգիչ	Accuracy	Precision	Recall	F1 չափողակ.	Ուս. ժ. (*վայրկյան*)
Logistic regression Bag of Words	0,773	0,772	0,773	0,773	0,005948
Logistic regression word2vec	0,812	0,819	0,812	0,815	0,042347
Random Forest	0,750	0,758	0,750	0,706	1,414245

Ըստ աղյուսակի ավելի լավ արդյունքներ ստացվել են տրամաբանական ռեգրեսիայի կիրառման դեպքում, երբ օգտագործվել է տվյալների ներկայացման word2vec մեխանիզմը:

Մեր կողմից մշակված հայերեն լեզվով նոր գրառումներ ներբեռնելու և կառուցված մոդելները իրականացնելու ծրագրային մոդուլները տեղադրվել են github.com կայքում https://github.com/hripman/Armenian-corpus: Այն օգտատերերին հնարավորություն կտա խնայել սոցիալական ցանցերում ներկայումս առկա մեծածավալ տեղեկատվությունը կարդալու վրա ծախսվող ժամանակը:

Գրականություն

1. **Peter D. Turney.** Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. - Philadelphia, July 2002. - P. 417-424.
2. **Pang B., Lee L.** Opinion Mining and Sentiment Analysis // Foundations and Trends in Information Retrieval. - 2008.- Vol. 2, No. 1–2. - P. 1–135.
3. **Пазельская А.Г., Соловьев А.Н.** Метод определения эмоций в текстах на русском языке // Компьютерная лингвистика и интеллектуальные технологии - М.: Изд-во РГГ, 2011. - Вып. 10 (17). - С. 510–522.
4. **Daniel Jurafsky, James H. Martin.** Speech and Language Processing // An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Stanford University. - Draft of August 28, 2017.
5. **Leon Derczynski.** Complementarity, F-score, and NLP Evaluation. - University of Sheffield S1 4DP, UK, 2013.
6. Матрица ошибок (Confusion matrix) http://ru.learnmachinelearning.wikia.com/wiki/
7. **Акобир Ш.** Деревья решений - С4.5 Математический аппарат.Часть 1, https://basegroup.ru/community/articles/math-c45-part1
8. Материал из Википедии, Random Forest, https://ru.wikipedia.org/wiki/Random_forest 28.08.2018.

С.Г. Саркисян, А.А. Харатян, А.С. Овакимян

МЕТОДЫ АНАЛИЗА ТОНАЛЬНОСТИ АРМЯНСКОГО ТЕКСТА

Разработаны механизмы сбора, форматирования и векторного представления армянских текстов из некоторых социальных сетей, используемые для создания и реализации моделей логистической регрессии и случайного леса. Построены несуществующие до настоящего времени разделы корпуса армянского языка (twitter-corpus.scv и facebook-corpus.scv). Для пользователей разработаны программные модули с целью загрузки новых записей на армянском языке и реализации построенных моделей, которые размещены на сайте github.com https://github.com/hripman/Armenian-corpus:

Ключевые слова: *логистическая регрессия, анализ тональности текста, случайный лес, обучение с учителем, обучающее множество, матрица ошибок.*

S.G. Sargsyan, A.A. Kharatyan, A.S. Hovakimyan

SENTIMENT ANALYSIS METHODS FOR ARMENIAN TEXT

Mechanisms for the collection, formatting and vector representation of Armenian texts from some social networks are developed, used for logical regression and for the creation and implementation of random forest models. Non existing up to now corpses for texts in Armenian (twitter-corpus.scv and facebook-corpus.scv) have been built. Software modules have been created for users for the purpose of downloading new records in Armenian and to implement the built-in model on github.com (https://github.com/hripman/Armenian-corpus).

Keywords: *logistic regression, text sentiment analysis, Random Forest, supervised learning, training set, confusion matrix.*

Սարգսյան Սիրանուշ Գեղամի - ֆիզ.մաթ. գիտ. թեկնածու, դոցենտ, ԵՊՀ
Խառատյան Ալվարդ Ալբերտի - տնտես. գիտ. թեկնածու, դոցենտ, ԵՊՀ
Հովակիմյան Աննա Սեդրակի - տեխն. գիտ. թեկնածու, դոցենտ, ԵՊՀ