ხმის ამომცნობი სისტემების შედარებითი ანალიზი

ვახტანგ ველიაშვილი[1], ეკატერინე ჩიკაშუა[2]

[1] დოქტორანტი, ინფორმატიკისა და მართვის სისტემების ფაკულტეტი საქართველოს ტექნიკური უნივერსიტეტი, თბილისი, კოსტავას 77

[2] ასოცირებული პროფესორი, ინფორმატიკისა და მართვის სისტემების ფაკულტეტი საქართველოს ტექნიკური უნივერსიტეტი, თბილისი, კოსტავას 77

ანოტაცია

სტატიაში განხილულია საუბრის ამოცნობის მიდგომების დადებითი და უარყოფითი მხარეები, განხილულია საუბრის სიგნალის ტიპები, სპიკერის მოდელის ტიპები, ლექსიკონების ტიპები და საუბრის ამოცნობის პროცესი. ასევე გაკეთებულია არსებული ხმის ამოცნობის სისტემების შედარებითი ანალიზი, მათი დადებითი და უარყოფითი მხარეები. ამ სისტემების შედარება გაყოფილია ორ ნაწილად, სისტემები ღია კოდით და სისტემები დახურული კოდით.

საკვანძო სიტყვები: *საუბრის ამოცნობა, საუბრის ამოცნობის სისტემების შედარება, ამოცნობის პროცესი*

1.0 შესავალი

საუბრის ამოცნობა (SRS) არის პროცესი, რომელიც ალგორითმის მეშვეობით, გამოყოფს სიტყვებს საუბრის სიგნალისგან. ეს არის კომპიუტერული პროგრამის შესაძლებლობა, მოახდინოს საუბარში სიტყვების და ფრაზების ამოცნობა და მანქანისთვის წაკითხვად ფორმატში გადაყვანა.

საუბრის ამოცნობის იდეა 1920 წელიწადს დაიბადა. ეს ტექნოლოგია პირველად ჩაინტეგრირებული იყო რადიოში. ტექნოლოგია დროდადრო იხვეწებოდა და დღესაც განიცდის ცვლილებებს.

ხმის ამოცნობის სისტემები სწრაფი ტემპით ვითარდება, თუმცა უნდა აღინიშნოს, რომ სხვადასხვა აქცენტების, დიალექტების და ჟარგონის გარჩევა ჯერ კიდევ საკმაოდ რთულ ამოცანას წარმოადგენს ხმის ამოცნობის სისტემებისთვის. წლიდან წლამდე ეს სისტემები იხვეწება, მაგალითად Google-მა Connectionist Temporal Classification (CTC) ნეირონული დროებითი კლასიფიკაციის მეშვეობით შეძლო გაეზარდა ხმის ამოცნობის სიზუსტე და სიჩქარე მუდმივი ხმაურის და გარე ხმების პირობებში.

2.0 საუბრის სიგნალის ტიპები:

საუბრის ამოცნობის სისტემის შესაძლებლობა ამოიცნოს საუბარის სიგნალი, შეიძლება გაიყოს ორ ნაწილად.

- **გამოყოფილი სიტყვები:** ამ ტიპის სისტემა ერთჯერადად იღებს ერთ წარმოთქმას. და როგორც წესი საჭიროებს წარმოთქმის წინ და ბოლოში სიჩუმეებს, ამიტომ წარმომთქმელი უნდა გაჩერდეს ყოველი სიტყვის შემდეგ. ეს სისტემა კარგად

მუშაობს ცალკეული სიტყვების ამოცნობაზე თუმცა, რამდენიმე, გადაბმულად ნათქვამი სიტყვის შემთხვევაში ცუდ შედეგს იძლევა.

- **დაკავშირებული სიტყვები:** ამ ტიპის სისტემა მიწოდებულ რამდენიმე სიტყვის ჯაჭვად იშლება და მუშავდება ცალკეულ სიტყვებად, რომელთა შორისაც არის მცირე პაუზები.
- **უწყვეტი საუბარი:** ამ ტიპის სისტემა აღიქვავს უწყვეტ საუბარს. ესეთი სისტემები რთული შესაქმნელია რადგან ისინი დასანერგად მოითხოვენ განსაკუთრებულ მეთოდს.
- **სპონტანური საუბარი:** ამ ტიპის დროს ბუნებრივ და სპონტანურ სიტყვას აქვს შესაძლებლობა გაუმკლავდეს სხვადასხვა გარემოებებს, როგორიცაა სიტყვების ერთად წარმოთქმა, სიტყვის არასწორად წარმოთქმა, უცხო სიტყვები და არასწორი დებულებები, რომლებიც რთულია აღსაქმელად.

სპიკერის მოდელის ტიპები

ყველა სპიკერს აქვს უნიკალური მახასიათებლები, რომელიც გავლენას ახდენს ხმაზე. ამ მახასიათებლების საფუძველზე სისტემები იყობა ორ ძირითად კლასად.

1. **სპიკერზე დამოკიდებული მოდელი:** ეს მოდელი დამოკიდებულია კონკრეტულ სპიკერზე. ეს მოდელები მარტივია დასაინტეგრირებლად და ნაკლებად ძვირადღირებულია. სპიკერიდან გამომდინარე ის იძლევა განსხვავებულ შედეგებს.
2. **სპიკერისგან დამოუკიდებელი მოდელი:** ეს მოდელი დამოკიდებულია ბევრ სხვადასხვა სპიკერზე. ეს მოდელები უფრო რთული დასაინტეგრირებელია და ბევრად ძვირადღირებული. ბევრი სპიკერის შემთხვევაში იძლევა კარგ შედეგს და კონკრეტული სპიკერის შემთხვევაში ნაკლებად კარგ შედეგს.

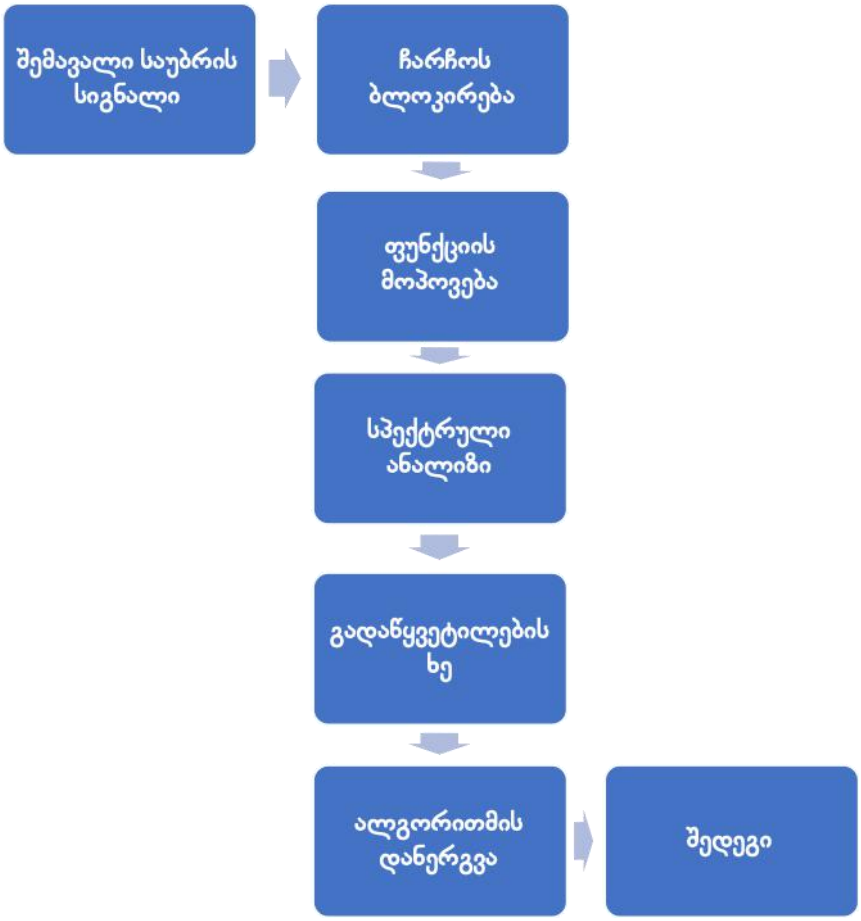
ლექსიკონების ტიპები

სისტემის სიზუსტის, სირთულის და დამუშავების მოთხოვნები დამოკიდებულია საუბრის ამომცნობის ლექსიკონის ზომაზე. ზოგი აპლიკაცია მოითხოვს რამდენიმე სიტყვას, სხვები კი დიდი ზომის ლექსიკონს. ლექტიკონის ტიპების კლასიფიკაცია შეიძლება შემდეგნაირად მოხდეს.

1. პატარა ზომის ლექსიკონი, რომელიც რამდენიმე ათეული სიტყვისგან შედგება.
2. საშუალო ზომის ლექსიკონი, რომელიც რამდენიმე ასეული სიტყვისგან შედგება.
3. დიდი ზომის ლექსიკონები შეიცავს ათასობით სიტყვას.
4. ძალიან დიდი ზომის ლექსიკონები შეიცავს ათი ათასობით სიტყვას.

3.0 საუბრის ამოცნობის პროცესი

საუბრის ამოცნობის სისტემაში, უცნობი სიტყვის სიგნალი გარდაიქმნება მიმდევრობითი ფუნქციების ვექტორებად - სხვადასხვა საუბრის დამუშავების მეთოდებით. ის ალგორითმის გამოყენებით აკონვერტებს ფუნქციის ფონემს ბადედ. ამოცნობის მოდული ლექსიკონის მიხედვით სიტყვების ბადეში ტრანსფორმაციას უკეთებს ფონემის ბადეს. შემდეგ ხდება გრამატიკის გამოყენება სიტყვების ბადეზე, რათა მოხდეს კონკრეტული სიტყვების და ტექსტის ამოცნობა. ის მიმართავს სიტყვის ბადეს კონკრეტული სიტყვებისა თუ ტექსტის შესასწავლად. საუბრის ამოცნობის პროცესი დაყოფილია რამდენიმე ეტაპად. (სურ 1.)



სურ.1 საუბრის ამოცნობის პროცესი

ნაბიჯი 1: ამ ეტაპზე საუბრის სიგნალი იყოფა ერთნაირად დაშორებულ ბლოკებად, რათა მიიღოს მახასიათებლები, როგორიცაა ჯამური ენერგია, ნულოვანი გადაკვეთის სიძლიერე სხვადასხვა სიხშირის მონაკვეთებზე და სხვა. ამ მახასიათებლების გამოყენებით ფუნქციის ვექტორები უთავსებენ თითოეულ ბლოკს ფონემს იმისთვის, რომ მიიღონ ფონემების სტრიქონი.

ნაბიჯი 2: ამ საფეხურზე ხდება სპექტრული ანალიზის გამოყენება ყოველ ბლოკზე წრფივი პროგნოზირებადი კოდინგის ტექნიკის გამოყენებით, სწრაფი ფურიეს ტრანსფორმაცია (FFT) და სიხშირული ფილტრების ბანკი.

ნაბიჯი 3: ამ საფეხურზე გადაწყეტილების მიღების პროცესი სრულდება ყოველ ბლოკზე. ყოველ ფონემას აქვს განსხვავებული თვისებები, რომელიც ამცირებს ამ სფეროს.

ნაბიჯი 4: ეს საფეხური გამოიყენება გადაწყვეტილების პროცესის პერფორმანსის გასაძლიერებლად, მაღალი წარმატების მაჩვენებლის მისაღებად სხვადასხვა ალგორითმების გამოყენებით. ყოველი სიტყვისთვის ლექსიკონიდან დგება ალგორითმი და შემდეგ ფონემების სტრიქონი დარდება ყოველ არგორითმს.

4.0 საუბრის ამოცნობის მიდგომების დადებითი და უარყოფითი მხარეები

ნომ.	SRS მიდგომები	უპირატესობები	უარყოფითი მხარეები

1	აკუსტიკური ფონეტიკური ამოცნობა	1. ამცირებს დამუშავების დროს დაკავშირებული სიტყვებისთვის	1. არ არის ფართოდ გამოყენებული კომერციულ პროექტებში, იმის გამო, რომ დიდი დრო სჭირდება თითოეული სიტყვის დამუშავებას
2	ნიმუშის ამოცნობის მიდგომა	1. ნიმუშის ამოცნობა შეუძლია სწრაფად, მარტივად და ავტომატურად, რადგან ხდება სიტყვების დამთხვევა.	1. ხელსაყრელია სიტყვის სიტყვაზე დამთხვევის დროს 2. შაბლონის სანახი არის მთავარი პრობლემა 3. ნელი დამუშავება 4. ვერ ახდენს საუბრის ამოცნობას თუ მოხდა ახალი ნიმუშის ვარიაცია
3	შაბლონზე დაფუძნებული მიდგომა	1. უკეთ მუშაობს დისკრეტული სიტყვებისთვის 2. ნაკლები შეცდომა ხდება მცირე ცვლადების სეგმენტაციის და კლასიფიკაციის გამო	1. ძვირია, რადგან დიდი ლექსიკონი აქვს, ყველა სიტყვას აქვს თავისი საცნობარო ნიმუში 2. ნიმუშების დამთხვევა და მომზადება მოითხოვს მეტ დროს 3. რთულია ერთნაირი ნიმუშების ამოცნობა 4. მალე დეგრადირდება როცა შემავალი მონაცემი განსხვავდება ნიმუშისგან
4	დროის დინამიური გაწელვა	1. უწყვეტობა ნაკლებად მნიშვნელოვანია რადგან შესაძლებელია თანამიმდევრობის დამთხვევა გამოტოვებულ ინფორმაციაზე 2. საიმედო დროის დარეგულირება ცნობარს და სატესტო ნიმუშს შორის	1. ის ემთხვევა ორ მოცემულ მიმდევრობას შორის გარკვეული შეზღუდვებით 2. ის მოითხოვს მაქსიმალურ დროს რთული გამოთვლებისთვის 3. ნიმუშების შეზღუდული რაოდენობა აქვს

			4. ის ითხოვს უცნობ საუბრის სიგნალს ტრენინგისთვის
5	ვექტორული კვანტირების მიდგომა	1. გამოსადეგია მონაცემების ეფექტურად შემცირებისთვის	1. ის არის ტექსტზე დამოკიდებული, რადგან სჭირდება კოდის წიგნაკი დამთხვევისთვის
6	სტატისტიკაზე დაფუძნებული მიდგომა (მარკოვის ფარული მათოდი)	1. ლექსიკონის ზომა არის ძალიან დიდი და მას შეუძლია დაატრენინგოს დიდი ზომის მონაცემები 2. მას აქვს ზუსტი მათემატიკური სტრუქტურა 3. დატრენინგებული ალგორითმები ადვილად ხელმისაწვდომია 4. ადვილი დასაინტეგრირებელია და ყველას შეუძლია მარტივად შეცვალოს ზომა, ტიპი და არქიტექტურა ამ მოდულების რათა მოერგოს კონკრეტულს სიტყვებს 5. ბევრად უფრო მდგრადია, გამომდინარე ალბათობიდან რომ ზოგი სიტყვა ერთმანეთის გვერდით აღმოჩნდება 6. შესაძლებლობა აქვს მიაღწიოს მაღალი ამოცნობის კოეფიციენტს 7. მას აქვს ეფექტური სწავლების ალგორითმი 8. მას გააჩნია მოქნილი და გენერალური მოდელი რიგითობის თვისებებისთვის. 9. მას შეუძლია ისწავლოს არაკონტროლირებადი მონაცემებისგან	1. საგრძნობი ზრდა კომპიუტერული გამოთვლის სირთულეში 2. მოითხოვს დიდი რაოდენობით მონაცემებს
7	ხელოვნური ნეირონული ქსელის მიდგომა	1. მას შეუძია ამოხსნას რთული კომპიუტერული ამოცანები ეფექტურად, ნაკლებ დროში 2. მას აქვს შესაძლებლობა ავტომატურად დაატრენინგოს მონაცემები და ასწავლოს სისტემას ცვლილებები საწყისი	1. ის იძლევა არაეფექტურ შედეგს დიდი ლექსიკონის შემთხვევაში 2. ძვირია, რადგან ტრენინგისთვის ის საჭიროებს ბევრ

		სატრენინგო მოდელიდან შეცდომების გარეშე 3. შეუძლია გაუმკლავდეს ხმაურიან, დაბალი ხარისხის მონაცემებს ეფექტურად და მოითხოვს მინიმალურ სატრენინგო ლექსიკონის მონაცემებს	იტერაციას დიდ რაოდენობა სატრენინგო მონაცემებზე 3. რეალური ბუნება ნეირონული ქსელის ჯერ კიდევ არ არის ბოლომდე გაანალიზებული 4. მეტ სატრენინგო მონაცემებს მოითხოვს 5. მეტი შეცდომა ხდება რთული ნეირონული არქიტექტურის გამო

5.0 ხმის ამომცნობი სისტემების შედარებითი ანალიზი

ხმის ამოცნობის სისტემის შემუშავება საკმაოდ დიდ რესურსებს და დროს მოითხოვს, შესაბამისად ჩვენი პროექტისთვის არსებული სისტემის მორგება გადავწყვიტეთ.

ხმის ამოცნობის სისტემები შეიძლება გავყოთ ორ კატეგორიად: ღია კოდით და დახურული კოდით. ღია კოდის მქონე საიტები ძირითადად უფასოა და მასში ჩასწორებების შეტანა სხვებსაც შეუძლიათ. დახურული კოდის მქონე სისტემები ძირითადად ფასიანია და გამოსაყენებლად საჭიროა ლიცენზიის შეძენა.

განვიხილოთ დახურული კოდის მქონე ხმის ამოცნობის სისტემები და ინტეგრაციის გზები. ასეთ სისტემებში მსოფლიო ლიდერია Nuance Communications. ხელსაწყო Dragon Mobile SDK შედგება კლიენტის და სერვერის კომპონენტებისგან, და ასევე შეიცავს კოდის, ფრეიმვორკის, დოკუმენტაციის და შაბლონების სხვადასხვა მაგალითებს, რომლებიც საგრძნობლად ამარტივებს სერვისების და აპლიკაციების ინტეგრაციას. პლატფორმა Speech Kit შექმნილია ხმის ამოცნობის სერვისების პროექტებში და აპლიკაციებში მარტივი და სწრაფი დამატებისთვის. ასევე ამ პლატფორმის დახმარებით შესაძლებელია სერვერულ კომპონენტებზე წვდომის მოპოვება.

ხმის ამოცნობის ოპერაციების უმრავლესობაზე პასუხს აგებს სერვერების სისტემა. სერვერზე მთლიანად სრულდება ხმის ამოცნობა ან სინთეზი.

პლატფორმა Speech Kit - შინაარსობრივად არის ქსელური სერვისი, რომელიც ემორჩილება საბაზისო კონფიგურაციას, ხმის ამოცნობის ან სინთეზის კლასების გამოსაყენებლად. ამისთვის გამოიყენება ძირითადი ოპერაციები:

- აპლიკაციის განსაზღვრა და ავტორიზაცია.
- კავშირის დამყარება ხმის ამოცნობის სერვერთან.

ეს ყველაფერი უზრუნველყოფს სწრაფი მოთხოვნების შექმნას ხმის მონაცემების დასამუშავებლად და სამუშაოს ხარისხის გასაზრდელად.

Dragon Mobile SDK მაღალ სიზუსტეს უზრუნველყოფს ინგლისურ ენაზე (99%-მდე). მისი ნაკლია შეზღუდული უფასო ფუნქციონალი.

Google Speech Recognition API – კომპანია Google-ის პროდუქტია, რომელიც ხმით ძებნის საშუალებას იძლევა, ხმის ამოცნობის ტექნოლოგიის გამოყენებით. ეს ტექნოლოგია ინტეგრირებულია სმარტფონებში და კომპიუტერებში, სადაც შესაძლებელია ხმოვანი ინფორმაციის შეყვანა. ასევე ეს ტექნოლოგია დაინტეგრირებულია Google Chrome ბრაუზერში.

2014 წლის მაისიდან API-ზე წვდომა გახდა ლეგალური. ხმის ამოცნობის სისტემის მონაცემთა ბაზასთან სამუშაოდ საჭიროა Google Developers-ზე დარეგისტრირება. ასევე არის შესაძლებლობა, რომ Google-ის მიკროფონის ღილაკი დაემატოს ნებისმიერ საიტზე.

Google-ის ტექნოლოგიის გამოსაყენებლად საჭიროა POST მეთოდის გამოყენება ხმის ფაილის მისამართზე რომელიც იქნება .flac ან .spx ფორმატის. შემდგომ უნდა მოხდეს WAVE ფაილების ამოცნობა ნებისმიერი პროგრამული ენის გამოყენებით.

Google-ის ხმის ამოცნობის სისტემა ძალიან გავს Dragon Mobile SDK-ს და თანაც არ აქვს შეზღუდვა მიმართვებზე დღის განმავლობაში.

Yandex Speech Kit-ს დეველოპერები ირწმუნებიან, რომ მათი SDK არის საუკეთესო რუსული ენის ამოცნობის კუთხით. მიმართვების რაოდენობა დღეში შეზღუდულია 10000-ზე. მათი თქმით გლობალურ ბაზარზე გასვლა საკმაოდ უჭირთ, რადგან პატენტების უმრავლესობა ეკუთვნის Nuance-ს.

Yandex Speech Kit- ის ხმის ამოცნობის ეფექტურობა დამოკიდებულია პირველ რიგში ორიგინალი ხმის ხარისხზე, კოდირებაზე, ტექსტის მკაფიოობაზე, ტექსტის ტემპზე, ტექსტის სირთულეზე და სიგრძეზე. ხმის ამოცნობა ხდება რეალურ დროში აუდიო ინფორმაციის გადაცემასთან ერთად. შეფერხება არ აღემატება ერთ წამს. ასეთი მაღალი სიჩქარის უზრუნველყოფისთვის ტექნოლოგია მუშაობს ნაკადური ამოცნობის რეჟიმში შუალედური შედეგებით. ეს ნიშნავს, რომ როგორც კი ადამიანი იწყებს ლაპარაკს, მისი ნასაუბრები გადაეცემა სერვისს მცირე ნაწილებად.

Microsoft Speech API – Microsoft-ის ხმის ამოცნობის ტექნოლოგიაა.

ბოლო დროს, კორპორაციამ აქტიურად დაიწყო მსგავსი ტექნოლოგიების განვითარება.

(Cortana-ს დაანონსება, სკაიპში სინქრონული თარგმანის ტექნოლოგიის შემუშავება.)

API- ის გამოყენების სხვადასხვა ვარიანტები არსებობს:

Windows და Windows Server - საუბრის ტექნოლოგიის დამატება

Windows-ის აპლიკაციისთვის მართვადი ან საწყისი კოდის გამოყენებით, რომელიც API-დან მოდის.

Speech Platforms - პლატფორმის ჩართვა სხვადასხვა პროგრამებში,

Microsoft- ის დისტრიბუტივის გამოყენებით;

Embedded - გადაწყვეტილებები, რომლებიც საშუალებას აძლევს ადამიანებს ურთიერთქმედება მოახდინოს მოწყობილობებთან ხმის გამოყენებით;

Services - აპლიკაციის შემუშავება ხმოვანი ფუნქციებით რეალურ დროში გამოსაყენებლად.

ზევით განვიხილეთ ყველაზე გავრცელებული ხმის ამოცნობის სისტემები. საჭირო იყო აგვერჩია, რომელი სისტემა იქნებოდა უფრო გამართლებული ჩენს პროექტში გამოსაყენებლად. Dragon Mobile SDK საკმაოდ მოხიბვლელია და კარგი დოკუმენტაცია აქვს, თუმცა რთული ლიცენზირების სისტემა აქვს და გამოყენების მკაცრი წესები. აქედან გამომდინარე უფრო მიზანშეწონილი იქნებოდა Google

Speech Recognition API-ს გამოყენება, რომელსაც კომპანიის დიდი გამომთვლელი სიმძლავრეების გამო მაღალი სიჩქარით და მრავალმხრივი ინტეგრირებადობით

გამოირჩევა. ასევე დადებითი მხარეა, რომ მას არ აქვს დღიური შეზღუდვა მიმართვებზე. Google საკმაოდ აქტიურად ანვითარებს ხმის ამოცნობის ტექნოლოგიას, რაც ასევე დადებითად მოქმედებს გადაწყვეტილების მიღებაზე.

ახლა განვიხილოთ ხმის ამოცნობის სისტემები ღია კოდით. CMU Sphinx იყო შექმნილი Carnegie Mellon-ის უნივერსიტეტის პროგრამისტების ჯგუფის მიერ. სისტემა შედგება ხმის ამოცნობის (Sphinx 2–4) და აკუსტიკური მოდელი (Sphinx train) კომპლაქტისგან.

Sphinx წარმოადგენს უწყვეტი ტექსტის ამომცნობს, რომელიც იყენებს მარკოვის ფარულ მოდელებს და სტატისტიკური ენის მოდელებს. სისტემაში რეალიზებულია ხანგრძლივი ტექსტის ამოცნობის შესაძლებლობა დიდი ლექსიკური ამოცნობისთვის.

Sphinx 4 - სრულიად განახლებული Sphinx-ის ტექსტური ძრავაა, რომელიც არის ფუძე ხმის ამოცნობის ტექნოლოგიის გამოკვლევისთივს. Sphinx 4 შექმნილი იყო Java პროგრამულ ენაზე.

Julius - არის მაღალი წარმადობის ხმის ამოცნობის სისტემა დიდი ლექსიკონით, ასევე პროგრამული უზრუნველყოფის დეკოდერით, დაკავშირებული ტექსტის გამოსაკვლევად. დეკოდერი მუშაობს უმრავლესობა თანამედროვე კომპიუტერებზე, ლექსიკონის მოცულობა არის 60 ათასი სიტყვა. სისტემის მთავარი თავისებურება ის არის, რომ დანერგვის მხრივ არის ძალიან მოქნილი. პროექტის ძირითადი პლატფორმა არის Linux-ის და სხვა UNIX-ის სისტემები, თუმცა არსებობს Windows-ის ვერსიაც.

თავდაპირველად Julius შექმნეს იმისთვის რომ ამოეცნო იაპონური საუბარი, რაც მის მინუსად შეიძლება ჩაითვალოს. აღსანიშნავია, რომ არსებობს პროექტი VoxForge, რომელიც ქმნის აკუსტიკურ მოდელს ინგლისური ენისთვის Julius-ის ბაზაზე.

RWTH ASR (RASR) - არის საუბრის ამოცნობის სისტემა, რომელიც მოიცავს ტექნოლოგიას ავტომატური საუბრის ამოცნობის შესაქმნელად. RWTH ASR შედგება ხელსაწყოებისგან აკუსტიკური მოდელების და დეკოდერების შესაქმნელად, ასევე კომპონენტებისგან საუბრის ადაპტაციისთვის და ადაპტირებადი სისტემების სწავლებისგან. აღნიშნული საუბრის ამოცნობის სისტემა მუშაობს Linux და MacOS ოპერაციულ სისტემებზე. პროექტს აქვს დეტალური დოკუმენტაცია და სთავაზობს მზა შაბლოებს და მოდელებს სისტემების სწავლებისთვის.

Simon - ის საუბრის ამოცნობის სისტემა, რომელიც შექნილია Julius და HTK ძრავების დახმარებით. Simon საკმაოდ მოსახერხებელია სხვადასხვა ენებთან და დიალექტებთან სამუშაოდ. სისტემის მუშაობა დაფუძნებულია განსაზღვრული სცენარების შესრულებაზე. მომხმარებლებს თავად შეუძლიათ შექმნან სცენარები და გამოაქვეყნონ საზოგადოებისთვის. Simon მუშაობს ბევრ GPL-ის მსგავს მოდელთან. მომხმრებლები იყენებენ ამ მოდელებს წარმოთქმისთვის. თვითონ სისტემას შეუძლია წინასწარი ტრენინგის გარეშე მუშაობა.

ღია კოდის მქონე საუბრის ამოცნობის სისტემების გავრცელების განხილვის შემდეგ შეგვიძლია დავასკვნათ, რომ შედარებით უფრო საინტერესოა CMU Sphinx სისტემა. თუმცა, როგორც ადრე ვთქვით, ღია კოდის მქონე საუბრის ამოცნობის სისტემების კარგი მუშაობისთვის აუცილებელია პირველადი მონაცემების დიდი ბაზა, სხვა შემთხვევაში ამოცნობის სიზუსტე არ იქნება კარგი. მთლიანობაში ასეთი სისტემების გამოყენება საკუთარი სისტემისთვის სავსებით გამართლებულია.

ლიტერატურა

[1] Stefan Windmann and Reinhold Haeb-Umbach,, Approaches to Iterative Speech Feature Enhancement and Recognition, IEEE Transactions On Audio, Speech, And Language Processing, Vol.
[2] Sadaoki Furui, 50 years of Progress in speech and Speaker Recognition Research , ECTI Transactions on Computer and Information Technology
[3] Speech Input API Specification – [Электронный ресурс]. // W3C. – Режим доступа. – URL: https://www.w3.org/2005/Incubator/htmlspeech/2010/10/google-api-draft.html
[4] Голосовой поиск в Google Chrome – [Электронный ресурс] // Habrahabr. – Режим доступа. – URL: http://habrahabr.ru/post/111201
[5] Nuance Developers – // Nuance Communications, Inc. – Режим доступа. – URL: https://developer.nuance.com/public/index.php?task=home
[6] B.H.Juang and S.Furui, Automatic speech recognition and understanding: A first step toward natural human machine communication
[7] W. Abdulla, D. Chow, and G. Sin, Cross-words reference template for DTW-based speech recognition systems, in Proc. IEEE TENCON, Bangalore, India, 2003.
[8] M.Weintraub et.al, linguistic constraints in hiddenmarkov Model based speech recognition.
Spector, Simon Kinga and Joe Frankel, Recognition, Speech production knowledge in automatic speech recognition, Journal of Acoustic Society of America.

სტატია მიღებულია: 2019-06-24