DOI: https://doi.org/10.33216/2222-3428-2019-17-4

UDC: 004.855.5

Derevianchenko V.S., Biloborodova T.O., Skarga-Bandurova I.S., Fursa P.S., Koverha M.O.

A SURVEY OF OPEN-SOURCE SPEECH RECOGNITION SOFTWARE FOR VOICE ACTUATED CONTROL

Nowadays, speech recognition technologies are actively developing and find application in various fields, such as controlling a computer using voice, dictating texts, and human-computer interaction or communicating with a computer on an intellectual level. Non-contact and natural for human ways of interacting with a computer, based on automatic recognition and synthesis of speech, text, gesture and tactile information, as well as paralinguistic information, including non-verbal aspects of speech and text information, are especially relevant. Much attention is paid to creating an accessible environment for people with disabilities and disabilities. An important means of ensuring accessibility and improving the quality of life, social interaction, and integration into society for people with disabilities are computer facilities and specialized information systems. In this paper we presented a series of experiments and analyzed the open-source software for voice actuated control. Simon is a software for Internet surfing, mailing, managing multimedia applications that can be adapted to the needs of older people. To work with the acoustic model, we need to either have our own model or load it from the library. The speech recognition was evaluated using two criteria, they are WER and Latency. The analysis of the software for recognition of continuous speech showed that at present there is no universal system for recognizing continuous speech that would be capable of self-learning, would be speaker-independent, resistant to noise, and have a low error rate. The considered software solutions at the moment are not universal and accurate, the speech recognition error greatly depends on the presence of extraneous medium and high-frequency noise, as well as on the microphone quality.

***Keywords**: voice, speech recognition, software, human-computer interaction*

Introduction. According to a PwC report [1], only 10% of the 1000 respondents surveyed were not familiar with voice-enabled products and devices, and most used a voice assistant. According to forecasts, already in 2020, about 50% of search queries will be performed by voice, and by 2022 55% of American families will have smart columns. At present, it is planned to introduce chatbots capable of recognizing human speech and searching for information by voice commands. Telephony will be robotic based on chat bots: companies will be able to automatically receive and make calls when interacting with customers or partners, while the robot will be able to conduct complex dialogs, moving from topic to topic. In general, speech recognition comes down to three types:

- Recognition of separately pronounced words, so-called commands - used for verbal control of various objects, applications, site navigation, etc;
- Recognition of continuous speech in a large dictionary - aims at converting a person's natural speech into a text (automatic decoding of records, creation of transcripts);
- Speech-based identification — used for security purposes (protecting an object from unauthorized access, using the individual characteristics of each person's voice).

The task of automatic speech recognition in real conditions is relevant, given the variability of the source of the speech signal and acoustic noise, which hides the original sequence of audio segments. In recent years, significant progress has been made in this area, and there are commercial voice-independent applications that quite successfully recognize speech when processing voice commands (Google maps, Yandex maps), in interactive systems (Siri), in shorthand systems. The recognition accuracy of speech units in these systems has reached the required threshold. For example, the recognition accuracy of Google's updated voice assistant is comparable to human. To process the request and give a relevant response, the system takes no more than a second. However, to use the recognition system from Google, you must purchase a license, and it is quite expensive.

In this work we provide an analysis of open source speech recognition software for recognition of continuous speech.

Experiment design.

There are 2 types of licenses in most common speech recognition systems:

- BSD (Berkeley Software Distribution) license [2] includes audio speech recognition products: CMU Sphinx, PocketSphinx, Julius, etc.
- GPL (General Public License) [3] includes: Simon software, iATROS, RWTH ASR, etc.

Simon [4] is a software for Internet surfing, mailing, managing multimedia applications that can be adapted to the needs of older people. To manage the scripts, we need only a few terms and numbers from 0 to 9. Unlike existing commercial offerings, Simon offers a unique way of self-recognition of speech. Instead of predefined, pre-trained speech models, Simon does not come with any model at all. Instead, it provides an easy-to-use end-user interface for creating language and acoustic models from scratch. To work with the acoustic model, we need to either have our own model or load it from the library (export the active model). At this point, we were just looking for any speech patterns

for Simon. For testing, the acoustic model from [5] was used.

Experiment#1. Using Simon software to control mouse and Internet surfing using standard libraries and new phrases. To do this we worked with two scenarios:

(1) controlling mouse to click on the youtube icon via voice;

(2) controlling browser and testing the following commands: "page up", "page down", and searching keywords on the page by the command "48".

In a result, almost all voice commands were performed successfully except "48". Adaptation to the user's voice for key phrases/words of this script is successful - it does not return any errors Sometimes when recording words, it reacts to the voice loudness and the intelligibility of pronunciation. We had to repeat some words for several times. In case the word was said louder than usual, the program did not work and asked to overwrite. But this happened rarely, and mostly it successfully fixes the phrases/words.

As alternative speech recognition software, Deepspeech [6] and PocketSphinx [7] were reviewed and tested. DeepSpeech is an open source software product for converting speech to text. For training, a model trained in machine learning methods is used, based on Baidu's in-depth speech research. PocketSphinx also needs language model files, acoustic model files, and a pronunciation dictionary (phonetic dictionary). These applications were tested with data from Kaggle [8] and with their own recorded files.

Experiment #2. Recognition of continuous speech on existing sets. Table 1 presents information about the types of texts used in experiments.

Table 1

The overview and some examples of input data

ID	Country/ accent	Source text	Bitrate, Kbps	Sampling frequency, HZ
1	Newzeland	the boy tried to read what was written in the sand	1536	48000
2	Ireland	but the boy knew that he was referring to fatima	1536	48000
3	Scotland	why the shots stopped after the tenth no one on earth has tried to explain	1536	48000
4	Australia	the teacher thought that he'd taught himself all he could	1536	48000
5	Canada	the shop folks were taking down their shutters and people were opening their bedroom windows	1536	48000
6	USA	but everything had changed	1536	48000
7	England	he moved about invisible but everyone could hear him	1536	48000

The speech recognition was evaluated using two criteria, they are WER and Latency.

1) Word Error Rate (WER) - A word error rate measures word level mismatch: it compares the words issued by the recognizer with those that the user actually uttered. Each error (replacement, insertion or deletion) is counted in the recognizer. WER can be calculated as follows:

$$WER = \frac{Number\ of\ Substitution + Insertions + Deletions}{Total\ number\ of\ words} \cdot 100\%$$

2) Delay (en. Latency)

Delay is defined as the total time (in seconds) required to perform speech recognition. More precisely, we set the delay as the time from the moment when the recording ends, until the recognition results appear on the screen.

WER for data from Table 1 are presented in Table 2 and in Fig. 1.

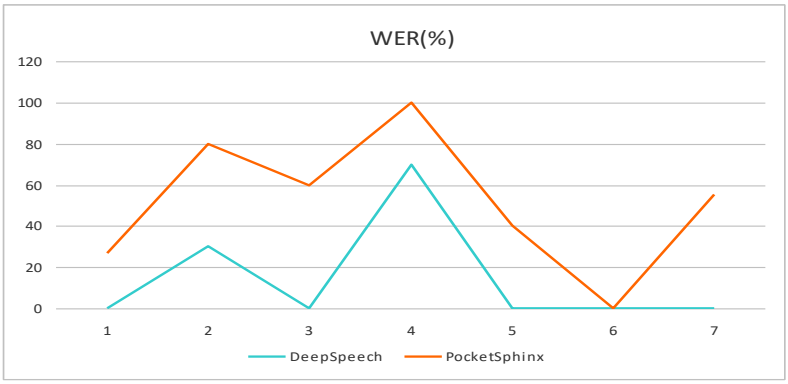

Fig. 1. WER for DeepSpeech (blue line) and PocketSphinx (orange line)

Table 2

WER for DeepSpeech and PocketSphinx

ID	Software	WER (%)	WER_{CP} (%)
1	DeepSpeech	0	14,28
2	DeepSpeech	30	
3	DeepSpeech	0	
4	DeepSpeech	70	
5	DeepSpeech	0	
6	DeepSpeech	0	
7	DeepSpeech	0	
1	PocketSphinx	27	51,71
2	PocketSphinx	80	
3	PocketSphinx	60	
4	PocketSphinx	100	
5	PocketSphinx	40	
6	PocketSphinx	0	
7	PocketSphinx	55	

The results of Latency calculation for data from Table 1 are presented in Table 3 and in Fig. 2.

Table 3

Latency for DeepSpeech and PocketSphinx

ID	Software	Latency, seconds	Average Latency, seconds
1	DeepSpeech	1.364	1,85
2	DeepSpeech	1.318	
3	DeepSpeech	3.051	
4	DeepSpeech	2.074	
5	DeepSpeech	2.360	
6	DeepSpeech	0.891	
7	DeepSpeech	1.894	
1	PocketSphinx	1.202	2,115
2	PocketSphinx	1.747	
3	PocketSphinx	3.069	
4	PocketSphinx	2.534	
5	PocketSphinx	2.320	
6	PocketSphinx	1.650	
7	PocketSphinx	2.285	

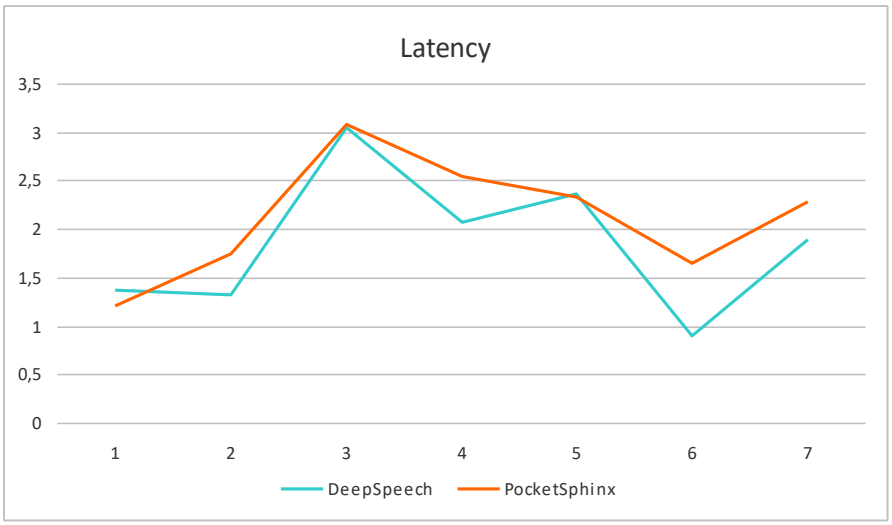

Fig. 2. Latency for DeepSpeech (blue line) and PocketSphinx (orange line)

Experiment #3. Recognition of continuous speech on new sets. Source data were recorded and tested using customized voice data [9]. The following two phrases were used:

1) Number ten scalpel for the initial incision.
2) Get the ring forceps, counterclockwise, 12, 9, 6, 3.

As it can be seen from the Table 3, the recognized results are far from the previous experiment and are not so optimistic yet.

Table 3

The results of audio recognition

ID	File name	Recognized text
1	fursa1.wav	nomebut tense cal po funisionaancesion
2	fursa2.wav	thate gerin force seps contuco quiss dwelve nine six wree
1	khotkin1.wav	nomber tents goupl fr anishian in cetiong
2	khotkin2.wav	he consering for sucs counter corquises to welfe nin six suy
1	koverha1.wav	nome bttense calpo for ine shoins sion
2	koverha2.wav	get a inhorseb sconte clock wice well min se ree
1	pokryshka1.wav	nomber tands cop of for in i shalin cesion
2	pokryshka2.wav	gevs ar ink for subscounted glock wice twellph line seex three

Conclusions. The analysis of the software for recognition of continuous speech showed that at present there is no universal system for recognizing continuous speech that would be capable of self-learning, would be speaker-independent, resistant to noise, and have a low error rate. The considered software solutions at the moment are not universal and accurate, the speech recognition error greatly depends on the presence of extraneous medium and high-frequency noise, as well as on the microphone quality.

In this regard, it should be noted that the tasks of the development of the PCP, the development and implementation of software and information solutions in this area are relevant and relevant.

References

1. UK Annual Report 2018 Leading in changing times: Working together Available at: https://www.pwc.co.uk/who-we-are/annual-report.html [Accessed 2 May. 2019].
2. "Original BSD license". Various Licenses and Comments about Them. Free Software Foundation. Available at: https://www.gnu.org/licenses/license-list.html#OriginalBSD
3. Montague B. Comparing the BSD and GPL Licenses on Technology Innovation Management Review by Available at: https://timreview.ca/article/67 [Accessed 2 May. 2019].
4. Simon https://simon.kde.org/ [Accessed 2 May. 2019].
5. Available at: http://www.speech.cs.cmu.edu/sphinx/models. [Accessed 2 May. 2019].
6. Deepspeech https://github.com/mozilla/DeepSpeech [Accessed 2 May. 2019].
7. PocketSphinx https://github.com/cmusphinx/pocketsphinx/blob/master/include/pocketsphinx.h [Accessed 2 May. 2019].
8. Common-voice Data. Kaggle Available at: https://www.kaggle.com/mozillaorg/common-voice [Accessed 2 May. 2019].
9. Customized data, Google drive. Available at: https://drive.google.com/open?id=11pNpJOK-Pr1AAb22jOW0FIhYJ9p1ekAE [Accessed 2 May. 2019].

Література

1. UK Annual Report 2018 Leading in changing times: Working together Available at: https://www.pwc.co.uk/who-we-are/annual-report.html [Accessed 2 May. 2019].
2. "Original BSD license". Various Licenses and Comments about Them. Free Software Foundation. Available at: https://www.gnu.org/licenses/license-list.html#OriginalBSD
3. Montague B. Comparing the BSD and GPL Licenses on Technology Innovation Management Review by Available at: https://timreview.ca/article/67 [Accessed 2 May. 2019].
4. Simon https://simon.kde.org/ [Accessed 2 May. 2019].
5. Available at: http://www.speech.cs.cmu.edu/sphinx/models. [Accessed 2 May. 2019].
6. Deepspeech https://github.com/mozilla/DeepSpeech [Accessed 2 May. 2019].
7. PocketSphinx https://github.com/cmusphinx/pocketsphinx/blob/master/include/pocketsphinx.h [Accessed 2 May. 2019].
8. Common-voice Data. Kaggle Available at: https://www.kaggle.com/mozillaorg/common-voice [Accessed 2 May. 2019].
9. Customized data, Google drive. Available at: https://drive.google.com/open?id=11pNpJOK-Pr1AAb22jOW0FIhYJ9p1ekAE [Accessed 2 May. 2019].

Технології розпізнавання мови активно розвиваються і знаходять застосування в різних областях, таких як управління комп'ютером за допомогою голосу, диктування текстів та реалізація взаємодії людина-комп'ютер або спілкування з комп'ютером на інтелектуальному рівні. Природні для людини способи взаємодії з комп'ютером, засновані на автоматичному розпізнаванні та синтезі мови, тексту, жестів та тактильної інформації, а також паралінгвістичної інформації, включаючи невербальні аспекти мовної та текстової

інформації, є особливо актуальними. Велика увага приділяється створенню доступного середовища для людей з обмеженими можливостями. Важливим засобом забезпечення доступності та покращення якості життя, соціальної взаємодії та інтеграції людей з обмеженими можливостями в суспільство є комп'ютерні засоби та спеціалізовані інформаційні системи. У цій роботі ми представили серію експериментів та проаналізували програмне забезпечення з відкритим кодом для управління голосом. Simon – це програмне забезпечення для інтернет-серфінгу, розсилки, управління мультимедійними додатками, яке можна адаптувати до потреб літніх людей. Для роботи з акустичної моделлю нам потрібно або мати власну модель, або завантажити її з бібліотеки. Розпізнавання мовлення оцінювалося за двома критеріями: WER і Latency. Аналіз програмного забезпечення для розпізнавання мовлення показав, що в даний час не існує універсальної системи розпізнавання мовлення, яка була б здатна до самонавчання, була б незалежною від мовця, стійкої до шуму і мала б низький рівень помилок. Розглянуті програмні рішення на даний момент не є універсальними і точними, помилка розпізнавання мовлення багато в чому залежить від наявності сторонніх середовищ і високочастотних шумів, а також від якості мікрофона.
Ключові слова: *голос, розпізнавання мовлення, програмне забезпечення, взаємодія людини та комп'ютера*

Технологии распознавания речи активно развиваются и находят применение в различных областях, таких как управление компьютером с помощью голоса, диктовка текстов и реализация взаимодействия человек-компьютер или общения с компьютером на интеллектуальном уровне. Естественные для человека способы взаимодействия с компьютером, основанные на автоматическом распознавании и синтезе речи, текста, жестов и тактильной информации, а также паралингвистического информации, включая невербальные аспекты языковой и текстовой информации, особенно актуальны. Большое внимание уделяется созданию доступной среды для людей с ограниченными возможностями. Важным средством обеспечения доступности и улучшение качества жизни, социального взаимодействия и интеграции людей с ограниченными возможностями в общество компьютерные средства и специализированные информационные системы. В этой работе мы представили серию экспериментов и проанализировали программное обеспечение с открытым кодом для управления голосом. Simon - программное обеспечение для интернет-серфинга, рассылки, управления мультимедийными приложениями, которое можно адаптировать к потребностям пожилых людей. Для работы с акустической моделью нам нужно либо иметь собственную модель, либо загрузить ее из библиотеки. Распознавание речи оценивалось по двум критериям: WER и Latency. Анализ программного обеспечения для распознавания непрерывной речи показал, что в настоящее время не существует универсальной системы распознавания непрерывной речи, которая была бы способна к самообучению, была бы независимой от говорящего, устойчивой к шуму и имела бы низкий уровень ошибок. Рассматриваемые программные решения на данный момент не являются универсальными и точными, ошибка распознавания речи во многом зависит от наличия посторонних сред и высокочастотных шумов, а также от качества микрофона.
Ключевые слова: *голос, распознавания речи, программное обеспечение, взаимодействие человека и компьютера*

Дерев'янченко В.С. – аспірант кафедри комп'ютерних наук та інженерій СНУ ім. В.Даля
Білобородова Т.О. – доцент кафедри комп'ютерних наук та інженерій СНУ ім. В.Даля, кандидат технічних наук, ORCID ID: 0000-0001-7561-7484
Скарга-Бандурова І.С. – завідувач кафедри комп'ютерних наук та інженерій СНУ ім. В.Даля, професор, доктор технічних наук, ORCID ID: 0000-0003-3458-8730
Фурса П.С. – магістр кафедри комп'ютерних наук та інженерій СНУ ім. В.Даля
Коверга М.О. – магістр кафедри комп'ютерних наук та інженерій СНУ ім. В.Даля