ListenSmart: A Voice Recognition and Conversation Project.

Student's name

Figure 1. A disabled man using the Voice Recognition software. Photo credit: Robin Christopherson

ABSTRACT

Speech recognition technology is one of the fastest growing engineering technologies around the globe. Medical, process control, communication, electricity and other computer-based systems are becoming reliant on this aspect of computer software because it holds a plethora of benefits to users. One outstanding benefit is that it enables hands-free control of different devices and equipment. With nearly 20% of the world's population having one form of disability or the other, this makes it an essential assistive tool in digital accessibility and inclusion. ListenSmart is a voice user interface designed to significantly help disabled users in sharing information with others and operating computers through voice input, the basic building block of speech recognition.

ListenSmart recognizes speech and converts the audio input into text. Additionally, it enables users to enter voice commands for actions like, "open, save and exit" when using a file. Extensively, it assists the user in launching several system applications including Ms. Paint, Notepad, and Calculator. The application is developed using Python, SQL, and Pycharm to collect, manipulate and organize data in a way that will make ListenSmart user-friendly. It is available for Windows, Mac, Android, iOS, and Windows Phone devices. The program is modifiable, which means it can be upgraded in a subsequent study to cover more tasks. At its inception, however, an effort is made to help with the basic operations described above.

1. INTRODUCTION

“The great myth of our times is that technology is communication.”
—Libby Larsen, American composer.

The use of information and communications technology is becoming increasingly embedded in and interconnected with our everyday lives. From basic operations such as creating and editing documents, working with data in spreadsheets and databases, or creating PowerPoint presentations to advanced skills required for specialized work, individuals and organizations continue to rely on more computer software than they have ever done before. This reliance has continually transformed lives and highlights the importance of digital inclusion. Today, digital inclusion is becoming a near-universal ambition. In the United Kingdom, the government is exponentially increasing its investments in the digital sphere so disabled users are not left behind and can effortlessly undertake local or global job roles. Several tech organizations are supporting this endeavor by creating accessible formats to mitigate the challenges faced by many disabled people when using computers. This global commitment to an inclusive digital community is evident in the initiatives and solutions developed in recent years. One of such initiatives is incorporating a speech recognition component into communication applications.

Speech recognition, also known as speech-to-text is a notable and relatively new interdisciplinary subfield of computer science and computational linguistics. Through methodologies and technologies that allow computers to recognize and translate spoken language into text, users primarily benefit in the scope of productivity, mobility, and essentially inclusion. It provides users with varying application options such as voice dialing (e.g., “Call Mary”), call routing (e.g., “I would like to make a call”), domotic appliance control, searching keywords (e.g., find a podcast where certain words were spoken), simple data entry (e.g., entering the digits on a credit card), preparation of structured documents, speech-to-text processing (e.g., emails and word processors), and aircraft (often termed as direct voice input).

Speech recognition has evolved over the years, and this is apparent in the explosion of voice recognition applications today. With newer ones looming somewhere in the coming years, these applications will not only enable voice control of PC and voice-to-text conversion, but support multiple languages, provide a variety of speaker voices for users to choose from, and integrate them into every aspect of their mobile device. Presently, Sensory's Trulyhandsfree Voice Control can hear and understand users even in noisy settings. This suggests a future when users will be able to control appliances like coffee makers, printers, and lights by voice command.

In light of this, ListenSmart has been developed. Through certain methodologies and technologies, it can translate speech into text and respond to voice commands. For the scope of this project, however, the developer puts into perspective basic computer operation patterns and their interfering variables. Although certain context-specific and high variables in human speech are remotely considered while developing the voice user interface. These are the variety of speech patterns, speaking styles, languages, dialects, accents, and phrasings. Through this, the developer hopes to lay the groundwork for a future voice user interface that will embody these complex features.

As indicated earlier, ListenSmart is designed in every sense of speech recognition, which means it captures sounds and turns them into written language that computers can understand and respond to. To achieve this, it simply adopts these essential four steps:

1. analyze the audio;
2. break it into parts;

3. digitize it into a computer-readable format; and
4. use an algorithm to match it to the most suitable text representation.

Therefore, ListenSmart can perform basic tasks such as opening YouTube or Gmail. It can also get top headline news and answer computational and geographical questions too. As the abstract of this project has rightly indicated, people with disabilities are the primary beneficiaries of this voice-controlled application. Nonetheless, it empathizes with a wider audience. From language instruction to hands-free activities, it offers a comfortable lifestyle, simplifies tasks, and enables productivity.

PROJECT'S OBJECTIVE.

This study aims to:

1. Understand speech recognition and its fundamentals.
2. Outline its application in different areas.
3. Examine tools and techniques used in speech recognition.
4. Implement ListenSmart as a desktop application
5. Develop software that can mainly be used for:
   A) Speech Recognition
   B) Speech Generation
   C) Text Editing

2. BACKGROUND

"Artificial intelligence? Perhaps. But people rarely make me smile or laugh. Alexa rarely fails to do so. And the enjoyment I get from having her in my home is anything but 'artificial.'" —Carla Martin-Wood, a very happy Amazon Alexa customer on Amazon.com

2.1. Understanding Speech Recognition.

Figure 2. *Speech recognition.*

A program's capacity to recognize spoken words and translate them into readable text is identified as speech recognition, also known as speech-to-text. Basic speech recognition software can only pick out words and phrases that are spoken clearly and has a small vocabulary. More advanced software can handle natural speech, various accents, and multiple languages.

To process spoken language, speech recognition systems use linguistics, mathematics, deep learning, and statistics. The program transforms the speech input into word output using statistical models or neural networks. Natural language processing (NLP) plays a crucial role in this process because it is used to deliver pertinent text in response to voice commands. The following processes are taken by computers to translate human speech:

- Electrical signals are produced by the microphone from sound vibrations.
- The received signals are digitalized by the computer.
- To recognize sounds and discriminate phonemes, speech recognition software examines digital signals (the smallest units of speech).
- Algorithms pair the signals with the appropriate text that corresponds to the sounds.

Accounting for background noise, context, accents, slang, crosstalk, and other influencing factors makes this approach more challenging. The processing of spoken interactions by speech recognition technology uses artificial intelligence and machine learning to increase performance and accuracy over time. Nonetheless, Speech recognition technology still requires that the user have a relatively consistent speaking voice because voice and breath stamina are considered when it is evaluated as a potential input method. Also, they are required to have moderately good reading skills to recognize errors and correct the program's text output. Voice and breath stamina are considered when it is evaluated as a potential input method.

Computer science, linguistics, and computer engineering researchers have all contributed to the field of speech recognition to aid easier or hands-free use of devices. Today, speech recognition features are built into many contemporary gadgets and text-focused software programs. These programs can provide an appropriate computer input method for individuals with a wide range of disabilities.

2.2 The History of Speech Recognition

Speech recognition has a much longer history than most people realize, dating back to the middle of the 20th century. Bell Laboratories developed Audrey, the first speech recognition system, in 1952. According to Pieraccini (2012), Audrey had rather basic technological skills and could only understand ten digits when they were spoken by specific individuals. The Shoebox Machine was created and unveiled by IBM about ten years later. The computer understood 16 different spoken words, including the ten digits "0" through "9" and calculations such as "plus" and "minus" (IBM, 2018). Shoebox became widely popular over the years, which has led to positive and quick advancements.

An example of speech recognition technology modelled after Shoebox is the Hidden Markov Model (HMM) which appeared in the middle of the 1970s (Rabiner, 1989). The HMM significantly changed the process of creating a workable speech recognition program. With the aid of HMM, speech recognition technology began utilizing a statistical technique to determine the likelihood that unidentified sounds were words. Due to the method's ability to increase the number of understandable words to a few thousand, the possibility of being able to recognize an infinite number of words becomes imminent.

From 1971 to 1976, the U.S Department of Defense made significant contributions to the development of speech recognition systems. It provided funding for the DARPA SUR (Speech Understanding Research) program. As a result, Carnegie Mellon created Harpy, a computer program that could understand 1011 words. It used a method for finding logical sentences that was more effective. Parallel technological developments also occurred, such as Bell Laboratories' creation of a device. As it could recognize the voices of multiple people, this device was revolutionary.

In the 80s, the creation of the Hidden Markov Model represented a significant advancement. To ascertain the likelihood that a word descended from an unidentified sound, this system used statistics. It didn't rely on speech patterns or predetermined templates. Many of these programs found use in commercial settings and industry. For example, in 1987, a doll for children was produced. It was called Julie, and youngsters could teach it to respond to their speech. However, the speech recognition technology of the 1980s had one drawback: one had to pause between each spoken word.

Nonetheless, voice recognition technology became practical with the development of faster microprocessors. The first consumer voice recognition system was introduced by the company Dragon in 1990 with the release of Dragon Dictate. They made improvements to it and produced Dragon NaturallySpeaking in 1997. Users could speak 100 words per minute with this solution.

When Google came along in 2001, the development of speech recognition technology took a different turn. Google developed an application called Google Voice Search that made use of data centers to compute the massive amount of data analysis required for matching user queries with actual examples of human speech. To create a better speech model, Google introduced personalized recognition for Android devices in 2010. This feature would record various users' voice queries and there are a bout 230,000,000,000 English words in it. In the end, cloud computing paved the way to create Apple's Siri.

Siri became instantly famous for her incredible ability to accurately process natural utterances, as well as her ability to respond using conversational language. She works well when it comes to sending text messages or carrying out simple commands, and its primary goal is to make device interaction easier. Nevertheless, it has drawn criticism for being difficult to understand (Van der Velde, 2018).

Siri’s success brought speech recognition technology to the forefront of innovation and technology. For example, Google Assistant and Siri have since been joined by Amazon’s Alexa as modern speech recognition technologies that have become part of everything from computers and smartphones to cars, fridges, watches and video games. Each can accurately react to a large range of enquiries and requests and is supported by a wide range of languages. To simplify life, these gadgets serve as hubs for connecting many gadgets and their associated apps.

2.3. Key Characteristics of Speech Recognition

The following characteristics are crucial for voice recognition systems to work:

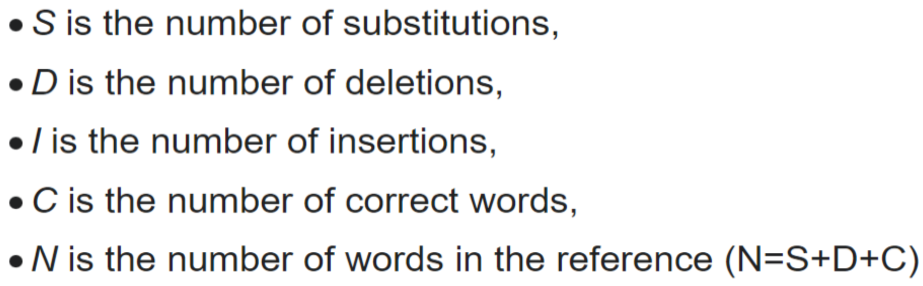
- Linguistic weighting: This function favors some words and phrases over others so that you can answer more appropriately in a particular situation. For instance, you may teach the software to focus on phrases that are particular to a given field or category of goods.
- Speaker identification: It identifies each participant in a conversation in order to highlight their unique contributions.
- Filtering for profanity: This feature detects and blocks offensive language.
- Acoustics training: Learns to filter out distractions by differentiating between background noise, speaker style, tempo, and volume. In crowded call centers and offices, this capability is useful.

2.4. Speech Structure and Interference

The three stages of speech communication are creation, transmission, and reception. In Speech Recognition, acoustic distortions brought on by gender, age, microphone, room, and other elements are present at every phase and are unavoidable. These differences play interfering roles in transmission and reception. For a Speech Recognition System (SRS) to be useful, its accuracy must be high. It can be difficult to reach a high level of accuracy, though. In a recent survey, accuracy was cited by 73% of participants as the largest barrier to adoption of voice recognition technology.

Word error rate (WER) is a regularly used metric to assess a voice recognition system's performance and accuracy. WER accomplishes this by adding the words the system omitted or mispronounced using an equation:

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C}$$



- *S* is the number of substitutions,
- *D* is the number of deletions,
- *I* is the number of insertions,
- *C* is the number of correct words,
- *N* is the number of words in the reference (N=S+D+C)

The SRS's ability to function with many languages, accents, and dialects is another substantial interference. The number of languages spoken worldwide exceeds 7000, and there are countless accents and dialects. More than 160 different dialects of English are spoken worldwide. No SRS can completely cover them. Even aiming for compatibility with only a few of the most widely used languages might be difficult. A key barrier to implementing speech recognition technology, according to 66% of respondents in the same study, was challenges with accent or dialect. Extending the dataset and aiming for optimal training for the AI/ML model that drives the SRS are two practical solutions proposed over the years to address this problem. Hence, one may want to deploy their SRS in more nations or areas.

2.5. Models in Speech Recognition.

Speech Recognition software often uses varying models to carry out complex tasks. An example of a complex task is its ability to distinguish audio from background noise that often accompanies the signal. Carrying out such complex duties can be done through either acoustic or language models. Acoustic models represent the relationship between linguistic units of speech and audio signals. On the other hand, language models match sounds with word sequences to distinguish between words that sound similar.

2.6. Alogrithms and Technologies in Speech Recognition

The complexities of human speech have made development of speech recognition software challenging. It is a complex area of computer science because it involves linguistics, mathematics and statistics. Speech recognizers are made up of a few components, such as the speech input, feature extraction, feature vectors, a decoder, and a word output. The decoder leverages acoustic models, a pronunciation dictionary, and language models to determine the appropriate output.

As mentioned earlier, Speech recognition technology is evaluated on its accuracy rate, i.e., word error rate (WER), and speed. Several factors can impact word error rate, such as pronunciation, accent, pitch, volume, and background noise. Therefore, reaching human parity – meaning an error rate on par with that of two humans speaking – has long been the goal of speech recognition systems. To achieve this, a group of algorithms and technologies are responsible for speech recognition features and superpowers. They consist of the following:

- Hidden Markov model: HMMs are used in autonomous systems where a state is only partially observable or when the sensor (in the case of speech recognition, a microphone) does not immediately have access to all the data required to decide. In acoustic modeling, for instance, a program must use statistical probability to match linguistic units to audio signal.
- Natural Language Processing: it is an AI method of communicating with intelligent systems using a natural language such as English. This processing is essential when a developer is intent on giving instructions to an intelligent system like robot to perform, or when the developer wants to hear decisions from a dialogue based clinical expert system. The field of NLP involves making computers perform useful tasks with the natural languages' humans use. The input and output of an NLP system can be in either speech or written text. Speech recognition is made simpler and faster by NLP.
- N-grams: A probability distribution for a sequence is produced using this straightforward method for language models. It is an algorithm that performs a variety of functions such as analyzing the most recent few words spoken, roughly reconstructing the history of the sample of speech, and then using that information to estimate the likelihood of the next word or phrase.
- Artificial intelligence: Modern speech recognition software frequently uses AI and machine learning techniques like deep learning and neural networks. To process audio and voice signals for speech, these systems use grammar, structure, syntax, and signal composition. Machine learning systems are well suited for nuances like accents because they learn more with each use.
- Neural networks: Primarily leveraged for deep learning algorithms, neural networks process training data by mimicking the interconnectivity of the human brain through layers of nodes. Each node is made up of inputs, weights, a bias (or threshold) and an output. If that output value exceeds a given threshold, its “fires” or activates the node, passing data to the next layer in the network. Neural networks learn this mapping function through supervised learning, adjusting based on the loss function through the process of gradient descent. While neural networks tend to be more accurate and can accept more data, this comes at a performance efficiency cost as they tend to be slower to train compared to traditional language models.
- Speaker Diarization (SD): Speaker diarization algorithms identify and segment speech by speaker identity. This helps programs better distinguish individuals in a conversation and is frequently applied at call centers distinguishing customers and sales agents.

2.7 Types of Speech Recognition Software

Speech recognition comes in two varieties, speaker-dependent and speaker-independent. While speaker-independent software is more frequently used in telephone applications, speaker-dependent software is frequently used for dictation software. Like voice recognition, speaker-dependent software functions by learning the distinctive qualities of a single person's voice. For the computer to analyze how people speak, new users must first "train" the software by speaking to it. This frequently means that users must first read a few pages of text aloud to the computer before using speech recognition software. Speaker-independent software, however, does not require such training. It is made to recognize voice automatically. Because businesses cannot require callers to read pages of text before using the system, it is the only practical option for applications like interactive voice response systems.

The drawback is that speaker-independent software is typically less accurate than speaker-dependent software. This is typically addressed in speaker independent speech recognition engines by restricting the grammar employed. Since speech engines are more likely to accurately recognize what a speaker says by using a smaller list of recognized words, speaker-independent software is ideal for most IVR systems and any situation in which many users will share the same system. In dictation software, where there will be only one user and a large grammar requirement, speaker dependent software is more commonly used.

2.8. Notable Works in Recent Years.

- IPhone's Siri: One of the first and best-known voice command services, Siri, was made available to the public via the iPhone. It was first made available on the iPhone in 2011, and it was quickly added to practically all Apple products (Apple Inc, 2018). Siri's primary goal is to make using the smartphone easier, and it excels at delivering text messages and carrying out simple commands. Although there have been complaints about its comprehension (Van der Velde, 2018).
- Amazon’s Alexa: One of the most well-known voice assistants on the market today is Alexa, which was created by Amazon.com INC and first introduced in 2014. Alexa's primary goal is to expand its platform by being unrestricted and integrating into a variety of devices.
- Microsoft’s Cortana: Released in 2014, Cortana is modeled after actual, live personal assistants. Competing services can locate data from devices, like search history and cookie tracks, making it possible for the smart system to find and register information about its users (Microsoft, 2018).
- Google's Assistant: Google LLC created Google Assistant, which was first made available in 2016. It is largely accessible through mobile and smart home devices, and it supports two-way communication. As the assistant is supported by Google's robust search technology, it not only provides accurate answers to questions but also provides more context and references (Whitwam, 2016). In such a short period of time, Google Assistant has made significant progress in catching up to Alexa, boasting a 95%-word accuracy rate for American English. This equals a 4.9% error rate, the lowest of all voice assistants currently available (Van der Velde, 2018).

3. DESIGN

“The lexical models are built by stringing together acoustic models, the language model is built by stringing together word models, and it all gets compiled into one enormous representation of spoken English, let’s say, and that becomes the model that gets learned from data, and that recognizes or searches when some acoustics come in and it needs to find out what’s my best guess at what just got said.”

— Mike Cohen, Manager of Speech Technologies at Google.

3.1. Design Methodology

It is crucial to design the voice user interface so that it provides the proper amount of information and gracefully handles the user's expectations even though it cannot entirely live up to the user's expectations of a genuine conversation partner. Consequently, the developer was open-minded about the technologies to utilize because they had no prior expertise creating voice user interface. A few different tools were examined to ensure the selection of the best options. However, only frameworks and techniques that had a good chance of programming ListenSmart effectively and can be supported in the future were taken into consideration.

Python is the primary programming language used to build ListenSmart. Although there were other programming languages available to the developer, such as JavaScript and PHP, Python was most preferred because its straightforward and simple syntax readily suits the design scope of ListenSmart. PyCharm was adopted by the developer to complement Python's use. For ListenSmart, it was primarily used for code inspection, on-the-fly error highlighting and quick fixes. Finally, SQL (Structured Query Language) is used by the developer for high-speed retrieval of data and connectivity. The developer also used it to separate databases into more manageable components to analyze every detail in the process.

To recognize phenomes and phonetics in speech, acoustic modelling was adopted. It allowed the developer to capture the most significant part of speech, such as words and sentences. On the other hand, Natural Language Processing and Neural Networks are the algorithms adopted to analyze sounds from audio data.

After careful consideration of the design methods, the developer started by collecting sound energy produced by the person speaking into a microphone. Next, the input is converted from the microphone into electrical energy. He then converts this electrical energy from analog to digital, and finally to text that can be read and responded to by the computer.

3.1.1 An Overview of Programming Tools Used.

Python

Python is a popular computer programming language used to create software and websites, automate processes, and analyze data. Python is a general-purpose language, which means it may be used to make many various types of applications and is not tailored for any issues. For this project, Python provides an API called Speech Recognition to allow the developer to convert audio input into text for further processing. It also provides the developer with Pyttxs3, a text-to-speech conversion library. Green and Requestium are python tools used to test the software.

PyCharm

PyCharm is one of the most popular Python IDEs (Integrated Development Environment) developed by JetBrains. The Linux, macOS, and Windows operating systems are all compatible with PyCharm. Among the top Python IDEs, PyCharm offers support for Python 2 (2.7) and Python 3 (3.5 and higher) versions. PyCharm includes a wide range of modules, packages, and tools to speed up Python programming while also significantly reducing the amount of work necessary to do it. Also, PyCharm can be modified to meet specific preferences and development needs. Its plethora of productive shortcuts, ability to view the entire

Python source code with a single click and availability of an array of plugins among other benefits are the reasons why the developer has chosen PyCharm.

Structured Query Language

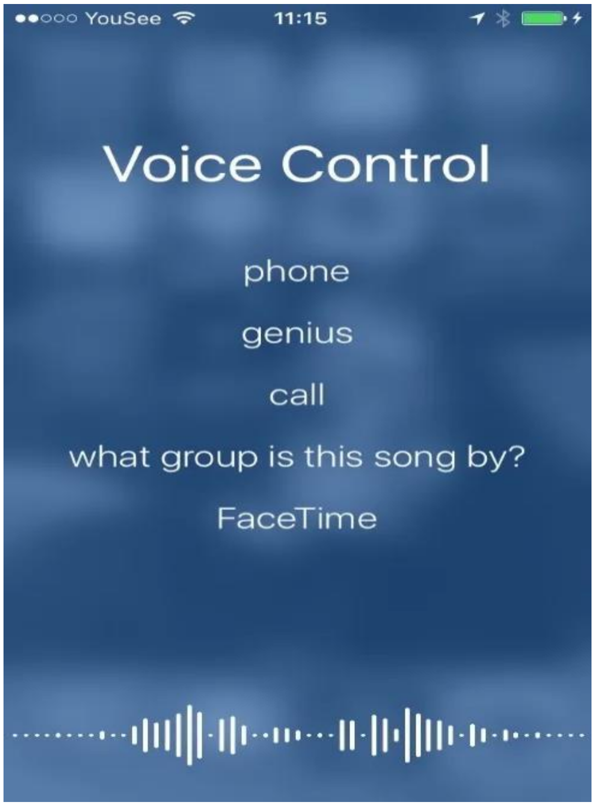
Structured query language (SQL) is a programming language for storing and processing information in a relational database. A relational database stores information in tabular form, with rows and columns representing different data attributes and the various relationships between the data values. SQL statements can be used to store, update, remove, search, and retrieve information from the database. It can also be used to maintain and optimize database performance. The developer chose to use SQL because it integrates well with different programming languages, python inclusive, and uses common English keywords in its statements.

3.1.2 Language Option

The default language in ListenSmart is English. Users interact with ListemSmart in English, adopting a user-centric (resembling natural human conversation) approach.

3.1.3 Visual Interface

Some tasks are inefficient or impossible to complete using only the voice. Listening to and browsing through search results by voice, for example, can be tedious. Therefore, It is critical to consider the visual aspect of user interaction because high-quality visual experiences create positive user impressions. The end goal is to create a more enjoyable and engaging multimodal experience.



Voice Control

phone

genius

call

what group is this song by?

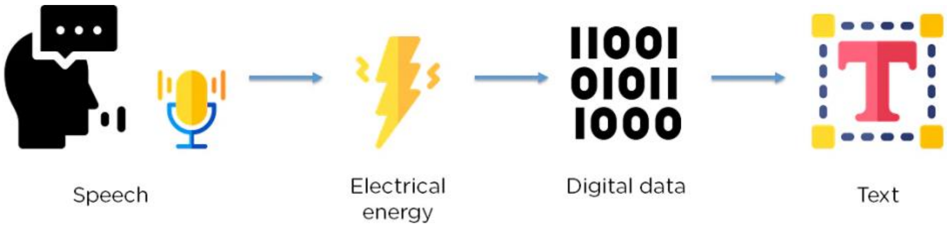
FaceTime

Figure 3. Author/Copyright holder: Apple Incorporated. Copyright terms and license: Fair Use.

3.1.4. Personalization

Personalization is more than just saying "Welcome back, %username%". Personalization entails understanding genuine user needs and desires and tailoring information to them. In this case, the system should be capable of recognizing new and returning users, creating user profiles, and storing the information gathered by the system. The developer's goal is to allow users to personalize their entire interaction. However, it necessitates a lengthy and intensive procedure, such as the types of information to collect from users in order to personalize the experience. Nonetheless, the developer ensured that ListenSmart provides users with a personalized experience by allowing the use of their names.

3.2. Design Process and Outcome



4. IMPLEMENATION AND TESTING

5. EVALUATIONS

6. CONCLUSION AND RECOMMENDATION

References.

1.