

A Real-Time Automatic Translation of Text to Sign Language

Muhammad Sanaullah^{1,*}, Babar Ahmad², Muhammad Kashif², Tauqeer Safdar², Mehdi Hassan³,
Mohd Hilmi Hasan⁴ and Norshakirah Aziz⁴

¹Department of Computer Science, Bahauddin Zakariya University, Multan, 60,000, Pakistan

²Department of Computer Science, Air University, Multan, 60,000, Pakistan

³Department of Computer Science, Air University, Islamabad, 44,000, Pakistan

⁴Centre for Research in Data Science, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Seri Iskandar, 32610, Perak, Malaysia

*Corresponding Author: Muhammad Sanaullah. Email: drsanaullah@bzu.edu.pk

Received: 13 April 2021; Accepted: 27 May 2021

Abstract: Communication is a basic need of every human being; by this, they can learn, express their feelings and exchange their ideas, but deaf people cannot listen and speak. For communication, they use various hands gestures, also known as Sign Language (SL), which they learn from special schools. As normal people have not taken SL classes; therefore, they are unable to perform signs of daily routine sentences (e.g., what are the specifications of this mobile phone?). A technological solution can facilitate in overcoming this communication gap by which normal people can communicate with deaf people. This paper presents an architecture for an application named Sign4PSL that translates the sentences to Pakistan Sign Language (PSL) for deaf people with visual representation using virtual signing character. This research aims to develop a generic independent application that is lightweight and reusable on any platform, including web and mobile, with an ability to perform offline text translation. The Sign4PSL relies on a knowledge base that stores both corpus of PSL Words and their coded form in the notation system. Sign4PSL takes English language text as an input, performs the translation to PSL through sign language notation and displays gestures to the user using virtual character. The system is tested on deaf students at a special school. The results have shown that the students were able to understand the story presented to them appropriately.

Keywords: Sign language; sign markup language; deaf communication; hamburg notations; machine translation

1 Introduction

The digital division of the world between people who have access and who do not have access leaves the deaf community behind by creating e-inclusion issues. According to WHO [1,2], there are approximately 466 million deaf people worldwide, with the fact that 5% population of every



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

country is deaf. According to these statistics, Pakistan has 10 million deaf people out of 200.81 million populations.

To make deaf people part of the global community, the technological advancements will allow them to interact and understand daily life information such as shown in media and TV channels. It is difficult for the deaf community to understand the spoken language completely. To fully understand and acquire the knowledge of any spoken language, sign languages exist to help the deaf community in their intellectual development.

Even daily life activities such as watching television and reading online about the weather forecast in their native language are challenging for a deaf person. However, if the same information is shown to him/her in Sign Language, the person will be able to understand the information. In this regard, to help the deaf community, many countries have developed applications that convert written text to their respective sign language. However, these applications are limited to alphabets, particular words. In contrast, the research on the dynamic construction of sentences from a knowledge base is limited. Furthermore, these applications are not platform-independent, high in cost, and do not provide a wide range of services.

Unfortunately, no such application exists for Pakistan Sign Language (PSL) to facilitate deaf people. Media, society, and institutes in Pakistan are dependent on human translators having PSL knowledge. The development of such an application will increase the intellectual level as it will open the doors of education, culture, and social interaction for normal people and the ones with special needs. The interpretation of PSL content is also helpful in healthcare, cultural, and entertainment context. Therefore, any application that is low in cost, lightweight, generic, platform-independent, and provides a wide range of services to PSL content will be helpful for the deaf community. The development of such an application is the motivation of this paper.

This paper provides Sign4PSL to support deaf people by showing a gesture performed by virtual signing character hereafter called "Avatar." The scope of this research is a sentence translation from Text to PSL synthesis via an Avatar using a sign-writing notation to generate gestures for Avatar.

The system is tested by demonstrating the textual stories via PSL using Avatar to deaf students in special education schools of Multan, Pakistan. The results have shown that students were able to understand the story properly.

- In summary, the contributions of this paper are as follows: A novel architecture that converts text to visual representation/gestures of Pakistan Sign Language (PSL). The architecture is used in an application named Sign4PSL.
- The proposed architecture is validated separately on alphabets, digits, words, phrases, and sentences. The results have shown that the architecture achieves a 100% accuracy on alphabets, digits, words, and phrases. However, for sentences, the architecture achieves 80% accuracy.
- The Sign4PSL is also tested on deaf students at a special school. The results have shown that all the students understood the story presented to them using the Sign4PSL application.

The rest of the sections in the paper presents concepts related to signs and gestures and technology used in the literature survey to provide the solution. The developed methodology follows this, text preprocessing, PSL grammar, machine translation, adapted animation generation engine, implementation of these components, and conclusion and future improvements.

2 Related Terminologies

2.1 Sign Language and Gestures

A deaf person uses his/her hand to deliver the message with the help of a specific gesture. Different movements of hands create a gesture with a specific meaning. The different gestures combine and make sentences. These sentences can be understood by deaf people. These gestures are categorized into manual and non-manual gestures. Manual gestures involve hand shape, position, movement, location, and orientation [3]. Whereas, non-manual gestures are related to facial expressions like the movement of lips, eyes with eyebrows and lid, body, and posture movements, such as shoulder raising and head-nodding [4]. Each gesture has a specific meaning that represents some letter, word, or expression. The meaning of any gesture varies for each country and even the regions within the country. For example, one can compare a sign for the word “what” in American Sign Language (ASL) and PSL shown in Fig. 1.



Figure 1: ‘What’ sign in ‘ASL’ and ‘PSL’

2.2 Sign Writing Notations

Sign writing notations are helpful in representing the words into a format that can be used in automated solution and translation of animations. The most famous and widely used notations are Stokoe, Gloss, Sign Writing, and HamNoSys [5]. Each notation method has its pros and cons. A comparison of notations is given in Tab. 1, in which HamNoSys is found most suitable for a lightweight solution.

Table 1: Comparison of famous sign writing notations

Notation name	Language dependency	Non-manual signs support	Used for	Computer supported encoding	Signs arrangement
Stokoe	Yes	No	Academic	ASCII Codes	Linear
Gloss	Yes	Yes	Academic	ASCII Codes	Linear
SignWriting	Yes	A few	Public	ASCII and Unicode	Pictures
HamNoSys	No	Yes	Academic	Unicode based font	Linear

2.3 HamNoSys

HamNoSys is one of the best notations as it uses ASCII and Unicode symbols instead of pictures and videos of signs which minimizes the space cost and helps to store signs on the computer [6].

2.3.1 Hand Shapes

HamNoSys groups different hand shapes which include flat-hand, fist, separated fingers, and combinations of thumb [6].

2.3.2 Hand Orientation

Hand orientation combines two components: extended finger direction defines the orientation of the hand, axis and second define palm orientation for extended finger direction [6].

2.3.3 Hand Movement

Hand movements are divided into the curve and straight movement at a basic level. The hand can be moved straight, parallel to the body. These include wavy, zigzag, circular, and spiral motions [6].

2.4 Sign Gesture Markup Language (SiGML)

SiGML is XML-based [7] representation of signs. It describes Unicode symbols of HamNoSys in tags that are sent to the software for playing sign by Avatar.

3 Literature Review

Majority research propose different systems that work well for the sign language to help the deaf community. The research done on sign language processing is discussed in this section and summarized in [Tabs. 2](#) and [3](#). This work is analyzed and compared based on a few identified parameters as shown in [Tab. 3](#).

Table 2: Comparison based on sign language, tools and coverage of signs

Sr.	Ref.	Year	Sign language	Prog. language	Sign presentation	Avatar	Coverage/features	Limitation/drawbacks
1	[8]	2015	Sinhala	Python	Expr. Of Signs Def. in Files	Yes	Word, Phrase, Finger Spelling	MB, DSC, DC
2	[9]	2013	American	Java	Sign Writing Markup Language	Yes	Under Development	DSC
3	[10]	2017	American	–	XML-Gloss	No	Words, Finger Spell	DSC
4	[11]	2017	Arabic	–	–	Yes	Words	DSC
5	[12]	2016	Indian	Java	SiGML	Yes	Words	DSC
6	[13]	2015	Brazil	C++	XML	Yes	Words	MB, DC, DSC
7	[14]	2019	Arabic	–	Gloss System	–	Arabic Words, Stored Sentences	MB, DSC
8	[15]	2019	Vietnam	–	HamNoSys	Yes	Words	DSC
9	[16]	2019	Arabic	Rule Based	Video Sequences	Yes	Words	DSC

Table 3: Expanding [Tab. 2](#): Comparison based on software platform

Sr.	Avatar mesh	3D software	Animation software	Animation engine	Multiplatform
1	Low poly	MakeHuman	Blender	Blender game engine	No
2	Low poly	–	–	WebSign player	No
3	No	No	No	No	No
4	Low poly	–	–	–	No
5	Low poly	–	–	JA SiGML App	No
6	Low poly	Blender	Blender	Irrlicht	No
7	–	–	–	–	No
8	Low poly	–	–	–	No
9	Low poly	–	–	–	No

These parameters help in the evaluation of the system concerning services and support it provides. The system is compared based on the software platform, technology in use, sign language, tools, and coverage of the signs. Parameters ‘Avatar Mesh and Avatar’ help in identifying the computer graphics as low poly characters are fast to render as compared to high poly characters used in animated movies. Parameter ‘Signs Coverage’ helps in identifying whether the system supports sentences and language features or not. Parameters ‘3D Software, Animation Software and Animation Engine’ help in finding whether signs animation are created manually for video sequences of Avatar or has used any engine to control skeleton and animation state to create animation at run-time. Parameter ‘Multi-platform’ helps in identifying whether the system runs on any platform like on ‘Mobile’ and ‘PC’ or not. Parameter ‘Sign Presentation’ helps in identifying which notation is used in the system to represent the sign language. This helps in finding the weakness of the notations that results in the scarcity of features in system.

Punchimudiyanse et al. [8] developed a system in python and blender for deaf in Sinhala sign language (SSL). The system can animate any sentence in the Sinhala language. This system was not based on video-sequences or any motion capturing hardware. Similarly, research on PSL has also been done but that involves video sequences. Bouzid et al. [9] developed a web-based signing system for American sign language (ASL). This system uses SignWritingmarkup Language to represent ASL signs and WebSign players as an animation engine for low poly 3D Avatar.

Othman et al. [10] presented the transcription system which annotates the American sign language. They show that their system would be helpful in machine translation in order to use it in different fields. For this, they present a new XML representation of sign. This new representation is based on annotation based Gloss system which is usually used in American sign language. Al-Barahamtosha et al. [11] developed a system for Arabic Sign language (ArSL) which uses speech module to capture the words and then translate them into ArSL using transition module. The goal of this system is to translate Arabic text to ArSL. The proposed technique uses Arabic language model and set of transformational rules to translate the text into ArSL. Literature indicates that researchers are using notations and XML based languages for sign automation.

Kaur et al. [12] created the Sign representation for Indian sign language using SiGML technology for the Indian sign language words. They covered only words for Indian sign language and stored them in SiGML format. For animation they used JA Signing App to test the signs representations. Similarly, for Brazilian sign language Gonçalves et al. [13] developed an automatic

synthesis system with 3d avatar in c++ language. The system gets Brazill signs in XML format as an input and sign it using 3d avatar. There system covers words for Brazil sign language. Their system was based on Irrlich animation engine. They created the animations in blender.

Luqman et al. [14] mentioned the sign representation for Arabic sign language in Gloss System. Their work shows that Arabic is a fully natural sign language. Their language has a different language structure lexicon and word order. They cover the Arabic signs and some sentence representations in Gloss System. In 2019, Da et al. [15] presented the paper for Viennese sign language. For this they converted the television news into 3D animations. Their work shows that Viennese words are store in Hamburg Notation and are converted to animation according to news script.

In 2019, Brour et al. [16] used the old fashioned way of showing animation of sign language. They stored the video sequences of Arabic sign language and used a ruled based system to play the animation of the words.

[Tabs. 2](#) and [3](#) presents a summary of related works along with their limitations. We categorize the limitations of related works. The explanation for each limitation is as follows:

- Dependency Chain (DC): In the existing solutions, a chain of dependency is found, which requires a cost-effective 3D software with animation experts along with a sign language expert for the recording of any word or sentence.
- Memory Bound (MB): Existing solutions use a high amount of storage memory and bandwidth.
- Dynamic Sentence Creation (DSC): Existing solutions work on pre-recorded words/sentences. For any new sentences, the existing solutions are not able to extract the gestures for those sentences.
- Multi-Platform (MP): Existing solutions are developed for specific desktop/mobile-based applications; therefore, they are platform-dependent (e.g., Android, Windows, Linux).

4 Methodology

Sign4PSL comprises all the standardized gestures, HamNoSys notations, PSL signs to HamNoSys transcription, and transcription to SiGML conversion. The Sign4PSL converts letter, word, phrase, and sentence to PSL. The architecture for Sign4PSL is represented in [Fig. 2](#). The methodology shows different modules with different functioning and interaction of each module. The user inputs a sentence in the form of text, and it is transferred to the lexical module of Sign4PSL. Later, it is passed to the stemmer and lemmatizer module for further sentence formation. After that, the process of sequence generation and sign production starts.

4.1 Participant Roles

Sign4PSL has two participating parties: one is the user who enters the sentence and expects the avatar to display the sign and the second party is the sign language expert who takes care of the PSL signs in the knowledge base, their notations, verification, and conversion.

- Managing of PSL signs
- Adding Hamburg Notation for new signs
- Verification of SiGML and Hamburg notation by matching it to PSL sign video
- Converting Hamburg Notation to SiGML

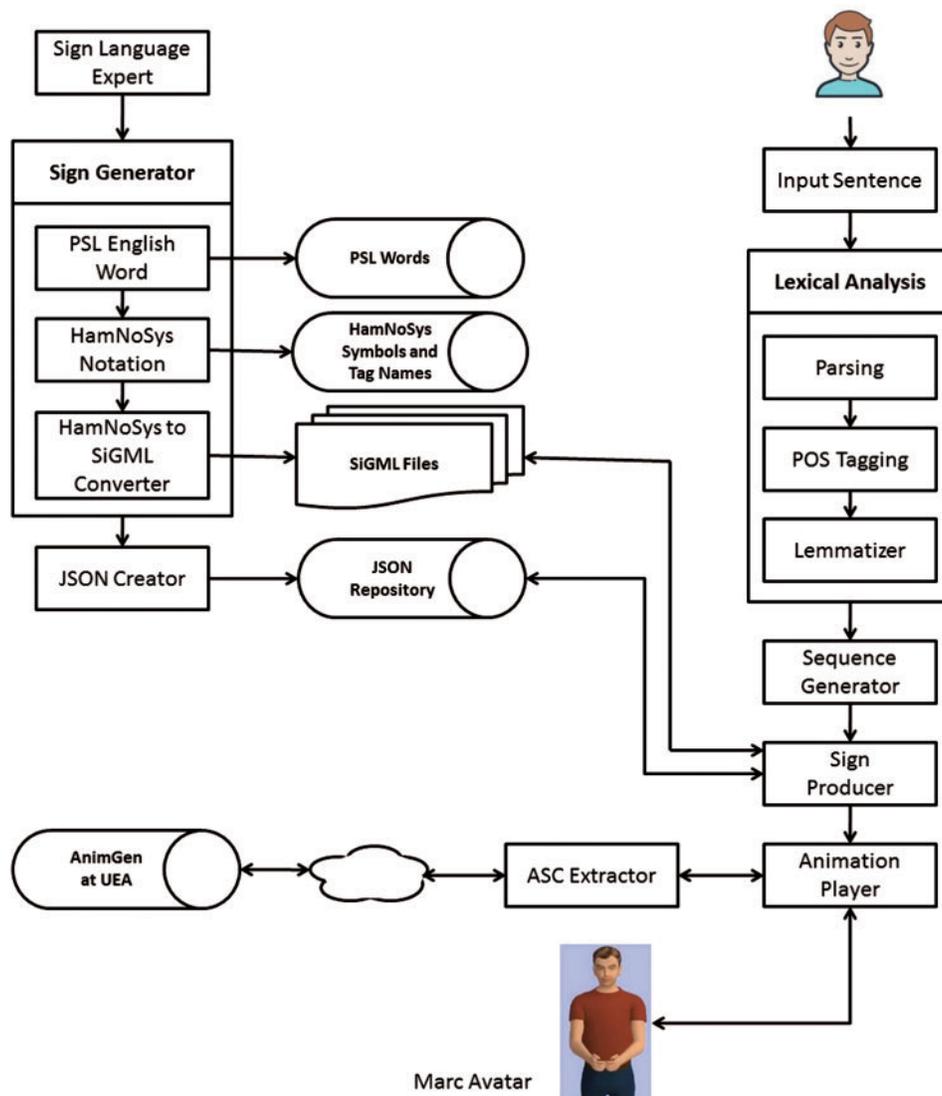


Figure 2: Architecture of Sign4PSL

4.2 Participant Roles

The knowledge base consists of a JSON structure where unique sign id, gloss name and SiGML data is stored. As mentioned in Participant Roles, a sign expert uses an interface to write HamNoSys of the sign. Moreover, the sign expert converts the HamNoSys into SiGML. After converting HamNoSys into SiGML, every notation/sign is converted to SiGML tags and automatically added to the knowledge base.

4.3 Lexical Analysis

4.3.1 Parsing and POS Tagging

LA parse and breaks text into a sentence and further break the sentence into tokens of the words and removes any white spaces, comments, and emotion. For example, the sentence ‘This car

mechanic lives in Dera Ismail Khan.’ will be broken into tokens (this, car, mechanic, lives, in, Dera Ismail Khan). During parsing, words like ‘in’ are removed. PSL does not use preposition like ‘on’, ‘in’, ‘under’ as much as the English language does. After parsing, part of speech (POS) tagging is done on generated tokens. POS tagging identifies the parts of speech of the sentence for generating sentence structure for PSL. Then, these tokens are passed to the stemmer and lemmatizer module to generate a base form of the word.

4.3.2 Stemming and Lemmatization

Stemming means removing the part of a word using rules and convert it to the base form, whereas lemmatization is the process of resolving words to their dictionary forms. For this purpose, it uses the WordNet database to transform the words to their base form called a lemma. For example, the word ‘walk’ may appear as ‘walking,’ ‘walked,’ ‘walks’ in a sentence, and its base form is ‘walk.’ Stemming also helps to get the base form of a verb but sometimes changes the meaning of the word also after stem might not be an actual word, e.g., “Wander” changes to “Wand.”

4.4 Sequence Generator

After converting tokens to the base form, this module finds a sequence for PSL words to play PSL signs according to the generated sequence. It ensures PSL sentence structure using rewriting rules for English and signs language grammar. Any understandable arrangement of signs can form the PSL sentence and structure. However, grammar concerns for proper structuring of PSL sign language are yet not available. For example, sentence ‘He goes to school,’ the structure in PSL sentence would be ‘he go school’ or ‘he school go’ or ‘school he go.’ However, there must be some rules for sentence structure; otherwise, the meaning of the sentence would have changed. For example, the rule “NN -> {love | like | fond} -> NN” is important because changing arrangement can change the meaning of a sentence. Like in a sentence, ‘Hamid likes Ayesha,’ should be played in PSL as ‘Hamid likes Ayesha’ instead of ‘Ayesha likes Hamid’ because it will change the entire meaning of the sentence, which shows Hamid’s one side affection towards ‘Ayesha’. Such language features with comparison are presented in [Tab. 4](#).

Table 4: Comparison of English language and PSL features

Sr.	English grammar	PSL grammar
1	It is well structured and has grammar rules to define sentences. English language has Subject-Verb-Object order	The creator of PSL are deaf people. Some signs are borrowed from other languages. Little work is done on PSL and there is no proper grammar for sign language.
2	English language has tenses, different form of verb, adjectives, and inflections according to type of sentence.	There is no concept of inflections in PSL. Rather, it uses root form of words.
3	Dictionary of English is very large.	Dictionary of PSL is limited.
4	English uses helping verb, conjunctions, and articles to make sentences.	In PSL, there is no concept of helping verbs, conjunctions, and articles. Instead, they are ignored.

4.5 Sign Producer

This module finds the signs stored in the knowledge base, and if the sign is available, it is added into the playing sequence; otherwise token goes to the sub-module where it is broken into letters, and then the sub-process starts. If the letter is found in the knowledge base, it is added into the playing sequence; otherwise, an error is generated to inform the sign language expert. If a token is not found or in the case of a noun, the process of finger spelling starts. Finger spelling means signing the letters of the token. The flow of the whole process is shown in Fig. 3.

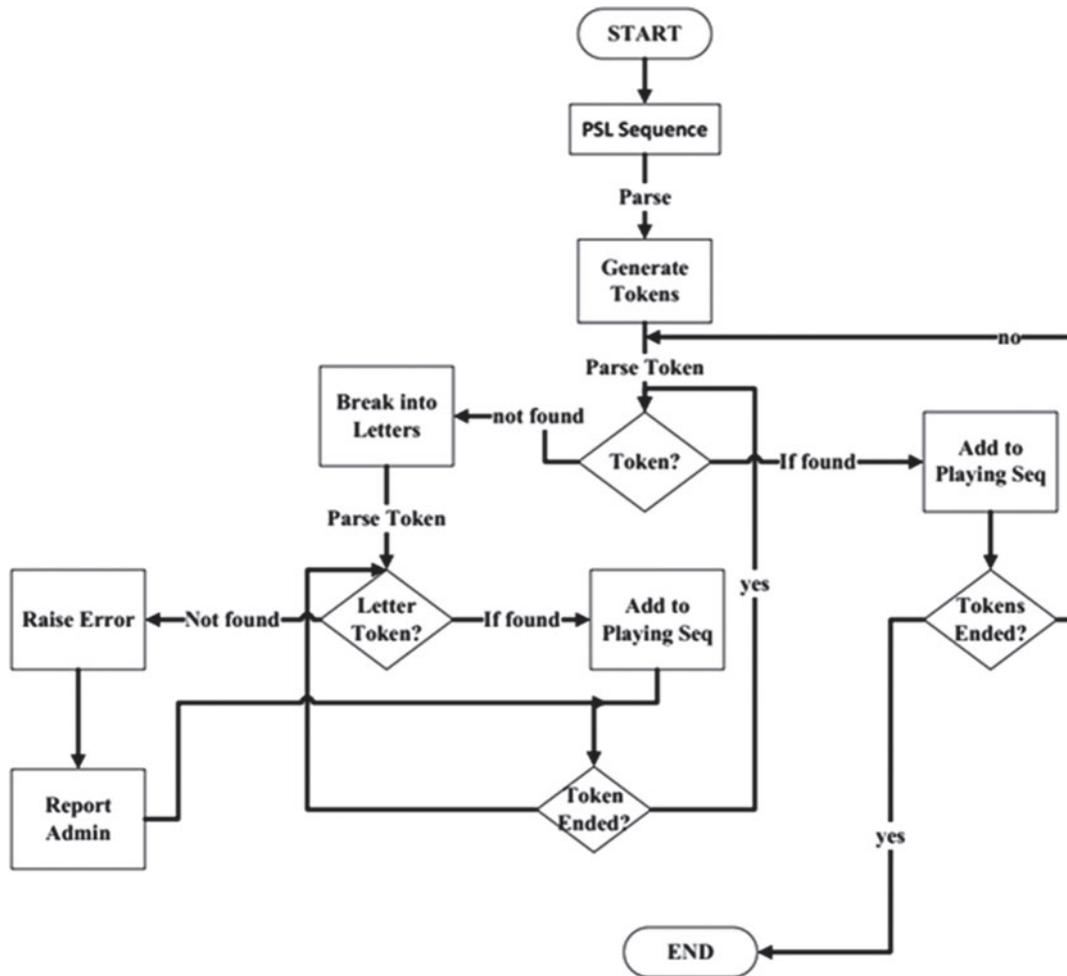


Figure 3: Sign producer flow

Sign producer was first implemented using MySQL storage. Later, it was changed to JSON knowledge base mechanism for better performance as new signs will be added in the future and therefore sign dumping process will take a long. The sign producer takes the signs information in the form of JSON and starts matching the tokens. In case of success, it is added into the playing sequence.

4.6 Sign Generator

This module converts PSL signs into SiGML, as shown in Fig. 4. The whole process follows a sequence of steps: First, the word is converted to HamNoSys Notation transcription according to sign specifications. Later, the transcription is converted to SiGML tags based on notation.

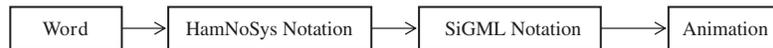


Figure 4: Sign generator

4.7 PSL Sign to SiGML

This module takes HamNoSys Notation as input and converts it into SiGML. Sign4PSL implements an algorithm (see Algorithm 1) which converts HamNoSys to SiGML notation automatically; it calls ‘hamConvert’ function and pass it two arguments, ‘ham’ and ‘gloss’. The first argument contains the Hamburg symbols written by a sign language expert, and the second argument contains words for which symbols are written. Algorithm gets all symbols and tags from the database in an associative array where the Hamburg symbol acts as key and tag as value. Further algorithm inserts the tags for symbols written by sign language expert one by one after matching keys. In the end, the algorithm concatenates the closing tags and return the SiGML form. The Hamburg notation for the word ‘hello’ and SiGML form generated from notation with their meaning can be seen in Tabs. 5 and 6.

Algorithm 1: Algorithm for converting Hamburg to SiGML

```

1 function hamConvert ham, gloss;
  Input : ham and gloss
  Output: SiGML
2 STEP1: Get all Hamburg symbols mapped to tags in an associative array from database ;
3 STEP2: Create string variable âsigmlâ with initial starting tags for gloss ;
4 foreach ham symbol as key do
5   | if key is in Associative array created at STEP 1 then
6   | | Concatenate tag mapped to that key at the end of âsigmlâ;
7   | end
8 end
9 STEP4: append closing tags at the end of âsigmlâ;
10 STEP5: return âsigmlâ;

```

4.8 JSON Creator

JSON is used to transfer data in the human-readable form to other modules for processing. This module converts all sign information, e.g., gesture information of a sign, name of a sign, unique id, etc., into a JSON knowledge base. This mechanism enables Sign4PSL to dump all information fast and once from local storage to main memory, increasing the performance.

4.9 ASC

Avatar Signing commands extraction (ASC) module sends SiGML XML form for PSL gloss to AnimGen client-server at UEA for the Avatar signing commands extraction.

Table 5: PSL word to SiGML

PSL word	Ham notation	SiGML
Hello		<pre> <sigml> <hns_sign gloss="hello"> <hamnosys_nonmanual> <hnm_mouthpicture picture="hV1U"/> </hamnosys_nonmanual> <hamnosys_manual> <hamflathand/> <hamthumboutmod/> <hambetween/> <hamfinger2345/> <hamextfingeru/> <hampalmd/> <hamshouldertop/> <hamlrat/> <hamarmextended/> <hamswinging/> <hamrepeatfromstart/> </hamnosys_manual> </hns_sign> </sigml> </pre>

Table 6: Meaning of symbols

Symbol	Meaning
	flat hand
	thumb finger opens outward
	Between: means thump in between fingers
	Represents index finger, middle finger, ring finger, baby finger
	Shows the direction right and diagonal
	Palm orientation
	Shoulder position top
	Means elliptic shape
	Left of
	Maximum distance extended arm
	Wrist/forearm swinging

4.10 Animation Player

The playing sequence from the sign producer module is passed to the animation player, where it plays each sign one by one according to the generated sequence. It also ensures Avatar states, i.e., busy and available. Sometimes signs data rush into the server, making Avatar halt. In this

case, Avatar is reset and made available to play the next sign. HamNoSys supports few Avatars from which we selected ‘Marc’ for PSL animations, as shown in [Fig. 5](#).



Figure 5: Marc Avatar from UEA of UK

4.11 Signing Avatar

Signing Avatar is an anthropomorphic virtual character that is used to represent signs. These signs can be captured from a human signer through a capture device, generated by some 3D software and by some notation.

5 Results

Sign4PSL has more than 400 basic signs of PSL, which uses manual features only. These are taken from the books of PSL [17]. Sign4PSL is tested with 26 alphabets, 10 numeric digits, 40 words, and 25 sentences, as shown in [Tabs. 7–12](#) respectively.

Table 7: Alphabets and digits validation

No.	Alphabets and digits	Compared PSL video	PSL expert
1	A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z	✓	✓
2	0, 1, 2.3.4.5.6.7.8.9	✓	✓

Table 8: Tested words validation

No.	Words	Compared PSL video	Validation by PSL Expert
1	Crow, Very, Thirsty, He, Search, Water, Here and There, At last, Saw, Garden, Pitcher, Low, Beak, Reach, Pebble, Rise up, Drink, Fly, away, AoA, Hello, Go, School, sit, Chair, Happy, Rain, Road, Grandmother, Hospital, Black, Curtains, live, Car, Mechanic, Work, Dictation, EidMubarak, You, Name	✓	✓

Table 9: Tested sentence validation

No.	Sentence	Compared PSL video	PSL expert
1	I go to school	✓	✓
2	He sits on chair	✓	✓
3	He was happy before rain	✓	✓
4	Babar was going on road	✓	✓
5	His grandmother was in hospital	✓	✓
6	He has black curtains	✓	✓
7	This car mechanic lives in DeraIsmailKhan	✓	✓
8	Sit and work on dictation	✓	✓
9	Eid Mubarak to You	✓	✓
10	A crow was very thirsty	✓	✓
11	He searches for water here and there	✓	✓
12	Atlast, he saw pitcher of water in garden	✓	✓
13	The water was very low	✓	✓
14	His beak could not reach water	✓	✓
15	He saw pebbles in garden	X	X (Ambiguous)
16	He pebbles in pitcher	✓	✓
17	The water riseup	✓	✓
18	He drinks water and flies away happily	X	X (Ambiguous)
19	What is your name?	✓	✓
20	My name is Babar	✓	✓

The vocabulary of PSL signs is coded in HamNoSys symbolic notation and then converted to SiGML form. Sign4PSL has all the basic signs for alphabets, digits, and vocabulary used in daily life. For those nouns and words that are not found in the knowledge base Sign4PSL perform fingerspells. Using the context vocabulary, Sign4PSL plays sentences. Sign4PSL does not deal with past sentences; it only shows the ‘was’ sign whenever there is a talk about the past sentence.

Table 10: Detected ambiguity in sentences

No.	Problematic word	Tested sentence	PSL video	PSL expert
1	Right	You are right	✓	✓
		It is on right side	X	X
2	Fly	A fly was sitting on window	X	X
		He flies away happily.	✓	✓
3	Saw	He saw water	✓	✓
		This saw is very sharp	X	X
4	Ship	He ships on time	✓	✓
		I saw a ship on water	X	X
5	I	I am Alina I: subject (Representing Self) i: Alphabet	X	X

Results show that twenty sentences played accurately; however, five created ambiguities due to the English language. The dual meaning led Avatar to play the wrong animation. Therefore, a total of 80% of the sentences were processed accurately, as shown in [Tab. 9](#). The accuracy of the rest of the sentences was compromised because of their complex structure. These issues were because of the incompatibility of structured and unstructured behavior of the languages ‘English’ and ‘PSL’.

Because of English language rules and the dual meaning of the words, sentence translation seemed difficult. However, an effort is made for the processing of these sentences. Complex sentences show less accurate results as compared to the basic sentences following S-V-O order, but results for sentences were satisfactory with the help of NLP. The Sign4PSL is tested by inputting 25 sentences, and Avatar played all well except a few, which created ambiguity. All other words, letters, and short phrases were 100% played correctly except some of the long ambiguous phrases.

Signs played by 3D Avatar are checked by language experts and teachers of PSL in special education school of Pakistan. Results were encouraging for the use of Sign4PSL. Sign4PSL can be used in different scenarios and can work well for all PSL content by adding vocabulary. [Tab. 10](#) & [11](#) shows some of the words tested with their HamNoSys and their signs, and the Avatar played most of them correctly except for a few ambiguous sentences shown in [Tab. 10](#). Sign4PSL is also checked by playing ‘Thirsty Crow’ story in PSL that was showed to deaf children in special education school. In this way, Sign4PSL can also be used in special schools for learning deaf children. In Pakistan, because of the lack of facilities, teaching deaf children is difficult. Sign4PSL can help them to teach PSL to deaf children. It is a web-based solution with a mobile interface and can be accessed from any device with the internet. Sign4PSL needs a minimum of 2 Mbps bandwidth for processing the requests.

Conversion of PSL gestures to HamNoSys notation and then to SiGML opens new doors for PSL. Because HamNoSys is designed to be used internationally, the deaf community in Pakistan can understand Pakistani content in PSL and International content in English, connecting them to the world. In this way, they can understand the news, interact with social media, and many more.

Table 12: Sentence with animations



Table 13: Time measures for phases

Phrases	1	2	3	Avg.	Comment
Initialization loading	485 ms	547 ms	498 ms	510 ms	Only required once
Initialization loading (caching)	51 ms	43 ms	75 ms	56 ms	Only required once
Lexical analysis	17 ms	15 ms	19 ms	17 ms	Runs N-Time on N-Sign: $N \geq 1$
Text preprocessing	3.4 ms	5.3	4.2	4.3	Runs on each input
Stemming	53 ms	53 ms	37 ms	47 ms	Runs N-Time on N-Signs: $N \geq 1$
Signs dumping (PHP)	27 s	28 s	40 s	31 s	Runs once on start
Signs dumping (JSON)	49 ms	38 ms	43 ms	43 ms	Only runs on start
Signs finding	4 ms	6 ms	5 ms	5 ms	Runs N-time on N-signs $N \geq 1$
Animation player (Exclusive of signs animation)	15 s	14 s	26 s	15 s	Runs N-time for N-Signs: $N \geq 1$
Avatar loading	183 ms	182 ms	184 ms	183 ms	Runs once on start
Skin shader loading	358 ms	360 ms	502 ms	406 ms	Runs once on start

6 Conclusions

Deaf people learn sign language as their first language for communication. They prefer information to be displayed in their respective sign language. The creator of PSL is the deaf people of Pakistan, and some signs are also borrowed from other languages. Limited research is done on PSL, and there is no proper grammar available for sign language. Moreover, there is no concept of inflexions in PSL; instead, it uses the root form of words. As with other sign languages, the PSL dictionary is also limited. In PSL, there are no concept of helping verbs, conjunctions, and articles; instead, these are all ignored.

In this paper, an architecture is presented for English text to PSL translation using HamNoSys. In this technique, HamNoSys is generated for PSL alphabets, digits, words, and sentences, further converted to XML form known as SiGML. Sentences are the combination of these words. Signs in SiGML form are converted and already stored to help Sign4PSL in playing corresponding signs. Also, a module has been implemented to translate HamNoSys to SiGML conversion for the convenience of Sign4PSL's administrator. This will help in the addition and correction of more PSL signs in future. The results have shown that the architecture achieves 100% accuracy on alphabets, digits, words, and phrases. However, for sentences, the architecture achieves 80% accuracy.

7 Future Work

Sign4PSL works for manual features of PSL by combining all the tools. It takes input in the form of text and plays corresponding signs. In Sign4PSL, HamNoSys is provided only for basic hand movements. It can further be extended as Sign4PSL supports other complex signs and non-manual features of sign language. For this, non-manual features of PSL can be identified and added to the Sign4PSL. More signs can also be added in the future. Generation of HamNoSys is a difficult task. HamNoSys generator can be made for naive person to work with system and generate signs for PSL easily. However, the limitation of grammar for good PSL sentence's structure lacks. In future, the system can also be extended to support past, present, and future tenses.

Acknowledgement: This research is ongoing research supported by Yayasan Universiti Teknologi PETRONAS Grant Scheme; 015LC0029 and 015LC0277.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] A. Dhanjee, "A sign language accessibility for deaf people in Pakistan," *The PSL Interpretation Training Program, Youth Exchange and Study*, Karachi, Pakistan, pp. 1, 2018.
- [2] W. H. Organization, "Deafness and hearing loss," *World Health Organization*. [Online] Available: <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss> (1 April 2021).
- [3] M. L. Hall, V. S. Ferreira and R. I. Mayberry, "Syntactic priming in American sign language," *PLOS ONE*, vol. 10, no. 3, pp. 1–19, 2015.
- [4] T. Hanke, "Hamnosys-representing sign language data in language resources and language processing contexts" in *4th Int. Conf. on Language Resources and Evaluation*, Lisbon, Portugal, pp. 1–6, 2004.

- [5] J. Hutchinson, "Literature review: analysis of sign language notations for parsing in machine translation of SASL," Ph.D. Dissertation, Rhodes University, South Africa, 2012.
- [6] S. Prillwitz, R. Leven, H. Zienert, T. Hanke and J. Henning, "Hamburg notation system for sign languages: An introductory guide, *HamNoSys Version 2.0*, Signum, Seedorf, Germany, 1989.
- [7] R. Elliott, J. Glauert, V. Jennings and J. Kennaway, "An overview of the SiGML notation and SiGML signing software system," in *Proc. 4th Int. Conf. on Language Resources and Evaluation*, Lisbon, Portugal, pp. 98–104, 2004.
- [8] M. Punchimudiyanse and R. Meegama, "3D signing avatar for sinhala sign language" in *Proc. IEEE 10th Int. Conf. on Industrial and Information Systems*, Peradeniya, Sri Lanka, pp. 290–295, 2015.
- [9] Y. Bouzid and M. Jemni, "An avatar-based approach for automatically interpreting a sign language notation," in *Proc. IEEE 13th Int. Conf. on Advanced Learning Technologies*, Beijing, China, pp. 92–94, 2013.
- [10] A. Othman and M. Jemni, "An XML-gloss annotation system for sign language processing," in *Proc. 6th Int. Conf. on Information and Communication Technology and Accessibility*, Muscat, Oman, pp.1–7, 2017.
- [11] O. H. Al-Barahamtoshy and H. M. Al-Barhamtoshy, "Arabic text-to-sign (ArTTS) model from automatic SR system," *Proc. Computer Science*, vol. 117, pp. 304–311, 2017.
- [12] K. Kaur and P. Kumar, "Hamnosys to SiGML conversion system for sign language automation," *Proc. Computer Science*, vol. 89, pp. 794–803, 2016.
- [13] D. A. Gonçalves, E. Todt and L. Sanchez Garcia, "3D avatar for automatic synthesis of signs for the sign languages," in *Proc. Conf. on Computer Graphics, Visualization and Computer Vision*, Plzen, Czech Republic, pp. 17–23, 2015.
- [14] H. Luqman and S. A. Mahmoud, "Automatic translation of arabic text to arabic sign language," *Universal Access in the Information Society*, vol. 18, no. 4, pp. 939–951, 2019.
- [15] Q. L. Da, N. H. D. Khang and N. C. Ngon, "Converting the Vietnamese television news into 3D sign language animations for the deaf," In: T. Duong, NS. Vo (eds.) *Industrial Networks and Intelligent Systems, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 257, pp. 155–163, 2018.
- [16] M. Brour and A. Benabbou, "ATLASLang MTS 1: Arabic text language into arabic sign language machine translation system," *Proc. Computer Science*, vol. 148, pp. 236–245, 2019.
- [17] P. R. D. Team, "Pakistan sign language," *Family Education Services Foundation (FESF)*, 2020. [Online]. Available: <https://www.psl.org.pk>.