Design of translation software with multi-inputs to translate English to Hindi OR vice-versa

Ankit Kumar

Department Of Computer Science and Engineering

Apex institute Of Technology Chandigarh University Mohali-140413, Punjab

20BCS6801@cuchd.in

Aniket Mittal

Department Of Computer Science and Engineering

Apex institute Of Technology Chandigarh University Mohali-140413, Punjab

20BCS6819@cuchd.in

Akshit Jindal

Department Of Computer Science and Engineering

Apex institute Of Technology Chandigarh University Mohali-140413, Punjab

20BCS6728@cuchd.in

Bipandeep Singh Pannu

Department of Computer Science and Engineering

Apex institute of Technology Chandigarh University Mohali-140413, Punjab

20BCS6809@cuchd.in

Lata Gupta

AIT-CSE

Apex institute of Technology Chandigarh University Mohali-14041 Punjab

Lata.E13365@cumail.in

Dhawan Singh

AIT-CSE

University Centre for Research and Development

Chandigarh University Mohali-140413, Punjab

dhawan_deor@yahoo.co.in

Abstract—**In this paper, an extended combined approach of phrase-based statistical machine translation (SMT), example-based MT (EBMT), and rule-based MT (RBMT) to develop a novel hybrid data-driven MT system. This hybrid MT process is guided by the rule-based MT after receiving partial candidate translations from the EBMT and SMT subsystems. Previous works have shown that EBMT systems outperform phrase-based SMT systems, and RBMT generates more accurate results in terms of structure and morphology. The proposed hybrid approach improves the fluency, accuracy, and grammatical precision of the machine translation system. A comparison with renowned translators such as Google, BING, and Babylonian demonstrate that the proposed model performs better on sentences with ambiguity and idioms.**

Keywords—machine translation, Hindi-English, machine translation, example Based Machine translation, statistical based machine translation, ruled based machine translation, hybrid machine translation,translation quality analysis

I. INTRODUCTION

Machine translation is the name for computerized methods that automate all or part of the process of translating from one language to another.Cross-language communication is essential for fostering positive relationships and promoting mutual benefits between two countries. In today's digital age, machine translation serves as a valuable tool for facilitating this communication. Numerous countries have dedicated significant resources to the advancement of practical machine translation technologies [7]. Hindi Written in the Devanagari script is the official language of the union Government.Over the past few decades, many different methods and resources have been used to develop machine translation systems for various applications. These systems range from simple text translation to multilingual speech systems. Most machine translation systems assume that sentences will be grammatically correct and complete. India's average literacy level is 70 percent. [1]. Then such sentences are translated to target language with preserving the meaning in the source language. After Mandarin, Spanish and English, Hindi is the most negatively spoken language in the world, almost spoken by 260 million people according to Ethnologist, 2014 [4]. Yes, there is definitely a need for translators who can translate sentences from Hindi to other languages. In this case, English is the target language. It seems that most of the machine translation efforts have been focused on developing translation systems from English to Indian languages [5]. There have been some systems developed for translating Hindi to English, but they are not yet advanced enough to handle all the complexities and uncertainties of Hindi sentences.Many organizations like Google, Microsoft, IBM, and Many other etc. Are engaged in development of MT systems.[3]

There are various approaches that have been developed to overcome the limitations of automated machine translation, such as Statistical Machine Translation (SMT). SMT goes beyond simple word substitution and takes into account the semantic and syntactic constraints of the target language in order to deliver more accurate translations, EBMT [8] and RBMT [9]. These approaches have their own strengths and weaknesses.

There are already existing many freely available Hindi-English machine translation systems like Google Translator, MS-Sing and Babylon. These systems are developed based on different approaches i.e., Rule Based Machine Translation (RBMT), Example Based Machine Translation (EBMT), Sta- tactical Machine Translation (SMT) [10]. However, none of them are proficient at accurately handling word sense disambiguation, pronoun resolution, and idiomatic translations.

RBMT systems perform translation by applying linguistic rules that dictate how words, phrases, and structures in the source language should be transformed into the target language. On the other hand, EBMT and SMT systems automatically extract these rules from parallel corpora, eliminating the need for manual development. These data-driven approaches have gained popularity due to their ability to generate translations based on large amounts of available data.[12].

In this research work, model is deigned using new technology,which is new used in any of translator model like Google translator and all.The technology used in making model is Django framework. Presented in section 2. We have discussed, what are model available before design this model in section 3. In section 4, we elaborated method and approach which we used in making translator model. Working and detailed explain of implementation part are written in section 5. Final analysis of model are mentioned including running code of working of model in section 6.Finally, concluding remarks and future direction including best approaches, how to implement with best method are given in section 7.

II. LITERATURE REVIEW

Automatic translation or machine translation (MT) is a field of computational linguistics that aims to automate the process of translating from one source language to another target language [11,12]. There are several approaches to machine translation, including Rule Based MT, Direct Based MT, Corpus Based MT, and Knowledge Based MT[8].In this paper, several Hindi-English translation divergences have been studied in order to identify language specific divergences and further to be incorporated in EBMT and RBMT phases during the translation. In terms of configuration characteristics, English is more rigid and restrictive that follows fixed word order patterns as opposed to Hindi. For instance, one of the translation divergences related to specific word-order pattern is the interpretation of a Hindi question particle "§या" [15]."§या' can be used both as a question particle in yes/no type of question sentences and as a type of interrogative pronoun in content question sentences as shown in example 1.

Example 1:

1. "§या आप tलख रहे ह¢?" ⇒ Are you writing?

2. "आप §या tलख रहे ह¢?" ⇒ What are you writing?

Example 2:

1. "वह चलते चलते थक गया।" ⇒He got tired of walking.

A. Rule Based MT (RBMT)

The RBMT system takes the source text and breaks it down into an intermediate representation using specific rules for grammar, vocabulary, and morphology. This intermediate representation is then used to generate the target languages. RBMT can be done using either Transfer Based MT or Interlingual MT.

B. Direct Based MT (DBMT)

Direct translation, also known as word-for-word translation, is a method that translates each individual word in a sentence from one language to another using a dictionary. The motivation behind the project English to Hindi machine translation system is that many documents and records like government records, historical, news etc. are written in English, which is not popular among the remote villagers of India.

C. Corpus Based MT (CBMT)

CBMT relies on the analysis of bilingual text corpora. Statistical MT (SMT) and Example Based MT are both considered Corpus Based MT methods. SMT is particularly useful for handling exceptions to rules when translating from English to Hindi. One of the advantages of SMT is that it does not require extensive linguistic knowledge for translation. However, the creation of a parallel corpus can be a challenging step in implementing an SMT system.

D. Knowledge Based MT (KBMT)

Knowledge-based machine translation (KBMT) is a machine translation method that uses external data sources to improve the translation process. KBMT outperforms traditional methods based only on statistical models or neural network training. Instead, it combines explicit knowledge about the world, domain-specific knowledge or language knowledge to improve translation quality. A domain-specific Knowledge Base (KB) is utilized in KBMT for translation purposes. The creation of the KB is dependent upon ontology and the semantic web.

E. Hybrid Based MT (HBMT)

Define HBMT is the combinations of two or more MT techniques.Hybrid-based machine translation (HBMT) refers to an approach that combines different machine translation methods or techniques to improve the overall quality of translations. Machine translation involves the use of software to automatically translate text or speech from one language to another. HBMT integrates different translation methods, typically combining rule-based, statistical, and neural machine translation methods to exploit the strengths of each.

III. EXISTING SYSTEMS

There has been significant development of machine translation systems in India. Research on machine translation began in the mid-1980s and 1990s. Some notable English to Hindi machine translation systems have been developed as a result of these efforts.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

A. MANTRA-MT system

The Machine Assisted Translation Tool for (MANTRA) is a system that translates English to Hindi. It can be used in various domains such as administration, office orders, and circulars. MANTRA utilizes Tree Adjoining Grammar for generation and parsing. Additionally, the system employs Transfer Based MT for translation. [13]

B. MaTra-MT system

MaTra, is a human-aided transfer-based system that is specifically designed for translating English to Hindi. This system can be effectively used in various domains such as news and technical phrases. MaTra utilizes Transfer Based Machine Translation to accomplish its translation tasks. [14]

C. UNL Based English to Hindi MT system

That's interesting! It's impressive that the Indian Institute of Technology is developing a machine translation system using the Universal Networking Language (UNL). The machine translation system using the Universal.Networking Language (UNL) that is Interlingua is developed by Indian Institute of Technology. This system aims to invent an Interlingua for all daily used human languages. [15]

D. Hinglish Machine Translation system

The Hinglish System, developed by Sinha and Thakur, is primarily designed for translating pure Hindi to pure English forms. It serves as an additional level to the existing English to Hindi translation system, (AnglaBharti-II).

E. AnglaBharti

The English Angla Bharti is a pseudo-interlanguage and rule-based translation It was further expanded as Anglo Hindi. English machine translation refers to the development of machine translation systems specifically designed to translate between Hindi and English or both, known as English. English is a common spoken language of Hindi and English in India, where speakers perfectly combine elements of both languages in their communication.Hinglish contains a unique linguistic phenomenon called code-switching, where speakers seamlessly switch between Hindi and English within the same sentence or conversation. A robust English machine translation system must manipulate code efficiently to produce accurate and natural translations.

F. Google Translator

The English Google Translate translates written text from one language to another It supports 90 languages. Limitations of Google Translate the number of paragraphs and technical terms to be translated.Some nouns, such as personal labels, still remain during translation. It allows users to translate text, documents, websites and even spoken words in multiple languages. Google Translate supports many languages that cover a wide range of linguistic diversity. Users can translate texts or documents between different languages such as English, Spanish, French, German, Chinese, Japanese and many more.Hindi and English have distinct grammatical structures and syntax. A good English Machine Translation system should be capable of capturing and translating the grammar and syntax of both languages appropriately to ensure coherent and meaningful translations.

TABLE II. SEVEN DECLENSION RULES

No	Significance	Case-Form
1	The naming case Denotes the subject The/ any proper nouns	The/ any proper nouns
2	Denotes the Object The	The
3	Denotes the agent. Denotes the instrument. Denotes the means	With, by, along, by means of
4	Indicates the direction in which the action denoted by verb takes place. Indicates the purpose	For, to
5	Denotes Separation. Denotes Source. Denotes Motive	From, out of
6	Denotes possession.	‘s, of, belonging to
7	Denotes the place or the situation of a thing	In, into, on, over, among, between, ,n the midst of

IV. PROPOSED SYSTEM ARCHITECTURE

After the We offer an HMT architecture with a combination Declination Ruled Based MT and SMT systems. erasers, or *vibhakti* are declension forms of nouns. The system architecture is described below. The main motive is to explore the creation of English subtitles and convert it to Hindi using a machine translation system viz. A neural machine translation system and the Google translation system. Still grounded different BLEU points analyzing the existing level Result of machine translation. In addition, natural language must define semantic information i.e develop visual language models understanding Each architectural sub module is called an important role.

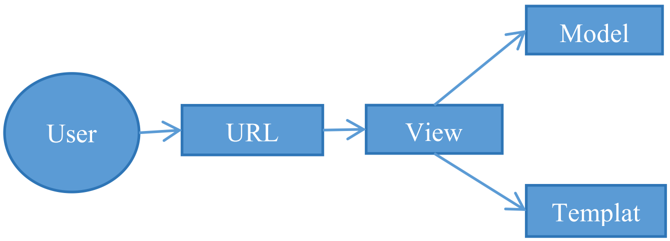


Fig. 1. System Architecture

The architecture components are explained below.

A. User:

Regarding web development “users” refer to individuals interacting with a web application. Users can access and interact with the application through a web browser or other client software. User experience and user interface design are important aspects of web development so that users can easily navigate and use the application.

B. Model:

The “model” in mvc refers to the application data and business logic. It represents the structure and storage of data and the rules and calculations performed on that data. In a web application, the model is responsible for retrieving and processing data and implementing business rules that govern data processing.

C. View

The “View” is responsible for presenting information to the user and handling the user interface. It represents the visual and interactive elements that users interact with. In a web application, a view generates HTML, CSS, and eventually JavaScript that is rendered in the user's browser. The view is separate from the model, which provides a clear separation of concerns, allowing for easier maintenance and flexibility in design.

D. Templates:

Templates are a special concept related to viewing in web development. Templates are used to dynamically structure data presentation. They often contain placeholders for data, and the model engine fills those placeholders with the actual data in the model. Templates help maintain a consistent layout and style across multiple web application.

E. *URL:*

URL is an email address that identifies the location of a resource on the Internet. In web development, URLs play a vital role in routing and navigation. They define the application's structure and routes, and determine how different URLs correspond to different views or controls. Proper URL design is essential to create user-friendly and intuitive navigation in a web application.

F. *Pos taggers:*

Parts of Speech (POS) tagging for the parsed input source sentence can also be done using Stanford POS tagger. Following is the tag set used by Stanford parser.

TABLE III. POS TAG SET

S.No	**Abbreviations**	**Type**
1	CC	Coordinating conjunction
2	CD	Cardinal number
3	DT	Determiner
4	EX	Existential
5	FW	Foreign word
6	IN	Preposition
7	JJ	Adjective
8	JJR	Adjective, comparative
9	JJS	Adjective, superlative
10	LS	List item marker
11	MD	Modal
12	NN	Noun, singular or mass

G. Sample GUI screen for the system:

There are multiple languages included in this model to translate as per users requirements. Users can also change the languages from Hindi to English or vice-versa. Multiple languages are given there, users can can switch to any languages at runtime, while code is running. It's worth noting that advancements in the field of machine translation continue to evolve, and new developments may have occurred since my last update.

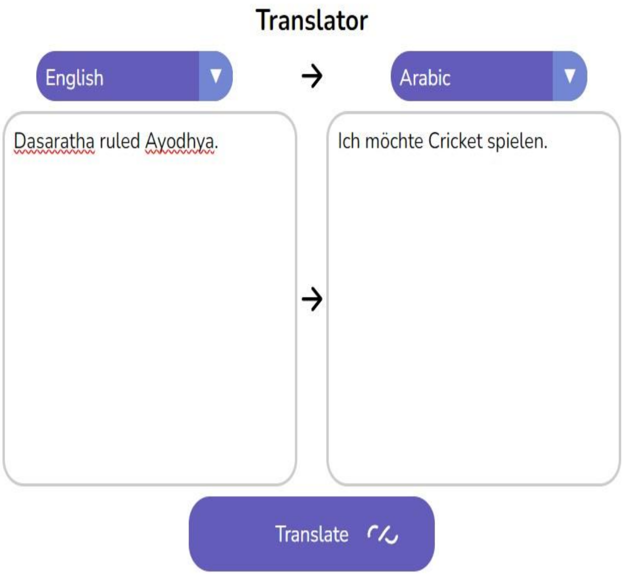


Fig. 2-Sample GUI for the Existing system.

V. IMPLEMENTATION

Methodologies

Python virtual environment is a self-contained directory that encapsulates a specific Python interpreter and its associated libraries and packages. It allows you to create isolated environments for different Python projects, ensuring that the dependencies of one project do not interfere with those of another. Virtual environments are a crucial tool for managing Python packages, especially when you work on multiple projects with different sets of requirement.

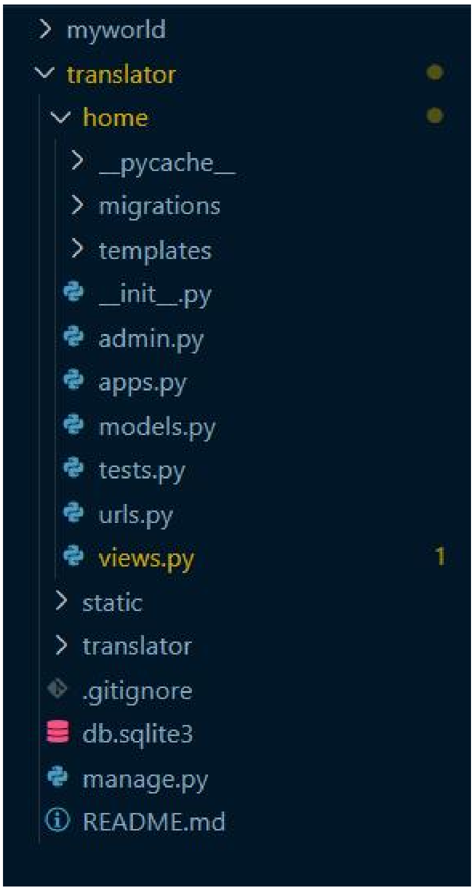
Python -m pip install --user virtual

```
# Check the version of virtual environment to ensure it is installed properly
Python -m virtual –version
# Now create a virtual environment named “my world”
Python -m virtual my-world
# To activate newly created environment, use the following command
source my world/Scripts/activate
```

After the virtual environment is created successfully, now we can install various dependencies and packages required for the project:

Django app: is a modular component of a Django web project. In Django, a project is the entire web application, while an app is a self-contained module within the project that serves a specific purpose or functionality. Apps are used to organize and structure your project, making it easier to manage and maintain.

Python manage.py start app home

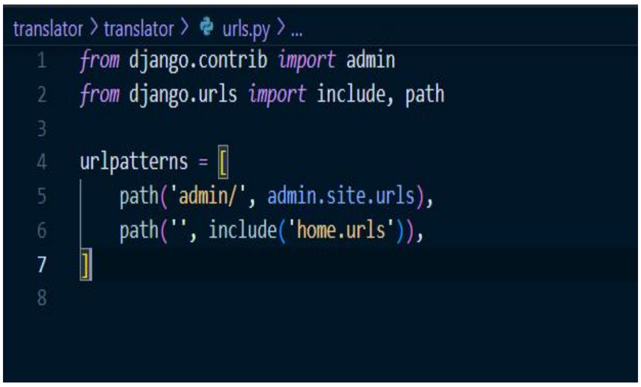


```
> myworld
v translator
  v home
    > __pycache__
    > migrations
    > templates
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py                      1
  > static
  > translator
  .gitignore
  db.sqlite3
  manage.py
  README.md
```

A Django application consists of the following components:

- URL dispatcher
- View
- Model
- Template

The main URLS.PY file in project folder looks like:



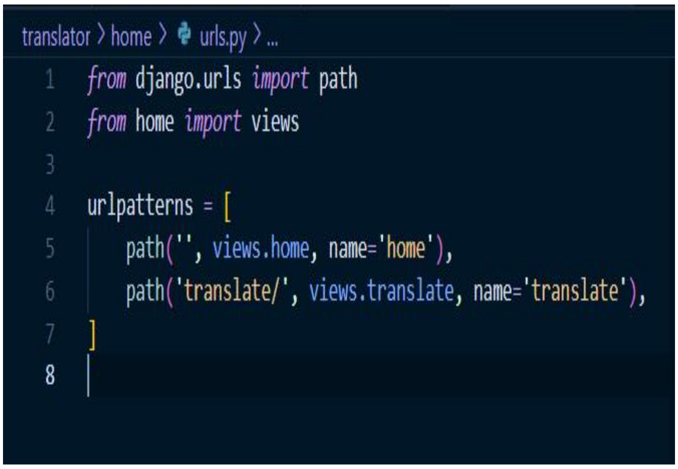
translator > translator > urls.py > ...

```python
from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('home.urls')),
]
```

G. URL dispatcher

Django's URL dispatcher mechanism is equivalent to the controller in the MVC architecture. The URLS.**PY** module in the Django project's package folder acts as the dispatcher. It defines the URL patterns. Each URL pattern is mapped with a view function to be invoked when the client's request URL is found to be matching with it. The URL patterns defined in each app under the project are also included. Here's the URLS.**PY** file in the app folder.



translator > home > urls.py > ...

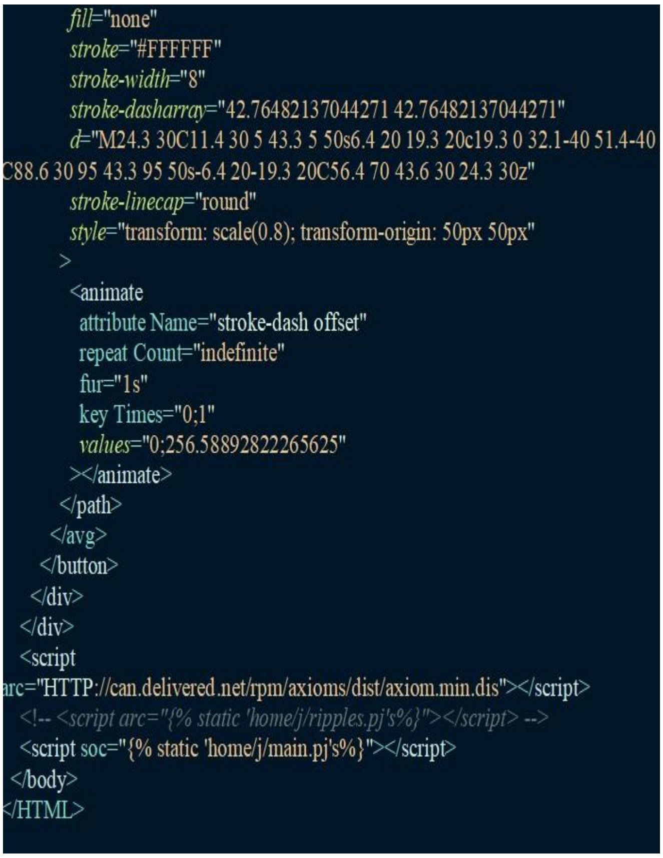
```python
from django.urls import path
from home import views

urlpatterns = [
    path('', views.home, name='home'),
    path('translate/', views.translate, name='translate'),
]
```

H. View:

The view function reads the path, query, and body parameters included in the client's request If required, it uses this data to interact with the models to perform CRUD operations. A view can be a user-defined function or a class. You create View definitions in the **views.PY** file of the respective app package folder.

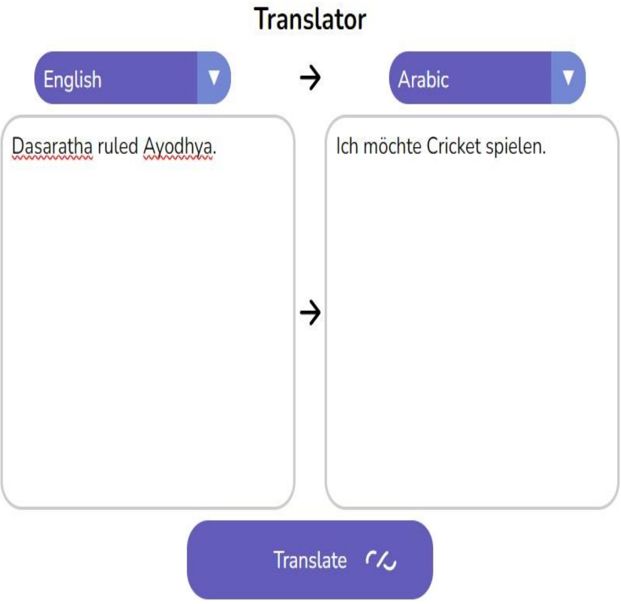
This file contains the front end part for our project, it provides the user interface and passes the input to the java script file which translates the input via axiom library through API we created in views.PY file using Django-rest-framework.



```html
        fill="none"
        stroke="#FFFFFF"
        stroke-width="8"
        stroke-dasharray="42.76482137044271 42.76482137044271"
        d="M24.3 30C11.4 30 5 43.3 5 50s6.4 20 19.3 20c19.3 0 32.1-40 51.4-40
C88.6 30 95 43.3 95 50s-6.4 20-19.3 20C56.4 70 43.6 30 24.3 30z"
        stroke-linecap="round"
        style="transform: scale(0.8); transform-origin: 50px 50px"
       >
        <animate
         attribute Name="stroke-dash offset"
         repeat Count="indefinite"
         fur="1s"
         key Times="0;1"
         values="0;256.58892822265625"
        ></animate>
       </path>
      </avg>
     </button>
    </div>
   </div>
   <script
arc="HTTP://can.delivered.net/rpm/axioms/dist/axiom.min.dis"></script>
   <!-- <script arc="{% static 'home/j/ripples.pj's%}"></script> -->
   <script soc="{% static 'home/j/main.pj's%}"></script>
  </body>
</HTML>
```

VI. OUTPUT ANALYSIS

We compared our system with Google translator. In Google translator, Hindi language has not got its perfection in terms of grammar and semantic rules. The accuracy of Arabic sentence which is the output for corresponding English sentence in the below Fig. 1 is not in accordance with the fact. The given sentence is in English, and it is converted into Arabic through the translator.


English → Arabic

Dasaratha ruled Ayodhya.

Ich möchte Cricket spielen.

Fig. 3: Output

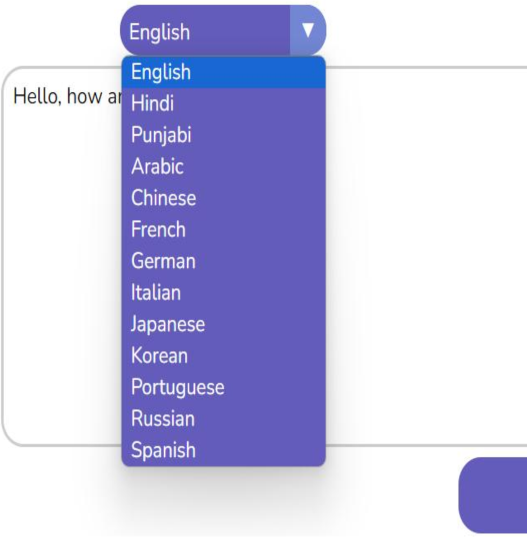

Hello, how ar

Fig. 4. Multiple Languages

TABLE IV.OUTPUT COMPARISON

Case-ending name	**Significance**	**English**	**Hindi**
Nominative	The naming case denotes the subject		ने
Accusative	Denotes the object		को
Instrumental	Denotes the agent. Denotes the instrument	With, by, along, by means of	सेके साथ, के द्वार
Dative	Indicates the direction in which the action denoted by verb takes place. Indicates the purpose	For, to	के लिए, को
Ablative	Denotes separation. Denotes source. Denotes motive.	From, out of	से
Genitive	Denotes possession.	Of, belonging to	का, के, की
Locative	Denotes the place or the situation of a thing	In, into, on, over, among, between, in	में, पर

VII. CONCLUSION

In this article, we have presented an effective method English to Hindi based on the MT declension. We have proposed a new approach to MT system design that has not been done were taken into account in all existing MT systems. in new rules to improve the translation process can be added to the system.This technique can be left out better efficiency compared to other approaches English to Hindi. For example, a Google result translator, while the translation is that the Hindi was not received its perfection according to grammatical rules. So we suggested system design shows accurate results than other systems. The system design proposed here is only once in the initial state successfully executed, it can be evaluated using BLUE points and other techniques. The future of Hindi-English MT is bright. With continued research and development, MT is likely to become increasingly accurate, robust, efficient, accessible, and user-friendly. This will make MT even more useful for a wider range of applications, such as news translation, business translation, government translation, education, and entertainment.It's important to note that the field of machine translation is dynamic, and new developments may have occurred since my last update.

REFERENCES

[1] F. Ren and H. Shi, “Parallel machine translation: principles and prac- tice,” in *Engineering of Complex Computer Systems, 2001. Proceedings. Seventh IEEE International Conference on*. IEEE, 2001, pp. 249–259.

[2] O. Abiola, A. Adetunmbi, and A. Oguntimilehin, “A review of the various approaches for text to text machine translations,” *International Journal of Computer Applications*, vol. 120, no. 18, 2015.

[3] U. Tiwary and T. Siddiqui, *Natural language processing and information retrieval*. Oxford University Press, Inc., 2008.

[4] Ethnologue, “Most Widely Spoken Languages in the World,” http: //www.infoplease.com/ipa/A0775272.html, 2014, [Online; accessed 24- June-2016].

[5] P. K. Goswami and S. K. Dwivedi, “An empirical study on english to hindi e-contents machine translation through multi engines,” in *Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), 2014 3rd International Conference on*. IEEE, 2014, pp. 1–6.

[6] N. Sharma, “English to hindi statistical machine translation system,” Ph.D. dissertation, THAPAR UNIVERSITY PATIALA, 2011.

[7] Latha R. Nair David Peter S, Machine translation systems for Indian language, IJCA Journal 2012, Volume 39 - Number 1

[7] H. Somers, “Review article: Example-based machine translation,” *Ma- chine Translation*, vol. 14, no. 2, pp. 113–157, 1999.

[8] J. Centelles and M. R. Costa-Jussa, “Chinese-to-spanish rule-based machine translation system,” 2014.

[9] D. Groves and A. Way, “Hybrid data-driven models of machine trans- lation,” *Machine Translation*, vol. 19, no. 3-4, pp. 301–323, 2005.

[10] B. Nithya and S. Joseph, “A hybrid approach to english to malayalam machine translation,” *International Journal of Computer Applications*, vol. 81, no. 8, 2013.

[11] P. Dungarwal, R. Chatterjee, A. Mishra, A. Kunchukuttan, R. Shah, and P. Bhattacharyya, “The iit bombay hindi-english translation system at wmt 2014,” *ACL 2014*, p. 90, 2014.

[12] B. J. Dorr, “Machine translation divergences: A formal description and proposed solution,” *Computational Linguistics*, vol. 20, no. 4, pp. 597– 633, 1994.

[13] R. Sinha and A. Thakur, “Disambiguation of hindi to english machine translation,” in *Sixth International Conference of South Asian Languages (ICOSAL-6)*, 2005.

[14] D. Marcu, “Towards a unified approach to memory-and statistical-based machine translation,” in *Proceedings of the 39th annual meeting on as- sociation for computational linguistics*. Association for Computational Linguistics, 2001, pp. 386–393.

[15] P. Desai, A. Sangodkar, and O. P. Damani, “A domain-restricted, rule based, english-hindi machine translation system based on dependency parsing,” in *11th International Conference on Natural Language Pro- cessing, ICON*, 2014.

[16] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer, “The mathematics of statistical machine translation: Parameter estimation,” *Computational linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[17] K. Chunyu, P. Haihua, and J. J. Webster, “Example-based machine translation: A new paradigm,” *Translation and information technology*, p. 57, 2002.

[18] M. Rana and M. Atique, “Example based machine translation using fuzzy logic from english to hindi,” in *Proceedings on the International Conference on Artificial Intelligence (ICAI)*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2015, p. 354.

[19] M. Paul, T. Doi, Y.-S. Hwang, K. Imamura, H. Okuma, and E. Sumita, “Nobody is perfect: Atr’s hybrid approach to spoken language transla- tion.” in *IWSLT*, 2005, pp. 45–52.

[20] O. Bojar, V. Diatka, P. Rychlỳ, P. Stranák, V. Suchomel, A. Tamchyna, and D. Zeman, “Hindencorp-hindi-english and hindi-only corpus for machine translation.” in *LREC*, 2014, pp. 3550–3555.