TRANSLATING MULTI-LINGUAL TEXT INTO A UNIFORM LANGUAGE "ENGLISH"

PROJECT REPORT SUBMITTED IN FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY AND ENGINEERING

UNDER THE SUPERVISION OF

DR. RAJNI MOHANA

BY

ISHITA AGARWAL – 111406

TO

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that project report entitled “TRANSLATION OF A MULTILINGUAL TEXT INTO A UNIFORM LANGUAGE ENGLISH”, submitted by ISHITA AGARWAL in fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Vaknaghat , Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date: **DR . RAJNI MOHANA**

16-12-2014 **Asst. Professor(senior grade)**

ACKNOWLEDGEMENT

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to DR. RAJNI MOHANA for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards my parents & my friends for their kind co-operation and encouragement which help me in completion of this project.

16.12.2014 ISHITA AGARWAL

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This project creates a web based application that translates multi lingual text into a uniform language English and it also has a feature of text to speech conversion. Multilingualism is the use of two or more languages, either by an individual speaker or by a community of speakers. Multiple languages is becoming increasingly frequent, thereby promoting a need to acquire additional languages. Multilingual communication is an increasingly important social phenomena. Communication between individuals in multiple languages.

This project has been made for the convenience of the web users. It identifies the language in which the text is written and translates it into English. It speaks the translated text. My project also shows the translated text in 5 different languages and we have applied an algorithm which converts SMS language into English. We have also tried to put the feature of speech to text conversion.

Both the translation and identification part can be done very easily. The interface has been made user friendly.

CHAPTER 1

ABOUT THE PROJECT

1.1 INTRODUCTION

Multiple languages is becoming increasingly frequent, thereby promoting a need to acquire additional languages. Multilingual communication is an increasingly important social phenomena. Communication between individuals in multiple languages.

A language translator translates the given text into the English language. It auto-identifies the language in which the text is written and then converts it into English .Various translators already exists , but in order to explore it’s working process ,I thought of taking this as my project.

My project also shows the translated text in 5 languages and if the text of 2 languages are written together ,then it can show it’s translated version. After translation, identification is certain ,the text to speech conversions and speech to text conversions happens in my translator. We have also tried to develop an algorithm to convert SMS language into English. Both the translation and identification part can be done very easily. The interface has been made user friendly.

1.2 ADVANTAGES

- Ensure data accuracy
- It shows the translated text in 5 languages
- Removes few shortcomings of other translators.
- Greater efficiency ,User friendly and interactive
- Less time consuming
- If texts of 2 languages are written together, then it can show the translated version of that also.
- Very simple and easy to implement

1.3 OVERALL DESCRIPTION

GOALS OF THE PROPOSED SYSTEM:

Accuracy: The level of accuracy in the translator will be higher. All operations would conform to integrity constraints and correctness and it will be ensured that whatever id is received at or sent from/by the api is accurate.

Reliability: The reliability of the translator will be high due to the above mentioned reasons.

No redundancy: In the translator it will be ensured that no repetition of information occurs ; neither on a physical storage nor on logical implementation level. This economizes on resource utilization.

Immediate retrieval of information: The main objective of the translator is to provide a quick and efficient platform for retrieval of information .i.e. translating the text and identifying the language in which it is written.

Ease of operation: The system should be simplistic in design and use.

1.4 PROJECT REQUIREMENT

Software requirements:

- Operating System: Windows 7/8/XP/Vista
- Front End &Rear end: .net,java script,html.
- Platform required: Visual studio 2013
- Design Tool: smartdraw

Hardware requirements:

- Processor: x86 compatible processor
- Hard Disk: 20 GB or greater
- RAM: 512 MB or greater
- Monitor: VGA/SVGA
- Keyboard: 104 keys standard
- Mouse: 2/3 button. Optical/ Mechanical.

1.5 CONSTRAINTS

- This translator does not convert two different languages written together in a single text box into one.

- This translator needs more advanced algorithm implementation for conversion of SMS language into English.

  This translator should convert text from any language to English without the help Of BING API.

CHAPTER 2

LITRATURE REVIEW

1. [*Rodney P- Hamann ,**MULTI-LINGUAL DATA PROCESSING OTHER PUBLICATIONS SYSTEM AND SYSTEM AND METHOD FOR TRANSLATING TEXT USED IN COMPUTER SOFTWARE UTILIZING AN EMBEDDED TRANSLATOR**, Microsoft Systems Journal, Jul. 18, 2000, 11]*

1.ABSTRACT

A multi-lingual data processing system can operate in a source language and one or more target languages by automatically translating text, such as application text and system message text. The multi-lingual data processing System includes Computer Software that can be developed and deployed in a source language and that obtains translated text from one or more translation tables corresponding the target languages as the computer software is runs on the system.

The translation system includes a translation table builder that creates and/or modifies the translation tables by importing translated text and/or by allowing a user to insert translated text. A translation configuration selector allows the user to select translation configuration setting such as the selected locality. The translation tables include application text translation tables containing translated text that is obtained by the application programs as each object is created by the application programs. The translation tables also include system message translation tables for each of the target languages containing translated message text such that messages logged by the system can be displayed with the translated message text from the system message translation table corresponding to the selected locality.

2. METHOD FOR TRANSLATING TEXT USED IN COMPUTER SOFTWARE UTILIZING AN EMBEDDED TRANSLATOR

2.1 FIELD OF THE INVENTION

The present invention relates to a data processing system having a multi-lingual capability and a system and method for translating text embedded in and used by computer software from a source language to a target language, and

more particularly, to a multi-lingual data processing system and translation system and method in Which a computer program is developed and deployed in a source language and then translated into the target language upon executing the computer program on a computer system.

a. BACKGROUND OF THE INVENTION

The global nature of the market place today has created an increased market for computer hardware and software that can be used on an international basis. Computer hardware and software vendors are thus required to modify existing products to accommodate a large number of foreign languages. While the demands upon hardware systems are minimal and are often solved by the provision of special keyboard characters, the large textual content of many software applications leads to a more difficult translation problem. The text that is embedded in and used by computer software must be translated to the desired foreign language(commonly referred to as localization).

Existing methods of translating or localizing computer programs involve direct revisions of the software code. After the application text has been translated to the target language, the program is then recompiled and linked. The application program is then limited to that particular target language and is sold for use in that target language only. This procedure is inefficient, time consuming, expensive, and subject to errors.

The translation made in the source code of the application program is typically made by a person without sufficient training in programming.

This can result in inadvertent changes being made to the syntax of the application program, causing fatal errors when trying to compile, link or run the resulting code With translated text. This process of embedding the translated text within the application program itself also results in the need to separately compile each foreign language version of the application program.

The translated application program is then more difficult to evaluate by support personnel who are unfamiliar with the target language of the translated application program. Directly translating the text in the code of the application program also can result in an inconsistent and improper display of the text on the computer screen.

2.3 SUMMARY OF THE INVENTION

The present invention features a multi-lingual data processing system including software, for processing data. The multi-lingual data processing also includes a locality setting Identifying a target language into Which text associated With the software is to be translated, and at least one text translation table corresponding to the target language identified by the locality setting and including target language text for use With the computer software on the data processing system. The computer software obtains the target language text from the text translation table corresponding to the target language identified by the locality setting. The multi-lingual data processing system preferably includes a translation configuration selector including a configuration tool, responsive to a user input, for selecting one or more translation configuration settings.

In one embodiment, the computer software includes at least one application program that generates a plurality of displayed objects having text. The text translation table includes at least one application text translation table including source language text items and target language text items for each of the displayed objects having text.

3. [*David bijl,* **SPEECH T0 TEXT CONVERSION,** *Microsoft Systems Journal, Jan. 9, 2001,19]*

1. ABSTRACT

A speech-to-text conversion system is provided Which comprises at least one user terminal for recording speech, at least one automatic speech recognition processor to generate text from a recorded speech and communication means operative to return a corresponding text to a user in Which said at least one user terminal is remote from said at least one automatic speech recognition processor, and server is provided remote from said at least one user terminal to control the transfer of recorded speech to a selected automatic speech recognition processor.

2. FIELD OF INVENTION

The present invention relates to apparatus and methods for speech to text conversion using automatic speech recognition, and has various aspects

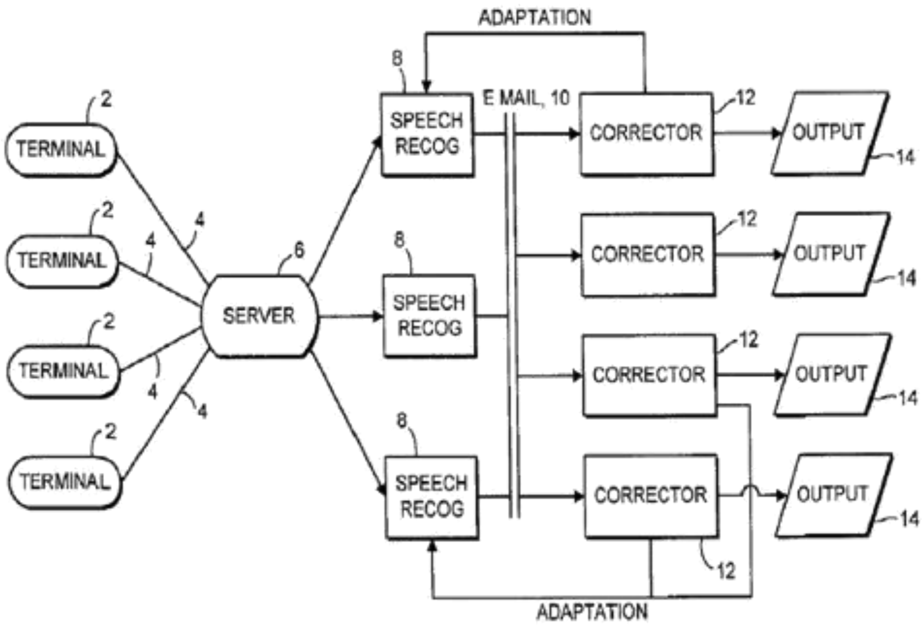


FIGURE 1

3. BACKGROUND OF THE INVENTION

Automatic speech recognition, as such, is known from, for example**,** **"Automatic Speech Recognition"**Conventional**.**known systems for converting speech to text involving automatic speech recognition are desktop stand alone systems, in which each user needs his or her own system.These known systems are able to transcribe human speech to text, imperfectly. The text results are presented to the user after a small delay whilst he or she is still dictating.

This has a number of disadvantages: Firstly the instantaneous text presentation can confuse and alter the behaviour of the user who is speaking. Also, it requires that the user must themselves correct errors, usually using a text editor . Accordingly, the user must switch between the tasks of speaking and correcting, resulting in inefficiency.IBM and Dragon have produced desktop speech to text conversion systems which are adapted to understand the speech of a particular user.

In a first aspect, the present invention relates to a speech to text convertor comprising at least one user terminal for recording speech, at least one automatic speech recognition processor, and communication means operative to return the resulting text to a user, in Which said at least one user terminal is remote from said at least one automatic speech recognition processor, the speech to text convertor including a server remote from said at least one user terminal, the server being operative to control transfer of recorded speech to a selected automatic speech recognition processor.

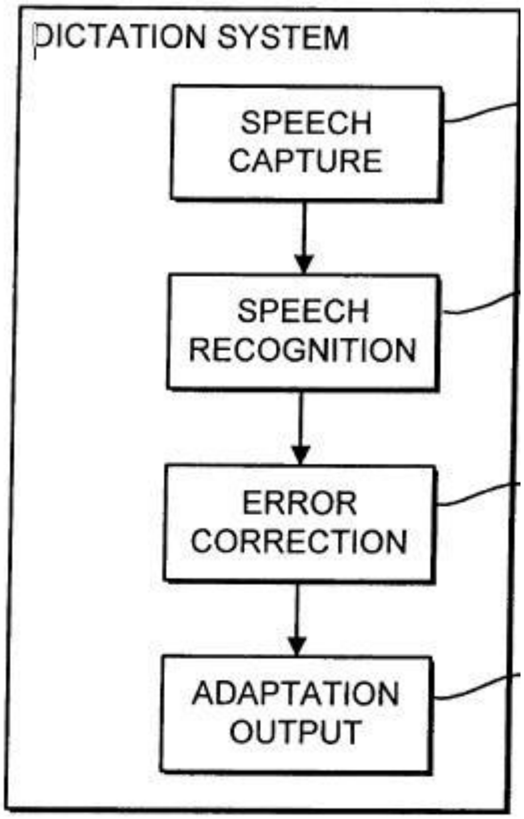


FIG. 2

4. SUMMARY OF INVENTION

Preferably , each user terminal communicates the recorded speech to the remote server by electronic mail. The use of electronic mail enables relaying information

from one terminal or machine to another, and preferably allows different operations (including entry to a dictation terminal, application of automatic speech recognition, and operation of a correction terminal) to occur on isolated computer networks. The machines which perform these separate operations need not be connected to any of the same equipment , or a common network, other than that loose (and usually global) network defined by an e-mail system .Furthermore, the respective machines and terminals need not be operated at the same time. All operations can be conducted in a manner that is off-line i.e involving batch rather than real time processing. The automatic speech recognition processors are preferably distributed remote from the server. The server preferably communicates with at least one speech recognition processor by electronic mail.

In a second aspect, the invention relates to a speech to text convertor comprising at least one user terminal for recording speech, at least one automatic speech recognition processor , and communication means operative to return the resulting text to a user, in which said at least one user terminal is remote from said at least one automatic speech recognition processor, in Which electronic mail is used to send text data resulting from automatic speech recognition together with the recorded speech data to a correction unit for manual correction. The text data and speech data are preferably sent together in a single preferably also includes timing data for relating text to speech. Preferably each Word of text has an associated start and end time recorded as part of the timing data. The text data can include text alternatives corresponding to a spoken word. Preferably said at least one user terminal and said at least one automatic speech recognition processor communicate using electronic mail.

Electronic mail can be used for communications between each of said at least one user terminal and a remote server which is operative to control assignment of the speech to the automatic speech recognition processors. The processors can be distributed remote from each other and the server. Electronic mail can also be used to send text to output terminals.

As regards the invention in both first and second aspects:

The recorded speech is preferably continuous speech .The server acts to control assignment of recorded speech for processing to automatic speech processors by

queuing the received speech and submitting them according to predetermined rules.

This allows more efficient use of the available automatic speech recognition resources, according to an off-line or batch processing scheme. Speech to text conversion can be done as a single fully automatic operation, or as a part-automatic and part-manual operation using the automatic speech recognition processor and corrector unit respectively.

5. .SPEECH RECORDING,PRE-PROCESSING AND TRANSFER TO THE SERVER

.

In the preferred embodiment, the user terminal 2 is a personal computer of the multi-media type including a voice recorder. In other embodiments, it could be a telephone handset or dictation recorder eg. dictaphone, or some other type of computer. A telephone handset can be digital or analog and could be cellular or radio handset or any other standard equipment for relaying audio speech. In further embodiments the user terminal could be a voice mail terminal or a recording apparatus as used in court rooms. The speech is recorded in continuous English, although the system could, of course, be adapted for other languages.

Once recorded, a request for dictation, which includes the recorded speech, is sent to the server 6. In preparing the dictation request, in some types of user terminal 2, as in the preferred embodiment, the speech is pre-processed so as to reduce the size of the dictation request. The dictation request is sent to the server 6 via E-mail.

The following operations having been performed on the user terminal:

pre-processing client speech into a dictation request, and packup-and-moving to a server (using electronic mail for transfer), the server acts to control the following operations:

1. Queuing the execution of the transferred dictation request,
2. Sending the dictation request to a speech recognition processor,
3. Dictation processing using automatic speech recognition,
4. Pack up-and-moving the resultant draft text document to a corrector terminal,

The correction terminal then is used to correct the draft text document, pack-and-moving the modifications of the dictation made at the connector terminal back to the server, the server then operates to:

5. Submit the modifications to the adaptive learning process of the ASR processors, and
6. Forward the text document to the originating client.

6. ERROR CORRECTION

At the corrector terminal 12 which includes personal computer, a VDU and a keyboard, the text file is viewed using standard word processor software. The original recorded speech is sent with the text file to the corrector terminal. The system includes several corrector terminals. The file is received by the corrector terminal 12 by E-mail.

The recorded speech is sent in a compressed format. The text is corrected using a conventional keyboard . Words highlighted in the text are simultaneous output as speech audio signals from a loud speaker of the correction terminal 12. This allows a simple “Watch of correction and which listen ”very time efficient.

4. *[Edwin R. Addison* **,TEXT TO SPEECH,** *Microsoft Systems Journal ,Mar. 8, 2005,22]*

1. ABSTRACT

A preferred embodiment of the method for converting text to speech using a computing device having a memory is disclosed. The inventive method comprises examining a text to be spoken to an audience for a specific communications purpose, followed by marking-up the text according to a phonetic mark up systems such as the Lessac System pronunciation rules notations. A set of rules to control a speech to text generator based on speech principles, such as Lessac principles. Such rules are of the tide normally implemented on prior art text-to-speech engines, and control the operation and a continuation of the Software and Characteristics of the speech now abandoned, and a continuation-in-part by a Computer using the Software to speak the marked-up text expressively. The step of using a computer to speak the marked-up- text is style of expression where each of the tonal, structural, and(Consonant energies, have a different balance in the Speech, are also spoken to a trained speech practitioners that listened to the spoken speech generated by the computer. The spoken (speech generated by the computer is then evaluated for consistency with style criteria and/or expressiveness. And audience is then assembled and the spoken speech generated by the computer is played back to the audience. Audience comprehension of spoken speech generated by the computer evaluated and correlated to a particular implemented rule which resulted relatively high audience comprehension are selected.

2. BACKGROUND OF THE INVENTION

While speech to text applications have experienced a remarkable evolution in accuracy and usefulness during the past ten or so years, pleasant, natural sounding easily intelligible text to speech functionality remains an elusive but sought-after goal. This remains the case despite what one might mistake as the apparent simplicity of converting known syllables with known sounds into speech, because of the subtleties of the audible cues in human speech, at least in the case of certain languages, such as English. In particular, while certain aspects of these audible cues have been identified, such as the increase in pitch at the end of a question Which might otherwise be declaratory in form, more subtle expressions in pitch and energy, some speaker specific, some optional and general in nature, and still others Word specific, combine With individual voice colour in the human voice to result in realistic speech.

In accordance with the invention, elements of individual speaker , randomness, and so forth are incorporated into output speech With varying degrees of implementation ,to achieve a pseudo-random effect. In addition, speaker is integrated with the same and combined with expressive models patterned on existing conventional speech coach to student voice training techniques. Such conventional techniques may include the system, which is aimed at improving intelligibility in the human voice in the context of theatrical and similar implementations of human speech.

In contrast to the inventive approach, conventional text to speech technology has concentrated on a mechanical, often high information density, approach. Perhaps the most convincing text to speech approach is the use of prerecorded

entire phrases, such as those used in some of the more sophisticated telephone answering applications. However, the amount of memory required for just a very few responses is relatively high and versatility is not a practical objective.

Still another approach is so-called “phrasesacing”where recorded human speech in the form of phrases is used to construct output speech. In addition, in accordance with this technology, the characteristics of segments of speech may be modified, for example by modifying them in duration, energy and pitch. In related approaches, such as play back, some of the problems of more limited systems are solved, such approaches tend to be both less intelligible and less natural than human speech. To a certain extent blending of pre recorded speech with synthetic speech will also solve some of these problems, but the output speech, while versatile and having wider vocabularies, is still relatively mechanical and character.

Another approach is to break up speech into its individual sounds or phonemes, and then to synthesize Words from these sounds. Such phonemes may be initially recorded human speech, but may have their characteristics varied so that the resulting phoneme has a different duration , pitch, energy or other characteristics or characteristics changed as compared to the original recording. Still another approach is to make multiple recordings of the phonemes, or integrate multiple recordings of words with word generation using phoneme building blocks.

3.SUMMARY OF THE INVENTION

A method for converting text to speech using a computing device having memory is disclosed. A text is received into the memory of the computing device. A set of the lexical parsing rules are applied to parse the text into a plurality of components. Pronunciation, and meaning information is associated with these components. A set of phrase parsing rules are used to generate marked up text. The marked up text is then phonetically parsed using phonetic parsing rules, and Lessac expressive parsing rules. The sounds are then stored in the memory of the computing device, each of the sounds being associated with pronunciation information .The sounds associated with the text maybe recalled to generate a raw speech signal from the marked up text after the arsing using phonetic and expressive parsing rules. In a preferred embodiment of the method for converting text to speech using a computing device having a memory is disclosed. Text, being made up of a plurality of Words, is received into the memory of the computing device. A plurality of phonemes are derived from the text.

Each of the phonemes is associated with a prosody record based on a database of prosody records associated with a plurality of Words.

A first set of the artificial intelligence rules is applied to determine context information associated with the text. The context influenced prosody changes for each of the phonemes is determined. Then a second set of rules, based on Lessac theory to determine Lessac derived prosody changes for each of the phonemes is applied. The prosody record for each of the phonemes is amended in response to the context influenced prosody changes and the Lessac derived prosody changes. Then a reading from the memory sound information associated with the phonemes is performed. The sound information is amended, based on the prosody record as amended in response to the context influenced prosody changes and the Lessac derived prosody changes to generate amended sound information for each of the phonemes. Then the sound information is outputted to generate a speech signal.

The sound information is associated with different speakers, and a set of artificial intelligence rules are used to determine the identity of the speaker associated with the sound information that is to be output.

4. DESCRIPTION OF THE BEST MODE

In accordance with the present invention, an approach to voice synthesis aimed to overcome the barriers of present system is provided. In particular, present day systems

based on pattern matching, phonemes, di-phones and signal process sounding speech with no significant level of human expressiveness. Applications of the present invention include customer service response systems, telephone answering systems, information retrieval, computer education, office assistance, and more.

Current speech synthesis tools are based on signal processing and filtering, With processing based on phonemes , diphones and/or phonetic analysis. Current systems are understandable, but largely have a robotic, mechanical , mushy or non human style to them. In accordance with the invention, speech synthesis is provided by implementing inventive features meant to simulate linguistic characteristics and knowledge-based processing to develop a machine implementable model simulating human speech by implementing human speech characteristics and a pseudo-natural text to speech model. There are numerous systems on the market today. While this would seem to validate an existing need for natural sounding text to speech systems, most current text to speech systems are based on old paradigms including pattern recognition and statistical processing, and achieving the less than desirable performance noted above.

The first task is referred to below as tokenization. In accordance with the invention , tokenization is used to extract a Word and punctuation list in sequential order from the text. The result is a word list and this word list is then processed using dictionary information. Processing includes looking up for each word :possible parts of speech Which it may constitute, depending upon context, possible ambiguity, and possible word combinations in various idiomatic phrases, Which are all contained in the dictionary consulted by the system following dictionary look up a, a phrase parser identifies the end of each phrase, removes lexical ambiguity and labels each word with its actual part of speech. Tokenization is completed with the generation of marked up text .The process of tokenization constitutes producing a word list for input text in a file or record being transformed into speech in accordance with the present invention.

5. *[S.K. SAMANTA, **AUTOMATIC LANGUAGE TRANSLATION FOR MOBILE SMS,** UNIVERSITY OF ESSEX ,COLCHESTER ,CO4 3NU, UK,VOLUME 2 ,ISSUE 1, 2010,16]*

1.ABSTRACT

Currently mobile users engage the services of a third party provider to translate an SMS text into a different language. The existing services have a number of drawbacks e.g. high cost to the user, not user friendly, they reduce the message space, and are inefficient . To communicate with a foreign person the sender must know the recipients preferred language and device display capability. What is needed is a service where a sender can send message in their native language without regard for the target tongue. Using an open source language translation package and a database server we show that a mobile operator can provide a transparent service where the text message is automatically converted to the recipients preferred language. We show that in comparison to the existing system our implementation is efficient and cost effective and has large implications for commerce, language learning and person-to-person communication.

2. INTRODUCTION

The need for language translation began when people started travelling from place to place and the first documented cases are between the 3rd and 1st centuries b.c. In modern times, telecommunications have allowed global interaction without the need to travel at all. Currently more than half of world’s population have a mobile phone and access to the global facilities. In most countries people use messaging services (e.g. short message and e-mail) more than the conventional voice calls. The mobile short messaging service (SMS) provides a convenient platform where the message can be delivered even if the recipient’s phone is engaged in voice communication.

Clearly not all people understand the same language and a textual message in the wrong language may be meaningless. Some messages are generated automatically and generally sent in one language, so many of the recipients may not understand. What is needed is a service that performs language translation according to the preferred language of the recipient, without them even knowing. The literature has focused on SMS language translation; where the service is implemented either in the mobile device or in the network

For translation in mobile devices an appropriate software interface such as Java Micro Edition (J2ME) is needed. This generally limits the number of languages which can be translated. Therefore devices with larger memory and high processing capability are required (e.g. phones with Symbian or Windows Mobile operating systems) which tend to be costly and can be a barrier to some people. Mobile users can avail the translation service without a costly handset if the translation is performed in the network; this is the focus of this paper.

3. BACKGROUND

In a mobile network an SMS is delivered using a store and forward mechanism Any device capable of sending and receiving a short textual message is designated as a Short Message Entity (SME). A mobile phone or an application server works as an SME. When a mobile user sends a text message, the network first transfers the message to a Short Message Service Centre (SMSC) which stores it for delivery to the recipient. In a similar way, application providers deliver value-added content (e.g. ring tone, news and weather reports) to the mobile users. The header of the message contains information such as the sender and recipient’s phone number. The SMSC stores the message and after collecting the routing information and status of the recipient’s handset (e.g. eligible to receive SMS), it forwards the message to the mobile network which deIn the process of delivery from the sender to the recipient the message content (i.e. text) transparently passes through the mobile network and the SMSC. If two people speak the same language (e.g. English) then they can exchange textual messages in their native language. If one speaks English and other German then there are two ways they could communicate: 1) one could have a phone which converts from one language to another, and 2) one of them could employ a third party translator. If the conversion is done on the phone then either the sender translates (e.g. English to German) before sending, or the text is sent in the sender’s native language and the receiver performs the translation. In either case the device must have the capability to translate many languages if it is to be truly international.

To translate a text message on the mobile, users needs to supply the source and target languages. This is not difficult for a sender if we assume that he/she knows the recipient's language preference. However the receivers face additional difficulty since they need to identify the source language from the received text. The recipient may not be able to identify the language not being a linguistic expert. Therefore the phone needs intelligent software to automatically detect the language and display the text in the user's native tongue. This will increase the cost of the phone which may be prohibitive to some people. In the absence of a high end handset either the sender or the recipient can engage a third party (who makes available a centralized server in the network) to provide the translation service delivers to the recipient.

4. IMPLEMENTATION

Using the Google Translate java API (an open source language translation package) and a database server we implement the functionality of ASLT. Our implementation allows message communication in the user's native language and also helps users in language teaching. The implementation is applicable to any messaging service (e.g. SMS, MMS and e-mail) but here we emphasis on SMS. It is assumed that the mobile handsets have the capabilities to input and display the preferred language of the user.

we show the architecture of ASLT where a box (containing language translation software and a database server) is inserted in the SMS delivery path from SMSC to mobile users. Mobile users who want to receive their communications in their preferred (e.g. native) language indicate their preferences once only by sending an SMS which is stored in the database of the ASLT. When an SMS is received (from a mobile user or an application server), the ASLT manager searches the recipient's number in the database.

If the entry is not found, the message is transparently delivered. If the number is available in the database, the language preference along with the text content is forwarded to the translator. The translator automatically detects the source language and performs the translation. The translated content is forwarded to the mobile network for ultimate delivery to the recipient. In addition to being user friendly, Fig. 4 indicates that in our implementation there is no change in

SMSC traffic.

We simulated the functionality of the ASLT on a PC using: The Google Translate java API and our own software also in java. Using J2ME (Java Micro Edition) as the Short Message Entity (SME) we simulated the origin and reception of the SMS. We show the screen shoot of our simulation. A text message: “All life is an experiment. The more experiments you make the better.” is sent to the recipient (number “8888888”) whose language preference is German. The message is automatically translated to “Alles Leben ist ein Experiment. Je mehr Sie machen Experimente, desto besser.” in German and delivered to the Recipient’s phone. The implementation in a real world network is relatively straightforward; it works as a relay box and therefore can work independently of other units. It can scale with the network since multiple threads allow translations in parallel.

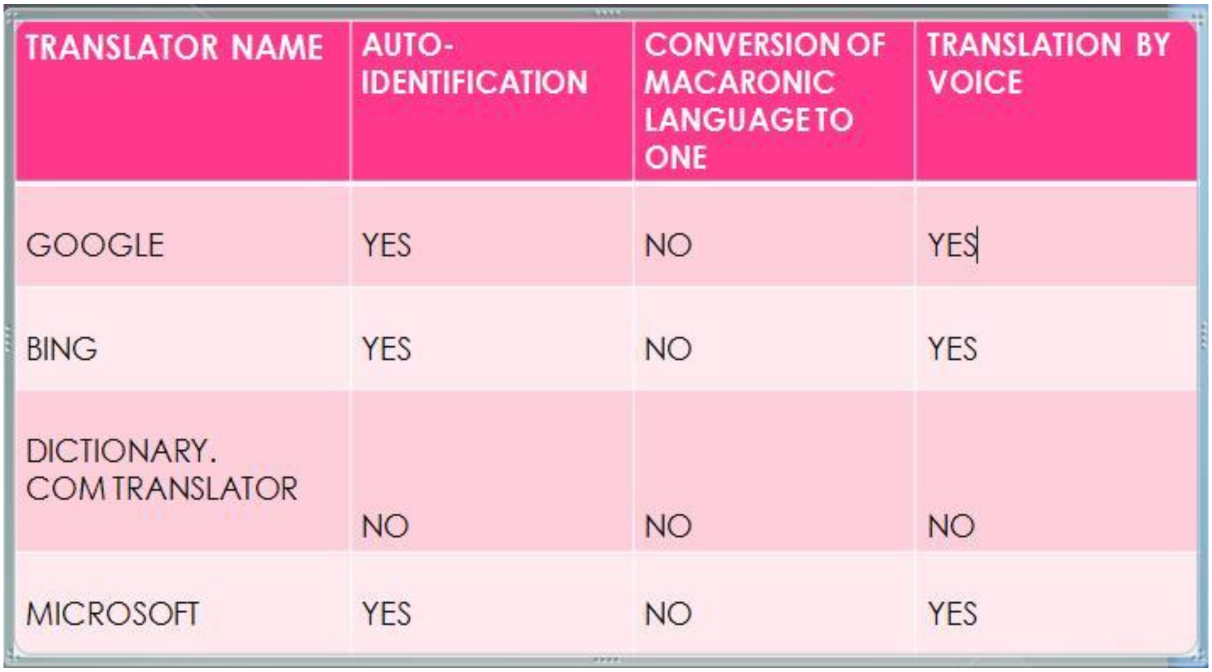
5. CONCLUSION

Using an open source language translation package and a database server we implemented automatic language translation for SMS text. We show that our mechanism will allow mobile operators to provide textual message delivery in a user’s native language without the need for a third party service provider. We show that in comparison to the third party provider our mechanism is user friendly and efficient. We demonstrate that using our platform a large number of services such as advertisement, auctions and sale of goods and services can be delivered to those who are currently not able to access them due to the language barrier.

CHAPTER 3

DESIGN AND DOCUMENTATION

1. COMPARISION BETWEEN VARIOUS TRANSLATORS



TRANSLATOR NAME	AUTO-IDENTIFICATION	CONVERSION OF MACARONIC LANGUAGE TO ONE	TRANSLATION BY VOICE
GOOGLE	YES	NO	YES
BING	YES	NO	YES
DICTIONARY. COM TRANSLATOR	NO	NO	NO
MICROSOFT	YES	NO	YES

TABLE 1

This table depicts the comparisons between various existing translators .

AUTO-IDENTIFICATION:

Google , Bing , Microsoft and my translator does auto detection. With the help of auto detection feature the language of the text to be translated can be auto-identified. It is very

useful in situations when the user doesn’ which language.

CONVERSION OF MACRONIC LANGUAGE TO ONE:

Macronic language is basically the SMS language ,none of the translator converts macronic language or sms language to one i.e English. Research is still going on .

TRANSLATION BY VOICE:

Google, Bing , microsoft and my translator does translation by voice. It translates the given text into the specified language in my case its English and then speaks the translated text.

CONSTRAINTS:

Google : Google translator works more efficiently on android phones .It has speech to text conversion facility available only on the android application.

Microsoft: It is paid application.

Bing : It has speech to text coversion facility for windows 8 and 8.1 and not for lower versions.

2. DIAGRAMATIC REPRESENTATIONS

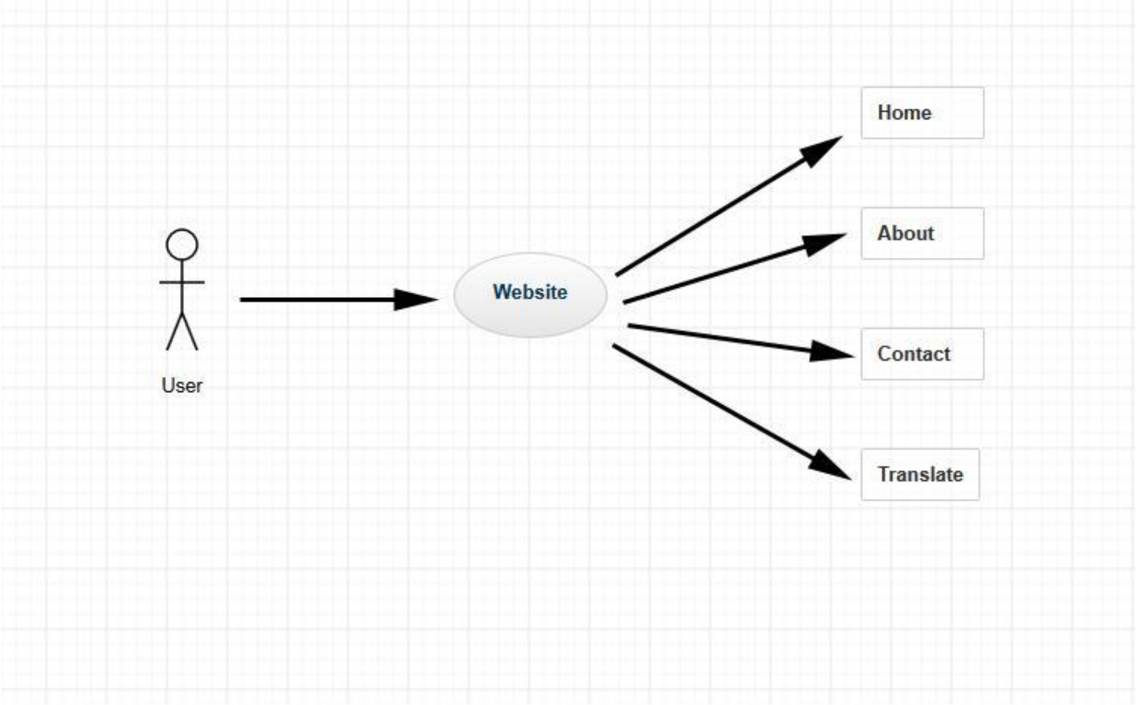


FIGURE 3

It represents the components of the front view of the web based application.

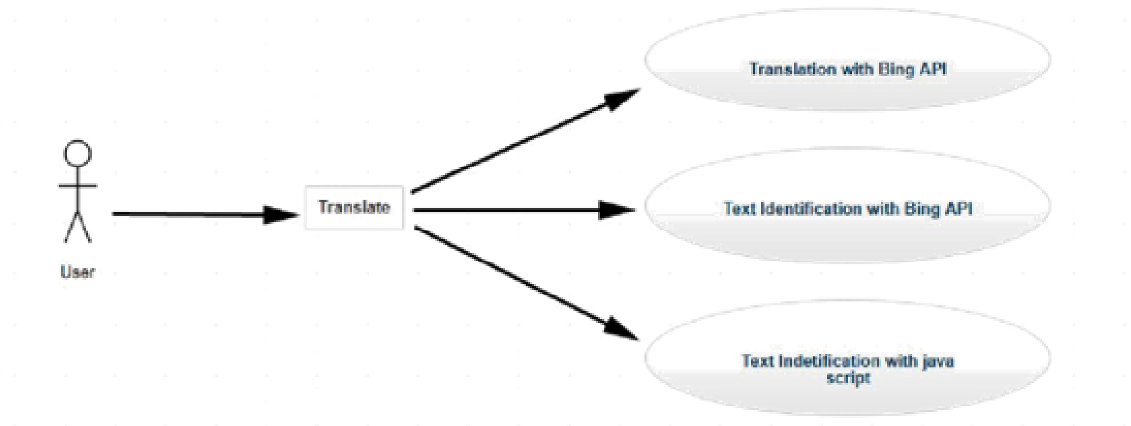

Translation with Bing API
User
Translate
Text Identification with Bing API
Text Indetification with java script

FIGURE 4

It represents the translation part further has 3 components.

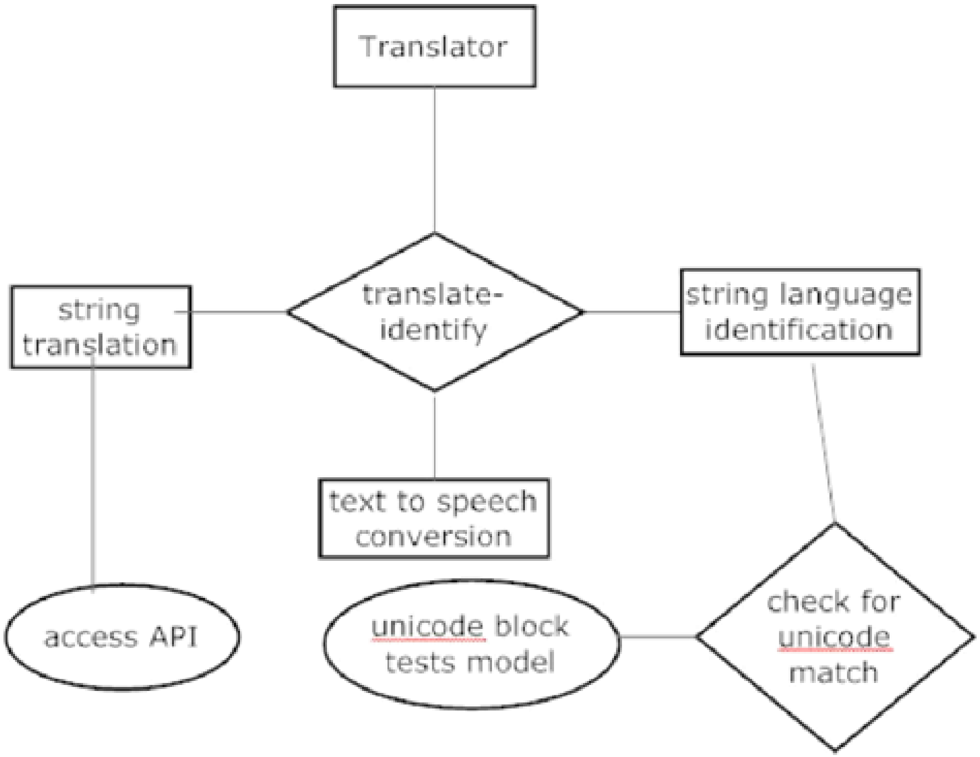

Translator
string translation
translate-identify
string language identification
text to speech conversion
access API
unicode block tests model
check for unicode match

FIGURE 5

This is an ENTITY RELATIONSHIP DIAGRAM(E-R D)

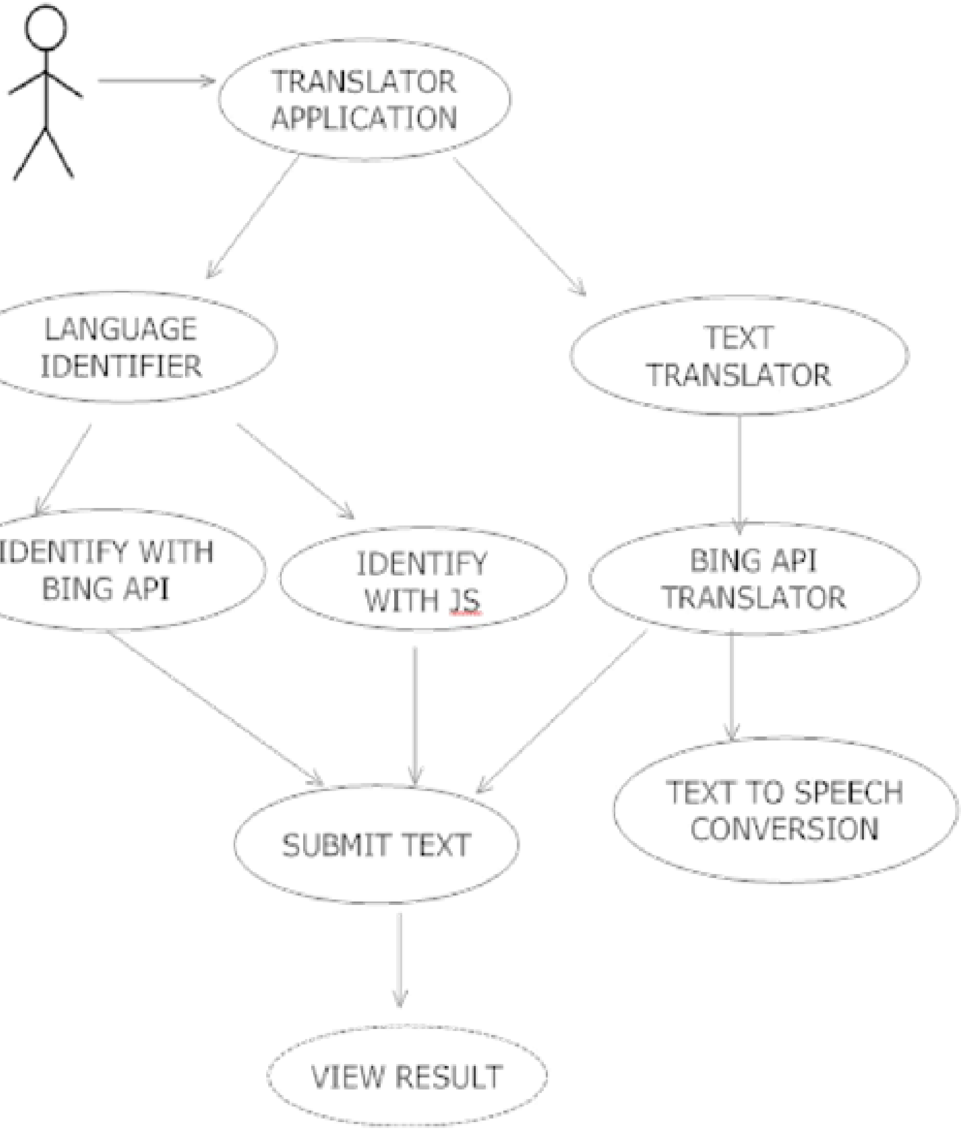


FIGURE 6

This is the USE CASE DIAGRAM

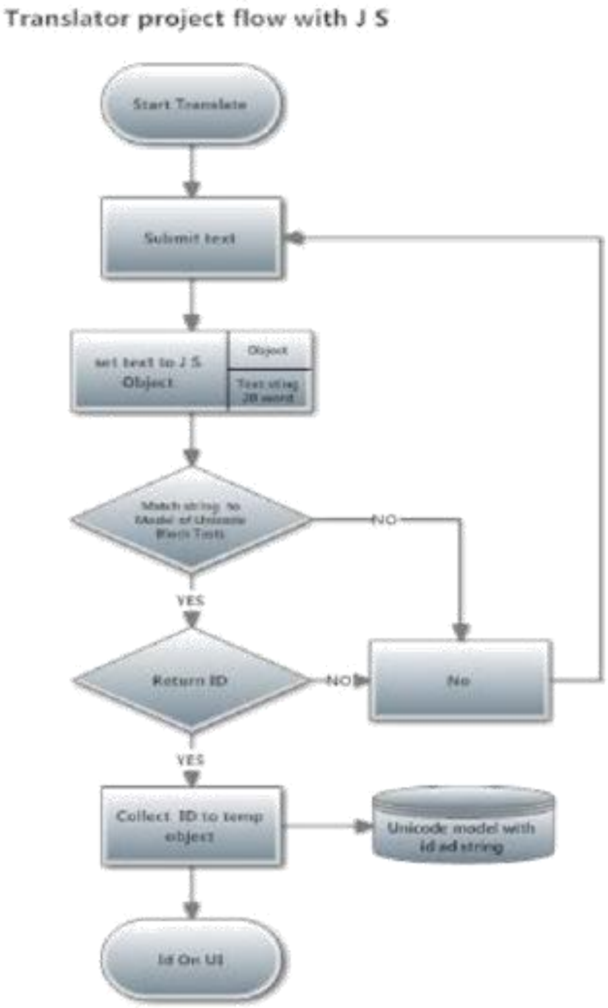

Translator project flow with J S
Start Translate
Submit text
set text to J S Object
Object
Text string 20 word
Match string to Model of Unicode Bech Tests
NO
YES
Return ID
NO
No
YES
Collect ID to temp object
Unicode model with id ad string
Id On UI

FIGURE 7

This is the FLOW DIAGRAM of the section of project , where translation is happening with the help of javascript.

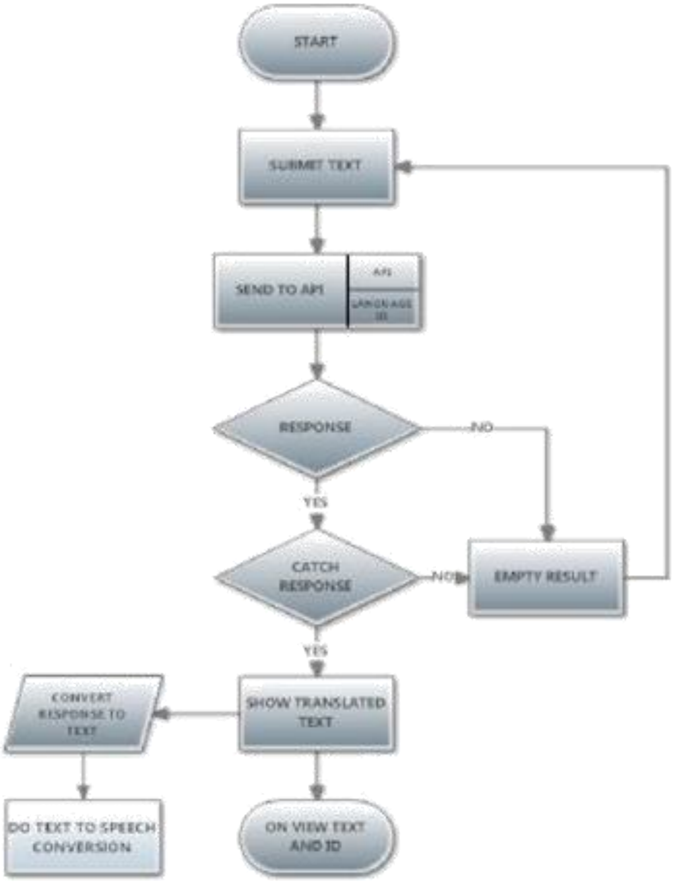


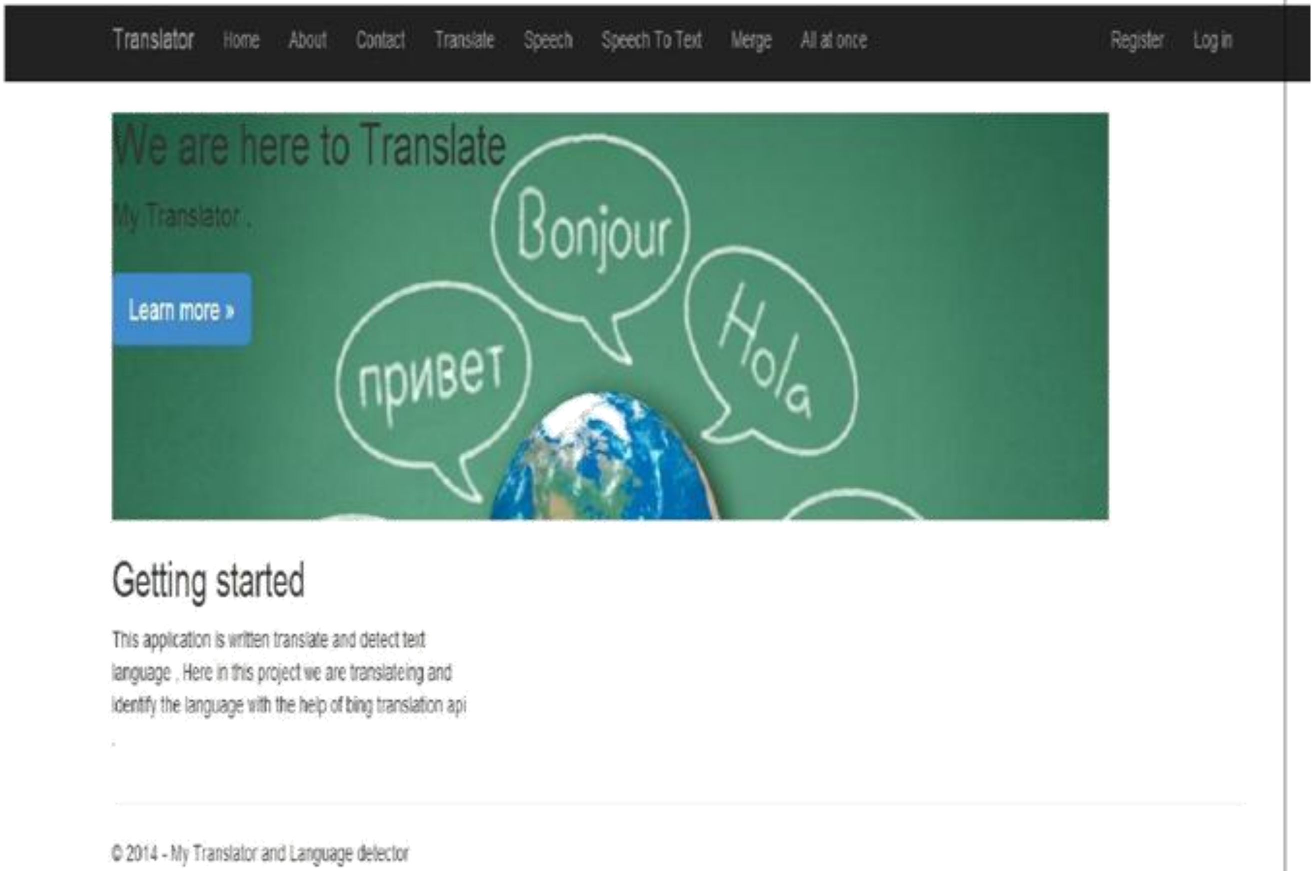
FIGURE 8

This figure depicts the flow diagram of the process of translation through Bing API,

CHAPTER-5

IMPLEMENTATION

1. SCREENSHOTS



We are here to Translate

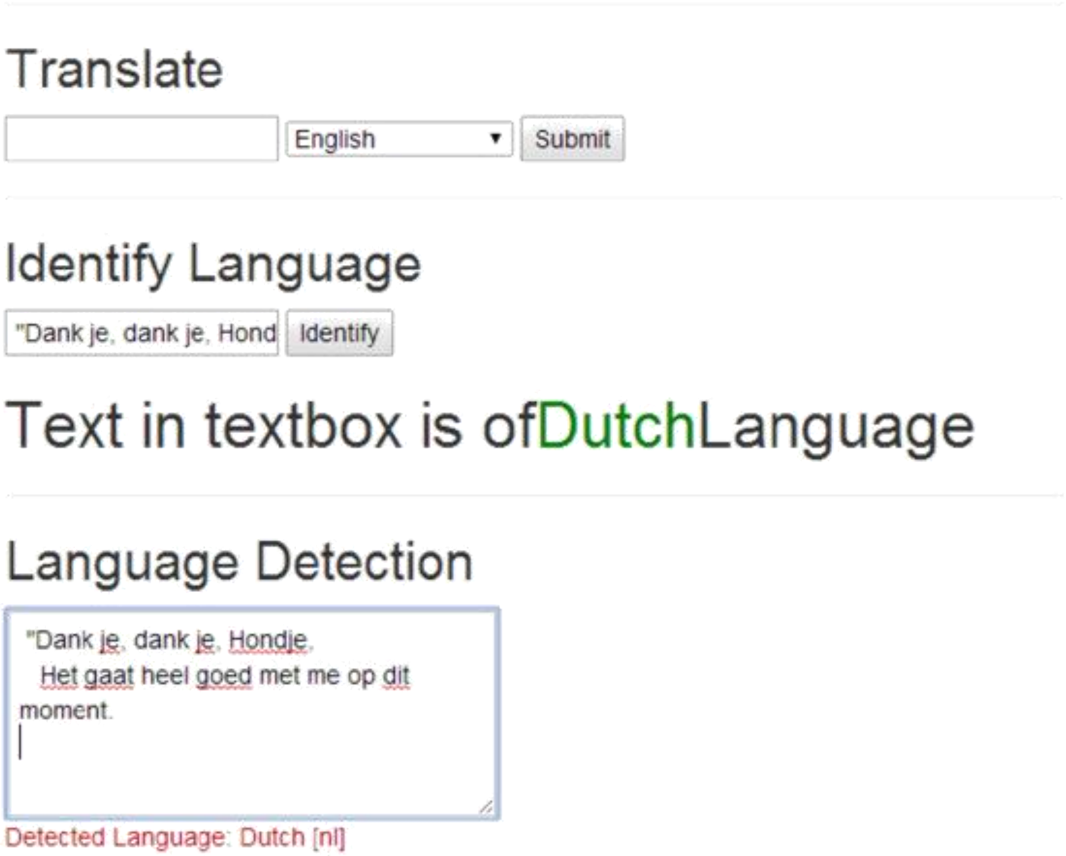
My Translator .

Getting started

This application is written translate and detect text language . Here in this project we are translateing and identify the language with the help of bing translation api .

© 2014 - My Translator and Language detector

This is the screenshot of the home page



"Dank je, dank je, Hond

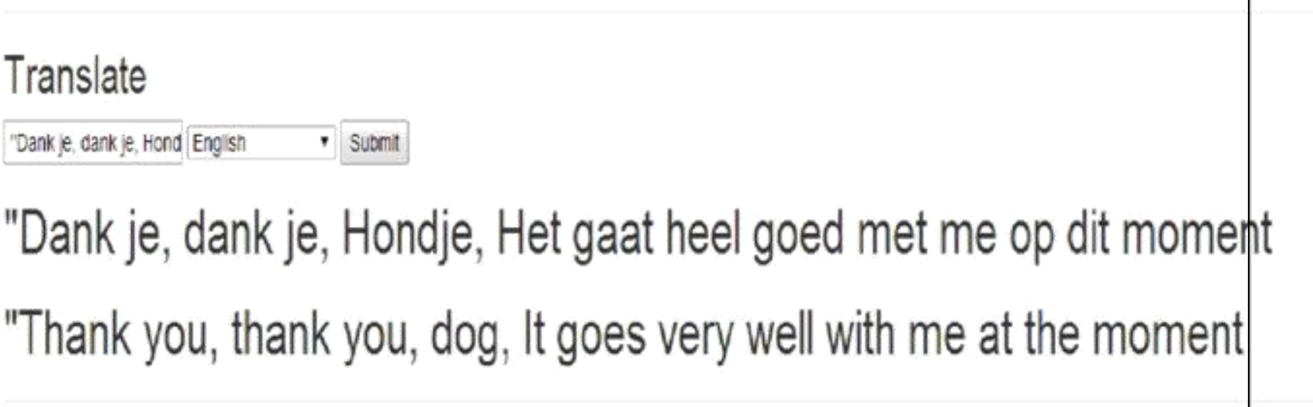
Text in textbox is ofDutchLanguage

"Dank je, dank je, Hondje,
Het gaat heel goed met me op dit moment.

Detected Language: Dutch [nl]

This is the screenshot of the main translation page,where we have 3 options :

1. First option is to translate the text into English.
2. Second option is to identify the text is written in which language
3. Third text box also does the language detection part.

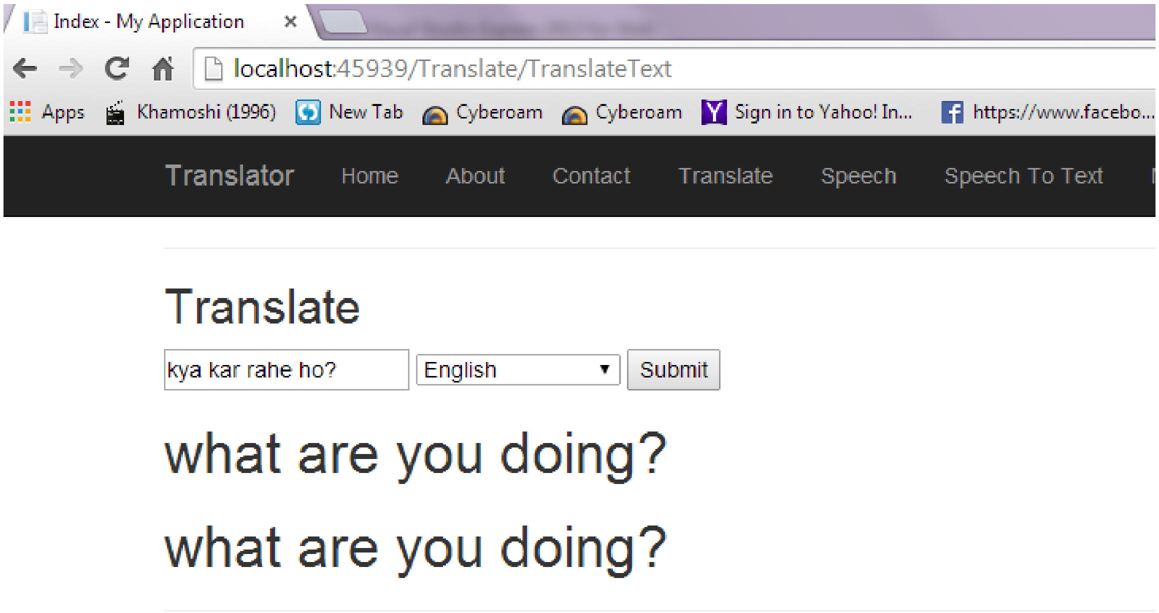


"Dank je, dank je, Hond

"Dank je, dank je, Hondje, Het gaat heel goed met me op dit moment

"Thank you, thank you, dog, It goes very well with me at the moment

Here, It translates the text into English and then speaks the translated text.

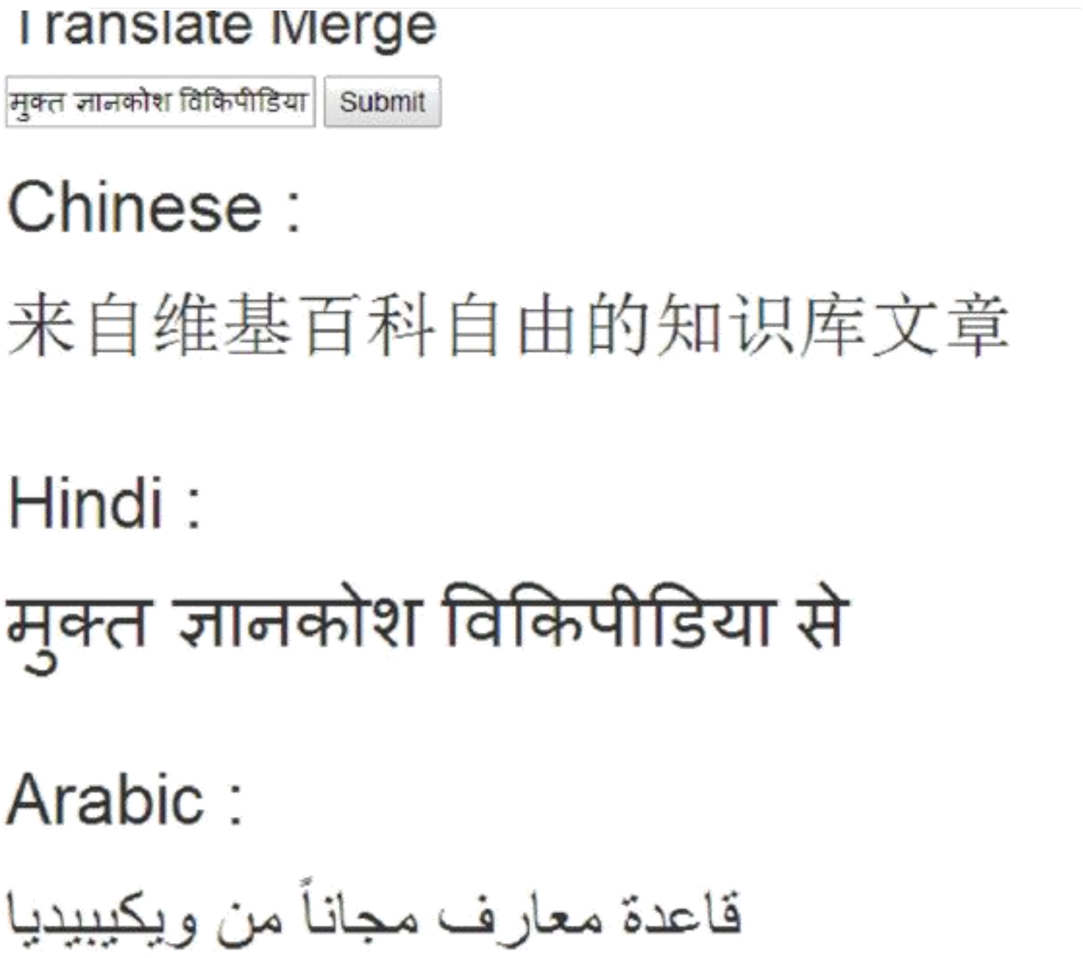


kya kar rahe ho?

what are you doing?

what are you doing?

It translates SMS language into English and then speaks the translated text.


Translate Merge

मुक्त ज्ञानकोश विकिपीडिया

Chinese :

来自维基百科自由的知识库文章

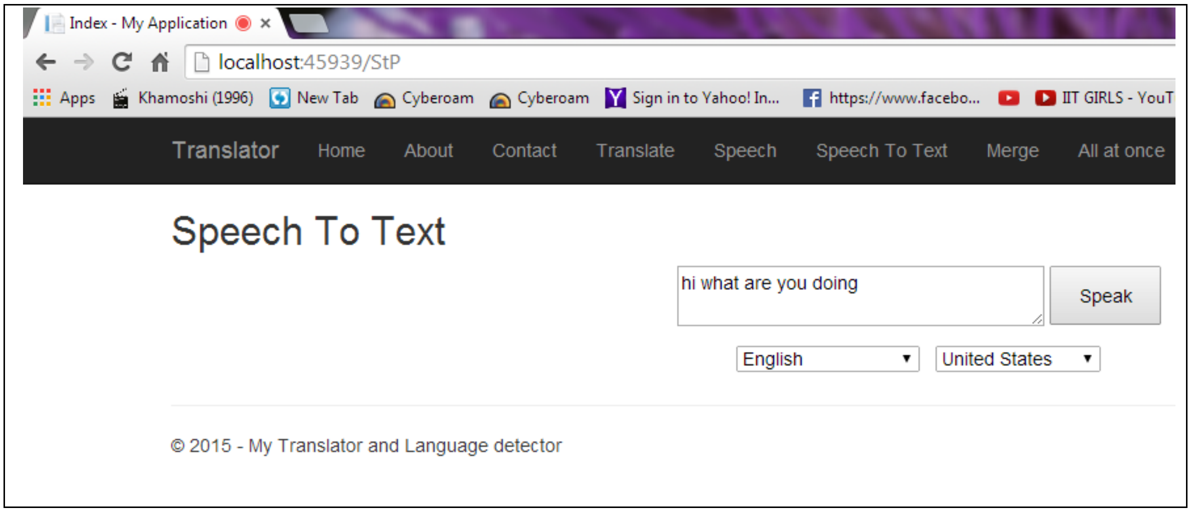
Hindi :

मुक्त ज्ञानकोश विकिपीडिया से

Arabic :

قاعدة معارف مجاناً من ويكيبيديا

Here,it translates the given text into 5 languages.



Speech To Text

hi what are you doing

English ▾ United States ▾

© 2015 - My Translator and Language detector

Here, Speech to text conversion happens.

CONCLUSION

A web based application has been developed that translates multilingual text into a uniform language English. It can easily detect the language in which the text is written and can easily translate the text of any language / languages into English . It can also convert SMS language into English which is certainly unique. It also has the feature of text to speech and speech to text conversion . This application has been developed on visual studio 2013. The data can be accessed, manipulated and retrieved very easily. To conclude this application has proved to be a user friendly interface.

FUTURE WORKS

In future I would try to find the shortcomings of existing translators and work on them .I would try to implement some of the new features and the key features of the translators that has not been implemented as of now. In future I will try to translate macronic language or sms language to one uniform language English. I will be working more on the merging part and speech to text conversion. .

REFERENCES

1. Rodney P- Hamann ,MULTI-LINGUAL DATA PROCESSING OTHER PUBLICATIONS SYSTEM AND SYSTEM AND METHOD FOR TRANSLATING TEXT USED IN COMPUTER SOFTWARE UTILIZING AN EMBEDDED TRANSLATOR, Microsoft Systems Journal, Jul. 18, 2000, 11

2. David bijl , SPEECH T0 TEXT CONVERSION, Microsoft Systems Journal, Jan. 9, 2001, 19.

3. Edwin R. Addison ,TEXT TO SPEECH, Microsoft Systems Journal , Mar. 8, 2005,22

4. S.K. Samanta ,AUTOMATIC LANGUAGE TRANSLATION FOR MOBILE SMS, university of Essex, Volume 2,Issue 1,2010, 16

5. http://www.translate.google.co.in

6. http:// www.wikipedia.com

7. http:// www.bing.com/translator/

8. http:// www.microsoft.com/translator/

9. http://www.vocapia.com (fot text to speech conversion)

10. http://www.dictionary.reference.com

APPENDIX

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using TranslationD.Models;
using SpeechLib;
using System.Xml;
using System.Web.Mvc.Html;


namespace TranslationD.Controllers
{
    public class AllAtOneController : Controller
    {
        //
        // GET: /AllAtOne/
        public ActionResult Index()
        {
            return View();
        }


        [HttpPost]
        public ActionResult TranslateText(TranslateSummary ts)
        {

            string strTranslatedText = null;
            TranslateSummary TranslateSummaryObj = new TranslateSummary();
            string SreviceResult;
            string thirdLanguage1 = (ts.Text == null ? "User is currently only selected 
English" : ts.Text);
            try
            {
              //  string fromCulture1 = ts.TextCulture.ToString(); // "zh-cn";
                string toCulture1 = "en";
                string appId1 = "78280AF4DFA1CE1676AFE86340C690023A5AC139";
                //string translation = null;

                TranslatorService.LanguageServiceClient client = new 
TranslatorService.LanguageServiceClient();
                client = new TranslatorService.LanguageServiceClient();
                TranslateSummaryObj.Eng = client.Translate(appId1, thirdLanguage1, "", "en");
                 TranslateSummaryObj.hi = client.Translate(appId1, thirdLanguage1, "", "hi");
                 TranslateSummaryObj.Ar = client.Translate(appId1, thirdLanguage1, "", "ar-
dz");

                 TranslateSummaryObj.Fr = client.Translate(appId1, thirdLanguage1, "", "fr-
be");

                 TranslateSummaryObj.Chi = client.Translate(appId1, thirdLanguage1, "", "zh-
cn");

                // txtTranslatedText.Text = strTranslatedText;

                //create voice object
                SpVoice Voice = new SpVoice();


                if (strTranslatedText != null)
                {
                    SreviceResult = strTranslatedText;
                }
                else
                {
                    SreviceResult = "Unable to process this Sorry.";
                }

                if (string.IsNullOrEmpty(SreviceResult))
                {

                    return null;
                }
            }
            catch (Exception ex)
            {

            }
            SreviceResult = null;
            return View("Index", TranslateSummaryObj);
        }
       }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace TranslationD.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult About()
        {
            ViewBag.Message = "Your application description page.";

            return View();
        }

        public ActionResult Contact()
        {
            ViewBag.Message = "Your contact page.";

            return View();
        }
    }
}

using System;
using System.Collections.Generic;
using System.Text;
using System.Collections;
using System.Text.RegularExpressions;

namespace LiteMiner.classes
{
    public class TrigramModel
    {
        Trigram[] trigrams;
        public Trigram[] Trigrams
        {
            get { return trigrams; }
        }

        public TrigramModel(Hashtable trigramsAndCounts)
        {
            List<string> keys2 = new List<string>();
            List<int> scores2 = new List<int>();
            //convert hashtable to arrays
            foreach (string key in trigramsAndCounts.Keys)
            {
                keys2.Add(key);
                scores2.Add((int)trigramsAndCounts[key]);
            }

            string[] keys = keys2.ToArray();
            int[] scores = scores2.ToArray();

            // sort array results
            Array.Sort(scores, keys);
            Array.Reverse(keys);
            Array.Reverse(scores);

            //build final array
            List<Trigram> result = new List<Trigram>();
            for (int x = 0; x < keys.Length; x++)
            {
                result.Add(new Trigram(keys[x], scores[x]));
            }

            trigrams = result.ToArray();

        }

        public TrigramModel(string[] tgrams)
        {
            List<string> keys2 = new List<string>();
            List<int> scores2 = new List<int>();
            //convert hashtable to arrays
            int score = 0;
            foreach (string key in tgrams)
            {
                keys2.Add(key);
                scores2.Add(score++);
            }

            string[] keys = keys2.ToArray();
            int[] scores = scores2.ToArray();

            // sort array results
            Array.Sort(scores, keys);
            Array.Reverse(keys);
            Array.Reverse(scores);
            //build final array
            List<Trigram> result = new List<Trigram>();
            for (int x = 0; x < keys.Length; x++)
            {
                result.Add(new Trigram(keys[x], scores[x]));
            }

            trigrams = result.ToArray();

        }

        public bool HasTrigram(string trigram)
        {
            foreach (Trigram t in trigrams)
            {
                if (t.t == trigram) return true;
            }
            return false;
        }

        public int GetScore(string trigram)
        {
            foreach (Trigram t in trigrams)
            {
                if (t.t == trigram) return t.score;
            }
            throw new Exception("No score found for '" + trigram + "'");
        }
    }

    public class Trigram
    {
        public string t = null;
        public int score = 0;
        public Trigram(string t, int s)
        {
            this.t = t;
            score = s;
        }
    }

    public class LanguageDetector
    {
        public const int MAX_LENGTH = 4096;
        public const int MIN_LENGTH = 20;
        public const int MAX_GRAMS = 300;

        LanguageStatistics langStat = new LanguageStatistics();
        Hashtable NAME_MAP = new Hashtable() {
        {"ab", "Abkhazian"},
        {"af", "Afrikaans"},
        {"ar", "Arabic"},
        {"az", "Azeri"},
        {"be", "Belarusian"},
        {"bg", "Bulgarian"},
        {"bn", "Bengali"},
        {"bo", "Tibetan"},
        {"br", "Breton"},
        {"ca", "Catalan"},
        {"ceb", "Cebuano"},
        {"cs", "Czech"},
        {"cy", "Welsh"},
        {"da", "Danish"},
        {"de", "German"},
        {"el", "Greek"},
        {"en", "English"},
        {"eo", "Esperanto"},
        {"es", "Spanish"},
        {"et", "Estonian"},
        {"eu", "Basque"},
        {"fa", "Farsi"},
        {"fi", "Finnish"},
        {"fo", "Faroese"},
        {"fr", "French"},
        {"fy", "Frisian"},
        {"gd", "Scots Gaelic"},
        {"gl", "Galician"},
        {"gu", "Gujarati"},
        {"ha", "Hausa"},
        {"haw", "Hawaiian"},
        {"he", "Hebrew"},
        {"hi", "Hindi"},
        {"hr", "Croatian"},
        {"hu", "Hungarian"},
        {"hy", "Armenian"},
        {"id", "Indonesian"},
        {"is", "Icelandic"},
        {"it", "Italian"},
        {"ja", "Japanese"},
        {"ka", "Georgian"},
        {"kk", "Kazakh"},
        {"km", "Cambodian"},
        {"pt_PT", "Portuguese (Portugal)"},
        {"pt_BR", "Portuguese (Brazil)"},
        {"ro", "Romanian"},
        {"ru", "Russian"},
        {"sa", "Sanskrit"},
        {"sh", "Serbo-Croatian"},
        {"zh_TW", "Traditional Chinese (Taiwan)"},
        {"zu", "Zulu"}
        };

        string[] SINGLETONS = {
        "Armenian", "hy",
        "Hebrew", "he",
        "Bengali", "bn",
        "Gurmukhi", "pa",
        "Greek", "el",
        "Gujarati", "gu",
        "Oriya", "or",
        "Tamil", "ta",
        "Telugu", "te",
        "Kannada", "kn",
        "Malayalam", "ml",
        "Sinhala", "si",
        "Thai", "th",
        "Lao", "lo",
        "Tibetan", "bo",
        "Burmese", "my",
        "Georgian", "ka",
        "Mongolian", "mn",
        "Khmer", "km"
        };
```

```
        string[] BASIC_LATIN = {"en", "ceb", "ha", "so", "tlh", "id", "haw", "la", "sw",
"eu", "nr", "nso", "zu", "xh", "ss", "st", "tn", "ts"};
        string[] EXTENDED_LATIN = {"cs", "af", "pl", "hr", "ro", "sk", "sl", "tr", "hu",
"az", "et", "sq", "ca", "es", "fr", "de", "nl", "it", "da", "is", "no", "sv", "fi", "lv",
"pt", "ve", "lt", "tl", "cy", "vi"};
        string[] ALL_LATIN;
        string[] CYRILLIC = {"ru", "uk", "kk", "uz", "mn", "sr", "mk", "bg", "ky"};
        string[] ARABIC = {"ar", "fa", "ps", "ur"};
        string[] DEVANAGARI = {"hi", "ne"};
        string[] PT = {"pt_BR", "pt_PT"};
        Hashtable RegexCache = new Hashtable();

        // Unicode char greedy regex block range matchers
        string[] unicodeBlockTests = {
        "Basic Latin", "[\u0000-\u007F]",
        "Latin-1 Supplement", "[\u0080-\u00FF]",
        "Latin Extended-A", "[\u0100-\u017F]",
        "Latin Extended-B", "[\u0180-\u024F]",
        "IPA Extensions", "[\u0250-\u02AF]",
        "Spacing Modifier Letters", "[\u02B0-\u02FF]",
        "Combining Diacritical Marks", "[\u0300-\u036F]",
        "Greek and Coptic", "[\u0370-\u03FF]",
        "Cyrillic", "[\u0400-\u04FF]",
        "Cyrillic Supplement", "[\u0500-\u052F]",
        "Armenian", "[\u0530-\u058F]",
        "Hebrew", "[\u0590-\u05FF]",
        "Arabic", "[\u0600-\u06FF]",
        "Syriac", "[\u0700-\u074F]",
        "Arabic Supplement", "[\u0750-\u077F]",
        "Thaana", "[\u0780-\u07BF]",
        "NKo", "[\u07C0-\u07FF]",
        "Devanagari", "[\u0900-\u097F]",
        "Bengali", "[\u0980-\u09FF]",
        "Gurmukhi", "[\u0A00-\u0A7F]",
        "Gujarati", "[\u0A80-\u0AFF]",
        "Oriya", "[\u0B00-\u0B7F]",
        "Tamil", "[\u0B80-\u0BFF]",
        "Telugu", "[\u0C00-\u0C7F]",
        "Kannada", "[\u0C80-\u0CFF]",
        "Malayalam", "[\u0D00-\u0D7F]",
        "Sinhala", "[\u0D80-\u0DFF]",
        "Thai", "[\u0E00-\u0E7F]",
        "Lao", "[\u0E80-\u0EFF]",
        "Tibetan", "[\u0F00-\u0FFF]",
        "Myanmar", "[\u1000-\u109F]",
        "Georgian", "[\u10A0-\u10FF]",
        "Hangul Jamo", "[\u1100-\u11FF]",
        "Ethiopic", "[\u1200-\u137F]",
        "Ethiopic Supplement", "[\u1380-\u139F]",
        "Cherokee", "[\u13A0-\u13FF]",
        "Unified Canadian Aboriginal Syllabics", "[\u1400-\u167F]",
        "Ogham", "[\u1680-\u169F]",
        "Runic", "[\u16A0-\u16FF]",
        "Tagalog", "[\u1700-\u171F]",
        "Hanunoo", "[\u1720-\u173F]",
        "Buhid", "[\u1740-\u175F]",
        "Tagbanwa", "[\u1760-\u177F]",
        "Khmer", "[\u1780-\u17FF]",
        "Mongolian", "[\u1800-\u18AF]",
        "Limbu", "[\u1900-\u194F]",
        "Tai Le", "[\u1950-\u197F]",
        "New Tai Lue", "[\u1980-\u19DF]",
        "Khmer Symbols", "[\u19E0-\u19FF]",
        "Buginese", "[\u1A00-\u1A1F]",
        "Balinese", "[\u1B00-\u1B7F]",
        "Phonetic Extensions", "[\u1D00-\u1D7F]",
        "Phonetic Extensions Supplement", "[\u1D80-\u1DBF]",
        "Combining Diacritical Marks Supplement", "[\u1DC0-\u1DFF]",
        "Latin Extended Additional", "[\u1E00-\u1EFF]",


        public LanguageDetector()
        {
            List<string> temp = new List<string>(BASIC_LATIN);
            temp.AddRange(EXTENDED_LATIN);
            ALL_LATIN = temp.ToArray();
        }

        private Hashtable FindRuns(string text)
        {

            Hashtable relevant_runs = new Hashtable();

            for (int x = 0; x < unicodeBlockTests.Length; x+=2)
            {
                string name = unicodeBlockTests[x];
                string regex = unicodeBlockTests[x + 1];
                if (RegexCache[name] == null) RegexCache[name] = new Regex(regex);
                // Count the number of characters in each character block.
                int charCount = ((Regex)RegexCache[name]).Matches(text).Count;

                // return run types that used for 40% or more of the string
                // always return basic latin if found more than 15%
                // and extended additional latin if over 10% (for Vietnamese)
                double pct = (double)charCount / (double)text.Length;

                relevant_runs[name] = pct;

            }

            return relevant_runs;
        }

        public string GetLanguageNameByCode(string code)
        {
            if (string.IsNullOrEmpty(code)) return null;
            code = code.ToLower();
            if (NAME_MAP[code] == null) return null;
            return NAME_MAP[code] as string;
        }

        public string Detect(string text)
        {

            if (string.IsNullOrEmpty(text)) return null;

            if (text.Length > MAX_LENGTH) text = text.Substring(0,MAX_LENGTH);
            text = Regex.Replace(text, "[\u0021-\u0040„”|\\-]", "");//remove numbers and
punctuations
            text = text.Replace("\t", " ").Replace("\n", " ").Replace("\r", " ");
            text = text.Replace("  ", " ").Replace("  ", " ").Replace("  ", " ").Replace("
", " ");//normalize multiple spaces
            return Identify(text.Trim());
        }

        private string Identify(string text)
        {

            Hashtable scripts = FindRuns(text);

            // Identify the language using most significant character usage.
            if ((double)scripts["Hangul Syllables"] + (double)scripts["Hangul Jamo"] + 
(double)scripts["Hangul Compatibility Jamo"] >= 0.4) {
                return "ko";
            }

            if ((double)scripts["Greek and Coptic"] >= 0.4) {
                return "el";
            }

            if ((double)scripts["Hiragana"] + (double)scripts["Katakana"] + 
(double)scripts["Katakana Phonetic Extensions"] >= 0.2) {
                return "ja";
            }

            if ((double)scripts["CJK Unified Ideographs"] + (double)scripts["Bopomofo"] + 
(double)scripts["Bopomofo Extended"] + (double)scripts["KangXi Radicals"] >= 0.4) {
                return "zh";
            }

            if ((double)scripts["Cyrillic"] >= 0.4) {
                return Check(text, CYRILLIC);//decide language using cyrillic letters
            }

            if ((double)scripts["Arabic"] + (double)scripts["Arabic Presentation Forms-A"] + 
(double)scripts["Arabic Presentation Forms-B"] >= 0.4) {
                return Check(text, ARABIC); //decide language using arabic letters
            }

            if ((double)scripts["Devanagari"] >= 0.4) {
                return Check(text, DEVANAGARI);
            }

            // Try languages with unique scripts
            for (int x = 0; x < SINGLETONS.Length; x+=2)
            {
                string name = SINGLETONS[x];
                string code = SINGLETONS[x + 1];
                if (scripts[name] != null)
                {
                    if ((double)scripts[name] >= 0.4)
                    {
                        return code;
                    }
                }
            }

            // Extended Latin
            if ((double)scripts["Latin-1 Supplement"] + (double)scripts["Latin Extended-A"] + 
(double)scripts["IPA Extensions"] >= 0.4)
            {
                string latin_lang = Check(text, EXTENDED_LATIN);
                if (latin_lang == "pt")
                {
                    return Check(text, PT);
                } else {
                    return latin_lang;
                }
            }

            if ((double)scripts["Basic Latin"] >= 0.15) {
                return Check(text, ALL_LATIN);
            }

            return null; //give up, no match
        }

        private string Check(string sample, string[] langs)
        {

            if (sample.Length < MIN_LENGTH) {
                return null;
            }

            Hashtable scores = new Hashtable();
            TrigramModel model = CreateOrderedModel(sample);
            int lowestScore = Int32.MaxValue;
            string lowestScoreLanCode = null;
            for (int i = 0; i < langs.Length; i++)
            {

                string lkey = langs[i].ToLower();

                if (langStat.Models[lkey] == null) continue;//next please, no known model for
this
                TrigramModel known_model = (TrigramModel)langStat.Models[lkey];

                int dist = Distance(model, known_model);
                scores[lkey] = dist;
                if (dist< lowestScore)
                {
                    lowestScore = dist;
                    lowestScoreLanCode = lkey;
                }

            }

            return lowestScoreLanCode;
        }

        private TrigramModel CreateOrderedModel(string content)
        {
            // Create a list of trigrams in content sorted by frequency.
            Hashtable trigrams = new Hashtable();
            content = content.ToLower();

            for (int i = 0; i <content.Length - 2; i++)
            {
                string trigramKey = "" + content[i] + content[i + 1] + content[i + 2];
                if (trigrams[trigramKey] == null) {
                    trigrams[trigramKey] = 1;
                } else {
                    trigrams[trigramKey] = ((int)trigrams[trigramKey]) + 1;
                }
            }

            return new TrigramModel(trigrams);
        }

        private int Distance(TrigramModel model, TrigramModel known_model)
        {
            // Calculate the distance to the known model.
            int dist = 0;

            for (int i = 0; i < model.Trigrams.Length; i++)
            {
                if (known_model.HasTrigram(model.Trigrams[i].t))
                {
                    dist += Math.Abs(model.Trigrams[i].score -
known_model.GetScore(model.Trigrams[i].t));
                } else {
                    dist += MAX_GRAMS;
                }
            }

            return dist;
        }

    }
}

using System;
using System.Collections;

namespace LiteMiner.classes
{
    class LanguageStatistics
    {
        private Hashtable tmodels = new Hashtable();
        private Hashtable models = new Hashtable();

        public Hashtable Models
        {
            get { return models; }
        }

        public LanguageStatistics()
        {

            //language-dependent trigram distribution tables

            tmodels["af"] = new string[] {"ie ", " di", "die", "en ", "ing", "an ", " en",
"van", " va", "ng ", "te ", "n d", "ver", "er ", "e v", " ge", " be", "de ", " ve", "nde", "
in", " te", "le ", "der", "ers", "et ", "oor", " 'n" , "'n " , "at ", "eer", "ste", "ord",
"aar", "sie", " wa", "es ", "e s", "aan", " on", "is ", "in ", "e o", "rde", "e b", "asi",
"rin", "ond", "e w", "el ", " is", "and", "e e", "eid", "e d", "om ", "ke ", " om", "eri", "
wo", "e g", "r d", "ale", "wat", " vo", "id ", "it ", "rd ", " aa", "lik", " we", "t d", "
op", "e t", "ngs", "se ", "end", "uit", " st", " le", "ens", "ter", " re", "e a", "ies",
"wor", "g v", "sta", "n s", " na", " pr", "n o", " me", "al ", "of ", " vi", "erd", "lee", "e
k", " de", "ite", "erk", "ik ", "e r", "e p", "n v", "e i", "e n", "een", "eli", "wer", "
of", " da", "tel", "nie", "ike", "s e", "taa", "ge ", "vir", "hei", "ir ", "reg", "ede", "s
v", "ur ", "pro", "ele", "ion", "wet", "e l", " mo", "e m", "daa", "sio", "s d", " he", "
to", "ent", "ard", "nge", " oo", "eur", "lle", "ien", "n b", "eke", "lin", "raa", " ni",
"ont", "bes", "rdi", "voo", "ns ", "n a", "del", "dig", "nas", " sa", " gr", "nis", "kom", "
ui", "men", "op ", "ins", "ona", "ere", "s o", " so", "n g", "ig ", "moe", " ko", "rs ",
"ges", "nal", "vol", "e h", "geb", "rui", "ang", "ige", "oet", "ar ", "wys", "lig", "as ", "n
w", " as", "met", "gs ", "deu", "t v", "aal", "erw", "dit", "ken", "sse", "kel", " hu",
"ewe", "din", "n t", " se", "est", "ika", "n p", "ntw", "t i", "eni", " ka", "n e", "doe",
"ali", "eme", "gro", "nte", " ho", "nsi", "gen", "ier", "gew", "n h", "or ", " ma", "ind",
"ne ", "ek ", "aat", "n '", " sk", "ide", " ta", "dat", "ska", "ger", "soo", "n k", "s i", "
af", "tee", "nd ", "eel", "hul", "nee", "woo", "rik", "d v", "n m", "re ", "art", "ebr",
"lan", "kke", "ron", "aam", "tre", "str", "kan", "ree", "lei", "t o", "gra", "het", "evo",
"tan", "den", "ist", " do", "bru", "toe", "olg", "rsk", "uik", "rwy", "min", "lge", "g e", "g
o", "nst", "r v", "gte", "waa", "we ", "ans", "esi", "ese", "voe", "epa", "gel", " hi",
"vin", "nse", "s w", "s t", "tei", "eit", "pre"};
            tmodels["ar"] = new string[] {" ال", "الع", "لعر", "عرا", "راق", " في", "في ",
"ين ", "ية ", "ن ا", "الم", "ات ", "من ", "ي ا", " من", "لأ ا", "ة ا", "اق ", " وا", "اء ",
"لإ ا", " أن", "وال", "ما ", " عل", "لى ", "ت ا", "ون ", "هم ", "اقي", "ام ", "ل ا", "أن ",
"م ا", "الت", "لا ", "لا ا", "ان ", "ها ", "ال ", "ة و", "ا ا", "رها", "م لا", "يين", " ول",
"م لأ", "نا ", "على", "ن ي", "الب", "اد ", "الق", "د ا", "ذا ", "ه ا", " با", "الد", "ب ا",
"مري", "لم ", " إن", " لل", "لا س", "أمر", "ريك", "مة ", "ى ا", "ا ي", " عن", " هذ", "ء ا",
"ر ا", "لكن", "قتل", "إسل", "الح", "وا ", " إل", "ا أ", "بال", "ن م", "الس", "رة ", "س لإ",
"ن و", "هاب", "ي و", "ير ", " كا", "لة ", "يات", " لا ", "انت", "ن أ", "يكي", "الر", "الو",
"ة ف", "دة ", "الج", "قي ", "وي ", "الذ", "الش", "امي", "اني", "ذه ", "عن ", "لما", "هذه",
"ول ", "اف ", "اوي", "بري", "ة ل", " أم", " لم", " ما", "يد ", " أي", "إره", "ع ا", "عمل",
"لا و", "إلى", "نة ", "هي ", " بل", " به", " له", " وي", "ا ك", "اذا", "اع ", "ت م", "تخا",
"خاب", "ر م", "لمت", "مسل", "ى أ", "يست", "يطا", " لأ ", " لي", "أمن", "است", "بعض", "ة ت",
"ري ", "صدا", "ق و", "قول", "مد ", "نتخ", "نفس", "نها", "هنا", "أعم", "أنه", "ائن", "لآ ا",
"الك", "حة ", "د م", "ر ع", "ربي"};
            tmodels["az"] = new string[] {"lər", "in ", "ın ", "lar", "da ", "an ", "ir ",
"də ", "ki ", " bi", "ən ", "əri", "arı", "ər ", "dir", "nda", " ki", "rin", "nın", "əsi",
"ini", " ed", " qa", " tə", " ba", " ol", "ası", "ilə", "rın", " ya", "anı", " və", "ndə",
"ni ", "ara", "ını", "ınd", " bu", "si ", "ib ", "aq ", "dən", "iya", "nə ", "rə ", "n b",
"sın", "və ", "iri", "lə ", "nin", "əli", " de", " mü", "ai ", "bar", "cti", "di ", "eri",
"gör", "gün", "gəl", "hbə", "ihə", "iki", "isi", "lin", "mai", "maq", "n k", "n t", "n v",
"onu", "qan", "qəz", "tə ", "xal", "yib", "yih", "zet", "zır", "ıb ", "ə m", "əze", " br", "
in", " ir", " pr", " ta", " to", " üç", "a o", "ali", "ani", "anl", "aql", "azi", "bri"};
            tmodels["bg"] = new string[] {"на ", " на", "то ", " пр", " за", "та ", " по",
"ите", "те ", "а п", "а с", " от", "за ", "ата", "ия ", " в ", "е н", " да", "а н", " се", "
ко", "да ", "от ", "ани", "пре", "не ", "ени", "о н", "ни ", "се ", " и ", "но ", "ане",
"ето", "а в", "ва ", "ван", "е п", "а о", "ото", "ран", "ат ", "ред", " не", "а д", "и п", "
до", "про", " съ", "ли ", "при", "ния", "ски", "тел", "а и", "по ", "ри ", " е ", " ка",
"ира", "кат", "ние", "нит", "е з", "и с", "о с", "ост", "че ", " ра", "ист", "о п", " из", "
са", "е д", "ини", "ки ", "мин", " ми", "а б", "ава", "е в", "ие ", "пол", "ств", "т н", "
въ", " ст", " то", "аза", "е о", "ов ", "ст ", "ът ", "и н", "ият", "нат", "ра ", " бъ", "
че", "алн", "е с", "ен ", "ест", "и д", "лен", "нис", "о о", "ови", " об", " сл", "а р",
"ато", "кон", "нос", "ров", "ще сед", "слу", "т и", "тов", "ува", "циа", "чес", "я з", " во",
" ил", " ск", " тр", " це", "ами", "ари", "бат", "би ", "бра", "бъд"};
            tmodels["ca"] = new string[] {" de", "es ", "de ", "la ", " la", "el ", "que", "
el", " co", "ent", "s d", " qu", " i ", "en ", "er ", " a ", "ls ", "nt ", " pe", "e l", "a
d", " en", "per", "ci ", "ar ", "ue ", "al ", " se", "est", "at ", " es", "ts ", " s ", "
pr", "aci", " un", "res", "men", "s e", "del", "s a", "s p", " re", "les", " l'", "na ", "a
l", " ca", " d'", "els", "a p", "ia ", "ns ", "con", " le", "tat", "a c", "i d", "a a", "ra
", "a e", " no", "ant", " al", "t d", "s i", " di", "ta ", "re ", "a s", "com", "s c", "ita",
"ons", "sta", "ica", " po", "r a", " in", "pro", "tre", " pa", "ues", "amb", "ion", "des",
"un ", " ma", "da ", "s s", "a "nal", " ri", "qua", "t l", " do", "t s", "rma", "ual", "i s",
"s f", "n p", "s v", "te ", "t i", " ba", "cte", "tam", "man", "l t", "ial", " fa", "ic ", "
ve", "ble", "a n", "all", "tza", "ies", " s'", "le ", "omp", "r c", " nc", "rti", "it ",
"rre", "fic", "any", "on ", " sa", "r p", "tur"};
            tmodels["ceb"] = new string[] {"ng ", "sa ", " sa", "ang", "ga ", "nga", " ka", "
ng", "an ", " an", " na", " ma", " ni", "a s", "a n", "on ", " pa", " si", "a k", "a m", "
ba", "ong", "a i", "ila", " mg", "mga", "a p", "iya", "a a", "ay ", "ka ", "ala", "ing", "g
m", "n s", "g n", "lan", " gi", "na ", "ni ", "o s", "g p", "n n", " da", "ag ", "pag", "g
s", "yan", "ayo", "o n", "si ", " mo", "a b", "g a", "ail", "g b", "han", "a d", "asu",
"nag", "ya ", "man", "ne ", "pan", "kon", " il", " la", "aka", "ako", "ana", "bas", "ko ",
"od ", "yo ", " di", " ko", " ug", "a u", "g k", "kan", "la ", "len", "sur", "ug ", " ai",
"apa", "aw ", "d s", "g d", "g g", "ile", "nin", " iy", " su", "ene", "og ", "ot ", "aba",
"aha", "as ", "imo", " ki", "a t", "aga", "ban", "ero", "nan", "o k", "ran", "ron", "sil",
"una", "usa", " us", "a g", "ahi", "ani", "er ", "ha ", "i a", "rer", "yon", " pu", "ini",
"nak", "ro ", "to ", "ure", " ed", " og", " wa", "ili", "mo ", "n a", "nd ", "o a", " ad", "
```

```
du", " pr", "aro", "i s", "ma ", "n m", "ulo", "und", " ta", "ara", "asa", "ato", "awa",
"dmu", "e n", "edm", "ina", "mak", "mun", "niy", "san", "wa ", " tu", " un", "a l", "bay",
"iga", "ika", "ita", "kin", "lis", "may", "os ", " ar", "ad ", "ali", "ama", "ers", "ipa",
"isa", "mao", "nim", "t s", "tin", " ak", " ap", " hi", "abo", "agp", "ano", "ata", "g i",
"gan", "gka", "gpa", "i m", "iha", "k s", "law", "or ", "rs ", "siy", "tag", " al", " at", "
ha", " hu", " im", "a h", "bu ", "e s", "gma", "kas", "lag", "mon", "nah", "ngo", "r s", "ra
", "sab", "sam", "sul", "uba", "uha", " lo", " re", "ada", "aki", "aya", "bah", "ce ", "d n",
"lab", "pa ", "pak", "s n", "s s", "tan", "taw", "te ", "uma", "ura", " in", " lu", "a c",
"abi", "at ", "awo", "bat", "dal", "dla", "ele", "g t", "g u", "gay", "go ", "hab", "hin", "i
e", "i n", "kab", "kap", "lay", "lin", "nil", "pam", "pas", "pro", "pul", "ta ", "ton",
"uga", "ugm", "unt", " co", " gu", " mi", " pi", " ti", "a o", "abu", "adl", "ado", "agh",
"agk", "ao ", "art", "bal", "cit", "di ", "dto", "dun", "ent", "g e", "gon", "gug", "ia ",
"iba", "ice", "in ", "inu", "it ", "kaa"};
            tmodels["cs"] = new string[] {" pr", " po", "ní ", "pro", " na", "na ", " př",
"ch ", " je", " ne", "že ", " že", " se", " do", " ro", " st", " v ", " ve", "pře", "se ",
"ho ", "sta", " to", " vy", " za", "ou ", " a ", "to ", " by", "la ", "ce ", "e v", "ist",
"le ", "pod", "í p", " vl", "e n", "e s", "je ", "ké ", "by ", "em ", "ých", " od", "ova",
"řed", "dy ", "ení", "kon", "li ", "ně ", "str", " zá", "ve ", " ka", " sv", "e p", "it ",
"lád", "oho", "rov", "roz", "ter", "vlá", "ím ", " ko", "hod", "nis", "pří", "ský", " mi", "
ob", " so", "a p", "ali", "bud", "edn", "ick", "kte", "ku ", "o s", "al ", "ci ", "e t", "il
", "ny ", "né ", "odl", "ová", "rot", "sou", "ání", " bu", " mo", " o ", "ast", "byl", "de ",
"ek ", "ost", " mí", " ta", "es ", "jed", "ky ", "las", "m p", "nes", "ním", "ran", "rem",
"ros", "ého", " de", " kt", " ni", " si", " vý", "at ", "jí ", "ký ", "mi ", "pre", "tak",
"tan", "y v", "řek", " ch", " li", " ná", " pa", " ře", "da ", "dle", "dne", "i p", "i v",
"ly ", "min", "o n", "o v", "pol", "tra", "val", "vní", "ích", "ý p", "řej", " ce", " kd",
"nos", "o o", "o p", "oce", "ody", "ohl", "oli", "ovo", "pla", "poč", "prá", "ra ", "rit",
"rod", "ry ", "sd ", "sko", "ssd", "tel", "u s", "vat", "veř", "vit", "vla", "y p", "áln",
"čss", "šen", " al"};
vo", "bou", "sam", "tim", "vot", "abo", "ant", "ds ", "ial", "ine", "man", "men", " or", "
po", "amp", "can", "der", "e l", "les", "ny ", "ot ", "rec", "tes", "tho", "ica", "ild", "ir
", "nde", "ose", "ous", "pre", "ste", "era", "per", "r o", "red", "rie", " bo", " le", "ali",
"ars", "ore", "ric", "s m", "str", " fa", "ess", "ie ", "ist", "lat", "uri"};
            tmodels["es"] = new string[] {" de", "de ", " la", "os ", "la ", "el ", "es ", "
qu", " co", "e l", "as ", "que", " el", "ue ", "en ", "ent", " en", " se", "nte", "res",
"con", "est", " es", "s d", " lo", " pr", "los", " y ", "do ", "ón ", "ión", " un", "ció",
"del", "o d", " po", "a d", "aci", "sta", "te ", "ado", "pre", "to ", "par", "a e", "a l",
"ra ", "al ", "e e", "se ", "pro", "ar ", "ia ", "o e", " re", "ida", "dad", "tra", "por", "s
p", " a ", "a p", "ara", "cia", " pa", "com", "no ", " di", " in", "ien", "n l", "ad ",
"ant", "e s", "men", "a c", "on ", "un ", "las", "nci", " tr", "cio", "ier", "nto", "tiv", "n
d", "n e", "or ", "s c", "enc", "ern", "io ", "a s", "ici", "s e", " ma", "dos", "e a", "e
c", "emp", "ica", "ivo", "l p", "n c", "r e", "ta ", "ter", "e d", "esa", "ez ", "mpr", "o
a", "s a", " ca", " su", "ion", " cu", " ju", "an ", "da ", "ene", "ero", "na ", "rec", "ro
", "tar", " al", " an", "bie", "e p", "er ", "l c", "n p", "omp", "ten", " em", "ist", "nes",
"nta", "o ", "cti", "cue", "div", "duc", "ens", "eti", "imi", "ini", "lec", "o q", "oce",
"ort", "ral", "rma", "roc", "rod"};
            tmodels["et"] = new string[] {"st ", " ka", "on ", "ja ", " va", " on", " ja", "
ko", "se ", "ast", "le ", "es ", "as ", "is ", "ud ", " sa", "da ", "ga ", " ta", "aja",
"sta", " ku", " pe", "a k", "est", "ist", "ks ", "ta ", "al ", "ava", "id ", "saa", "mis",
"te ", "val", " et", "nud", " te", "inn", " se", " tu", "a v", "alu", "e k", "ise", "lu ",
"ma ", "mes", " mi", "et ", "iku", "lin", "ad ", "el ", "ime", "ne ", "nna", " ha", " in", "
ke", " võ", "a s", "a t", "ab ", "e s", "esi", " la", " li", "e v", "eks", "ema", "las",
"les", "rju", "tle", "tsi", "tus", "upa", "use", "ust", "var", " lä", "ali", "arj", "de ",
"ete", "i t", "iga", "ilm", "kui", "li ", "tul", " ei", " me", " sõ", "aal", "ata", "dus",
"ei ", "nik", "pea", "s k", "s o", "sal", "sõn", "ter", "ul ", "või", " el", " ne", "a j",
"ate", "end", "i k", "ita", ", "ima", "its", "kka", "kuh", "l k", "lat", "maj", "ndu", "ni ",
"nii", "oma", "ool", "rso", "ru ", "rva", "s t", "sek", "son", "ste", "t m", "taj", "tam",
"ude", "uho", "vai", " ag", " os", " pa", " re"};
            tmodels["eu"] = new string[] {"en ", "an ", "eta", "ta ", " et", "iza", "n e",
"ko ", "ide", " ba", "a e", "giz", " es", " gi", "arr", "bid", "ren", "rri", "are", "la ",
"sku", " be", "asu", "esk", "sun", "tas", " iz", "ean", "eko", "ela", "ik ", "kub", "n a", "n
i", "tza", "ubi", "za ", "zan", " er", "a b", "ask", "era", "n b", "rre", "ten", "tze", "
as", " ko", "a a", "a g", "ald", "ani", "de ", "dee", "ea ", "ek ", "kat", "kon", "n d",
"ont", "uan", " du", " na", "ata", "egi", "est", "k e", "nik", "ntu", "ntz", "ska", "tua", "
de", " di", " ez", " he", "a d", "a k", "ak ", "aki", "ako", "art", "atu", "azi", "bat",
"ber", "itz", "kun", "n h", "o b", "ria", "rte", "tat", "une", "zar", " al", " ar", " ha",
"aku", "atz", "bai", "dar", "dea", "del", "een", "ema", "err", "iak", "iar", "in ", "ina",
"kia", "nar", "naz", "nea", "o e", "orr", "ra ", "ste", "tek", "zak", "zek", "zio", " da", "
em", " hi", " ho", " ma", " oi", "agu", "ate", "aur", "bes", "din", "dir", "dut", "ert", "ez
", "ezi", "har", "her", "hit", "ia ", "ien", "ika", "io ", "ire", "ite", "k b", "k g", "kid",
"kor", "lda", "n o", "nko", "o a", "oin", "ori", "rak", "rea", "rie", "rik", "rra", "tan",
"tea", "tu ", "una", "und", "unt", "urr", "ute", "z e", "zko", " au", " eg", " gu", " ir", "
ki", " or", "a h", "a j", "abe", "agi", "ai ", "ail", "ait", "ape", " eha", "elk", "enb",
"ete", "eti", "gab", "gin", "go ", "gus", "gut", "guz", "hau", "ibe", "inb", "ine", "ioa",
"iru", "iur", "izi", "izk", "izo"};

            tmodels["fr"] = new string[]  {"es ", " de", "de ", " le", "ent", "le ", "nt ",
"la ", "s d", " la", "ion", "on ", "re ", " pa", "e l", "e d", " l'" , "e p", " co", " pr",
"tio", "ns ", " en", "ne ", "que", "r l", "les", "ur ", "en ", "ati", "ue ", " po", " d'" ,
"par", " a ", "et ", "it ", " qu", "men", "ons", "te ", " et", "t d", " re", "des", " un",
"ie ", "s l", " su", "pou", " au", " à ", "con", "er ", " no", "ait", "e c", "se ", "té ",
"du ", " du", " dé", "ce ", "e e", "is ", "n d", "s a", " so", "e r", "e s", "our", "res",
"ssi", "eur", " se", "eme", "est", "us ", "sur", "ant", "iqu", "s p", "une", "uss", "l'a" ,
"pro", "ter", "tre", "end", "rs ", " ce", "e a", "t p", "un ", " ma", " ru", " ré", "ous",
"ris", "rus", "sse", "ans", "ar ", "com", "e m", "ire", "nce", "nte", "t l", " av", " mo", "
te", "il ", "me ", "ont", "ten", "a p", "dan", "pas", "qui", "s e", "s s", " in", "ist",
"lle", "nou", "pré", "'un" , "air", "d'a" , "ir ", "n e", "rop", "ts ", " da", "a s", "as ",
"au ", "den", "mai", "mis", "ori", "out", "rme", "sio", "tte", "ux ", "a d", "ien", "n a",
"ntr", "omm", "ort", "ouv", "s c", "son", "tes", "ver", "ère", " il", " m ", " sa", " ve", "a
r", "ais", "ava", "di ", "n p", "sti", "ven", " mi", "ain", "enc", "for", "ité", "lar",
"oir", "rem", "ren", "rro", "rés", "sie", "t a", "tur", " pe", " to", "d'u" , "ell", "err",
"ers", "ide", "ine", "iss", "mes", "por", "ran", "sit", "st ", "t r", "uti", "vai", "é l",
"ési", " di", " n'" , " ét", "a c", "ass", "e t", "in ", "nde", "pre", "rat", "s m", "ste",
"tai", "tch", "ui ", "uro", "ès ", " es", " fo", " tr", "'ad" , "app", "aux", "e à", "ett",
"iti", "lit", "nal", "opé", "r d", "ra ", "rai", "ror", "s r", "tat", "uté", "à l", " af",
"anc", "ara", "art", "bre", "ché", "dre", "e f", "ens", "lem", "n r", "n t", "ndr", "nne",
"onn", "pos", "s t", "tiq", "ure", " tu", "ale", "and", "ave", "cla", "cou", "e n", "emb",
"ins", "jou", "mme", "rie", "rès", "sem", "str", "t i", "ues", "uni", "uve", "é d", "ée ", "
ch", " do", " eu", " fa", " lo", " ne", " ra", "arl", "att", "ec ", "ica", "l a", "l'o" ,
"l'é" , "mmi", "nta", "orm", "ou ", "r u", "rle"};
            tmodels["ha"] = new string[]  {" da", "da ", "in ", "an ", "ya ", " wa", " ya",
"na ", "ar ", "a d", " ma", "wa ", "a a", "a k", "a s", " ta", "wan", " a ", " ba", " ka",
"ta ", "a y", "n d", " ha", " na", " su", " sa", "kin", "sa ", "ata", " ko", "a t", "su ", "
ga", "ai ", " sh", "a m", "uwa", "iya", "ma ", "a w", "asa", "yan", "ka ", "ani", "shi", "a
b", "a h", "a c", "ama", "ba ", "nan", "n a", " mu", "ana", " yi", "a g", " za", "i d", "
ku", "aka", "yi ", "n k", "ann", "ke ", "tar", " ci", "iki", "n s", "ko ", " ra", "ki ", "ne
", "a z", "mat", "hak", "nin", "e d", "nna", "uma", "nda", "a n", "ada", "cik", "ni ", "rin",
"una", "ara", "kum", "akk", " ce", " du", "man", "n y", "nci", "sar", "aki", "awa", "ci ",
"kan", "kar", "ari", "n m", "and", "hi ", "n t", "ga ", "owa", "ash", "kam", "dan", "ewa",
"nsa", "ali", "ami", " ab", " do", "anc", "n r", "aya", "i n", "sun", "uka", " al", " ne",
"a'a" , "cew", "cin", "mas", "tak", "un ", "aba", "kow", "a r", "ra ", " ja", " ƙa", "en ",
"r d", "sam", "tsa", " ru", "ce ", "i a", "abi", "ida", "mut", "n g", "n j", "san", "a ƙ",
"har", "on ", "i m", "suk", " ak", " ji", "yar", "'ya" , "kwa", "min", " 'y" , "ane", "ban",
"ins", "ruw", "i k", "n h", " ad", "ake", "n w", "sha", "utu", " ya", "bay", "tan", "yan",
"bin", "duk", "e m", "n n", "oka", "yin", "ɗan", " fa", "a i", "kki", "re ", "za ", "ala",
"asu", "han", "i y", "mar", "ran", "ƙas", "add", "ars", "gab", "ira", "mma", "u d", " ts",
"abb", "abu", "aga", "gar", "n b", " ɗa", "aci", "aik", "am ", "dun", "e s", "i b", "i w",
"kas", "kok", "wam", " am", "amf", "bba", "din", "fan", "gwa", "i s", "wat", "ano", "are",
"dai", "iri", "ma'" , " la", "all", "dam", "ika", "mi ", "she", "tum", "uni", " an", " ai", "
ke", " ki", "dag", "mai", "mfa", "no ", "nsu", "o d", "sak", "um ", " bi", " gw", " kw",
"jam", "yya", "a j", "fa ", "uta", " hu", "'a " , "ans", "aɗa", "dda", "hin", "niy", "r s",
"bat", "dar", "gan", "i t", "nta", "oki", "omi", "sal", "a l", "kac", "lla", "wad", "war",
"amm", "dom", "r m", "ras", "sai", " lo", "ats", "hal", "kat", "li ", "lok", "n c", "nar",
"tin", "afa", "bub", "i g", "isa", "mak"};
```

```
            tmodels["haw"] = new string[]  {" ka", "na ", " o ", "ka ", " ma", " a ", " la",
"a i", "a m", " i ", "la ", "ana", "ai ", "ia ", "a o", "a k", "a h", "o k", " ke", "a a", "i
k", " ho", " ia", "ua ", " na", " me", "e k", "e a", "au ", "ke ", "ma ", "mai", "aku", "
ak", "ahi", " ha", " ko", " e ", "a l", " no", "me ", "ku ", "aka", "kan", "no ", "i a", "ho
", "ou ", " ai", "i o", "a p", "o l", "o a", "ama", "a n", " an", "i m", "han", "i i", "iho",
"kou", "ne ", " ih", "o i", "iki", "ona", "hoo", "le ", "e h", " he", "ina", " wa", "ea ",
"ako", "u i", "kah", "oe ", "i l", "u a", " pa", "hoi", "e i", "era", "ko ", "u m", "kua",
"mak", "oi ", "kai", "i n", "a e", "hin", "ane", " ol", "i h", "mea", "wah", "lak", "e m", "o
n", "u l", "ika", "ki ", "a w", "mal", "hi ", "e n", "u o", "hik", " ku", "e l", "ele", "ra
", "ber", "ine", "abe", "ain", "ala", "lo ", " po", "kon", " ab", "ole", "he ", "pau", "mah",
"va ", "ela", "kau", "nak", " oe", "kei", "oia", " ie", "ram", " oi", "oa ", "eho", "hov",
"ieh", "ova", " ua", "una", "ara", "o s", "awa", "o o", "nau", "u n", "wa ", "wai", "hel", "
ae", " al", "ae ", "ta ", "aik", " hi", "ale", "ila", "lel", "ali", "eik", "olo", "onu", "
lo", "aua", "e o", "ola", "hon", "mam", "nan", " au", "aha", "lau", "nua", "oho", "oma", "
ao", "ii ", "alu", "ima", "mau", "ike", "apa", "elo", "lii", "poe", "aia", "noa", " in", "o
m", "oka", "'u " , "aho", "ei ", "eka", "ha ", "lu ", "nei", "hol", "ino", "o e", "ema",
"iwa", "olu", "ada", "naa", "pa ", "u k", "ewa", "hua", "lam", "lua", "o h", "ook", "u h", "
li", "ahu", "amu", "ui ", " il", " mo", " se", "eia", "law", " hu", " ik", "ail", "e p", "li
", "lun", "uli", "io ", "kik", "noh", "u e", " sa", "aaw", "awe", "ena", "hal", "kol", "lan",
" le", " ne", "a'u" , "ilo", "kap", "oko", "sa ", " pe", "hop", "loa", "ope", "pe ", " ad", "
pu", "ahe", "aol", "ia'" , "lai", "loh", "na'" , "oom", "aau", "eri", "kul", "we ", "ake",
"kek", "laa", "ri ", "iku", "kak", "lim", "nah", "ner", "nui", "ono", "a u", "dam", "kum",
"lok", "mua", "uma", "wal", "wi ", "'i " , "a'i" , "aan", "alo", "eta", "mu ", "ohe", "u p",
"ula", "uwa", " nu", "amo"};
            tmodels["hi"] = new string[]  {"ें ", " है", "में", " मे", "ने ", "की ", "के ", "है ",
" के", " की", " को", "ों ", "को ", "ा ह", " का", "से ", "ा क", "े क", "ं क", "या ", " कि", "
से", "का ", "ी क", " ने", " और", "और ", "ना ", "कि ", "भी ", "ी स", " जा", " पर", "ार ", "
कर", "ी ह", " हो", "ही ", "िया", " इस", " रह", "र क", "ुना", "ता ", "ान ", "े स", " भी", "
रा", "े ह", " चु", " पा", "पर ", "चुन", "नाव", " कह", "प्र", " भा", "राज", "हैं", "ा स", "ै क",
"ैं ", "नी ", "ल क", "ीं ", "़ी ", "था ", "री ", "ाव ", "े ब", " प्", "क्ष", "पा ", "ले ", "
दे", "ला ", "हा ", "ाजप", " था", " नह", "इस ", "कर ", "जपा", "नही", "भाज", "यों", "र स", "हीं",
" अम", " बा", " मा", " वि", "रीक", "िए ", "े प", "्या", " ही", "ं म", "कार", "ा ज", "े ल", "
ता", " दि", " सा", " हम", "ा न", "ा म", "ाक़", "्ता", " एक", " सं", " स्", "अमर", "क़ी",
"ताज", "मरी", "स्थ", "ा थ", "ार्", " हु", "इरा", "एक ", "न क", "र म", "राक", "ी ज", "ी न", "
इर", " उन", " पह", "कहा", "ते ", "े अ", " तो", " सु", "ति ", "ती ", "तो ", "मिल", "िक ",
"ियो", "्रे", " अप", " फ़", " लि", " लो", " सम", "म क", "र्ट", "हो ", "ा च", "ाई ", "ाने",
"िन ", "्य ", " उस", " क़", " सक", " सै", "ं प", "ं ह", "गी ", "त क", "मान", "र न", "ष्ट",
"स क", "स्त", "ाँ ", "ी ब", "ी म", "्री", " दो", " मि", " मु", " ले", " शा", "ं स", "ज़ा",
"त्र", "थी ", "लिए", "सी ", "़ा ", "़ार", "ांग", "े द", "े म", "्व ", " ना", " बन", "ंग्",
"कां", "गा ", "ग्र", "जा ", "ज्य", "दी ", "न म", "पार", "भा ", "रही", "रे ", "रेस", "ली ", "सभा", "ा
र", "ाल ", "ी अ", "ीकी", "े त", "ेश ", " अं", " तक", " या", "ई ह", "करन", "तक ", "देश",
"वर्", "ाया", "ी भ", "ेस ", "्ष ", " गय", " जि", " थी", " बड", " यह", " वा", "ंतर", "अंत",
"क़ ", "गया", "टी ", "निक", "न्ह", "पहल", "बड़", "मार", "र प", "रने", "ाज़", "ि इ", "ी र", "े
ज", "े व", "्ट ", "्टी", " अब", " लग", " वर", " सी", "ं भ", "उन्", "क क", "किय", "देख",
"पूर", "फ़्", "यह ", "यान", "रिक", "रिय", "र्ड", "लेक", "सकत", "हों", "होग", "ा अ", "ा द", "ा प",
"ाद ", "ारा", "ित ", "ी त", "ी प", "ो क", "ो द", " ते", " नि", " सर", " हा", "ं द",
"अपन", "जान", "त म", "थित", "पनी", "महल", "र ह", "लोग", "व क", "हना", "हल ", "हाँ", "ाज्",
"ाना", "िक्", "िस्"};
            tmodels["hr"] = new string[]  {"je ", " na", " pr", " po", "na ", " je", " za",
"ije", "ne ", " i ", "ti ", "da ", " ko", " ne", "li ", " bi", " da", " u ", "ma ", "mo ", "a
n", "ih ", "za ", "a s", "ko ", "i s", "a p", "koj", "pro", "ju ", "se ", " go", "ost", "to
```

```
", "va ", " do", " to", "e n", "i p", " od", " ra", "no ", "ako", "ka ", "ni ", " ka", " se",
" mo", " st", "i n", "ima", "ja ", "pri", "vat", "sta", " su", "ati", "e p", "ta ", "tsk", "e
i", "nij", " tr", "cij", "jen", "nos", "o s", " iz", "om ", "tro", "ili", "iti", "pos", "
al", "a i", "a o", "e s", "ija", "ini", "pre", "str", "la ", "og ", "ovo", " sv", "ekt",
"nje", "o p", "odi", "rva", " ni", "ali", "min", "rij", "a t", "a z", "ats", "iva", "o t",
"od ", "oje", "ra ", " hr", "a m", "a u", "hrv", "im ", "ke ", "o i", "ovi", "red", "riv",
"te ", "bi ", "e o", "god", "i d", "lek", "umi", "zvo", "din", "e u", "ene", "jed", "ji ",
"lje", "nog", "su ", " a ", " el", " mi", " o ", "a d", "alu", "ele", "i u", "izv", "ktr",
"lum", "o d", "ori", "rad", "sto", "a k", "anj", "ava", "e k", "men", "nic", "o j", "oj ",
"ove", "ski", "tvr", "una", "vor", " di", " no", " s ", " ta", " tv", "i i", "i o", "kak",
"roš", "sko", "vod", " sa", " će", "a b", "adi", "amo", "eni", "gov", "iju", "ku ", "o n",
"ora", "rav", "ruj", "smo", "tav", "tru", "u p", "ve ", " in", " pl", "aci", "bit", "de ",
"diš", "ema", "i m", "ika", "išt", "jer", "ki ", "mog", "nik", "nov", "nu ", "oji", "oli",
"pla", "pod", "st ", "sti", "tra", "tre", "vo ", " sm", " št", "dan", "e z", "i t", "io ",
"ist", "kon", "lo ", "stv", "u s", "uje", "ust", "će ", "ći ", "što", " dr", " im", " li",
"ada", "aft", "ani", "ao ", "ars", "ata", "e t", "emo", "i k", "ine", "jem", "kov", "lik",
"lji", "mje", "naf", "ner", "nih", "nja", "ogo", "oiz", "ome", "pot", "ran", "ri ", "roi",
"rtk", "ska", "ter", "u i", "u o", "vi ", "vrt", " me", " ug", "ak ", "ama", "drž", "e e", "e
g", "e m", "em ", "eme", "enj", "ent", "er ", "ere", "erg", "eur", "go ", "i b", "i z",
"jet", "ksi", "o u", "oda", "ona", "pra", "reb", "rem", "rop", "tri", "žav", " ci", " eu", "
re", " te", " uv", " ve", "aju", "an "};
            tmodels["hu"] = new string[]  {" a ", " az", " sz", "az ", " me", "en ", " el", "
ho", "ek ", "gy ", "tt ", "ett", "sze", " fe", "és ", " ki", "tet", " be", "et ", "ter", "
kö", " és", "hog", "meg", "ogy", "szt", "te ", "t a", "zet", "a m", "nek", "nt ", "ség",
"szá", "ak ", " va", "an ", "eze", "ra ", "ta ", " mi", "int", "köz", " is", "esz", "fel",
"min", "nak", "ors", "zer", " te", "a a", "a k", "is ", " cs", "ele", "er ", "men", "si ",
"tek", "ti ", " ne", "csa", "ent", "z e", "a t", "ala", "ere", "es ", "lom", "lte", "mon",
"ond", "rsz", "sza", "tte", "zág", "ány", " fo", " ma", "ai ", "ben", "el ", "ene", "ik ",
"jel", "tás", "áll", " ha", " le", " ál", "agy", "alá", "isz", "y a", "zte", "ás ", " al", "e
a", "egy", "ely", "for", "lat", "lt ", "n a", "oga", "on ", "re ", "st ", "ság", "t m", "án
", "ét ", "ült", " je", "gi ", "k a", "kül", "lam", "len", "lás", "más", "s k", "vez", "áso",
"özö", " ta", "a s", "a v", "asz", "atá", "ető", "kez", "let", "mag", "nem", "szé", "z m",
"át ", "éte", "ölt", " de", " gy", " ké", " mo", " vá", " ér", "a b", "a f", "ami", "at ",
"ato", "att", "bef", "dta", "gya", "hat", "i s", "las", "ndt", "rt ", "szo", "t k", "tár",
"tés", "van", "ásá", "ól ", " bé", " eg", " or", " pá", " pé", " ve", "ban", "eke", "ekü",
"elő", "erv", "ete", "fog", "i a", "kis", "lád", "nte", "nye", "nyi", "ok ", "omá", "os ",
"rán", "rás", "sal", "t e", "vál", "yar", "ágo", "ála", "ége", "ény", "ött", " tá", "adó",
"elh", "fej", "het", "hoz", "ill", "jár", "kés", "llo", "mi ", "ny ", "ont", "ren", "res",
"rin", "s a", "s e", "ssz", "zt ", " ez", " ka", " ke", " ko", " re", "a h", "a n", "den",
"dó ", "efo", "gad", "gat", "gye", "hel", "k e", "ket", "les", "mán", "nde", "nis", "ozz", "t
b", "t i", "t é", "tat", "tos", "val", "z o", "zak", "ád ", "ály", "ára", "ési", "ész", "
ak", " am", " es", " há", " ny", " tö", "aka", "art", "ató", "azt", "bbe", "ber", "ció",
"cso", "em ", "eti", "eté", "gal", "i t", "ini", "ist", "ja ", "ker", "ki ", "kor", "koz", "l
é", "ljá", "lye", "n v", "ni ", "pál", "ror", "ról", "rül", "s c", "s p", "s s", "s v",
"sok", "t j", "t t", "tar", "tel", "vat"};
            tmodels["id"] = new string[]  {"an ", " me", "kan", "ang", "ng ", " pe", "men", "
di", " ke", " da", " se", "eng", " be", "nga", "nya", " te", "ah ", "ber", "aka", " ya",
"dan", "di ", "yan", "n p", "per", "a m", "ita", " pa", "da ", "ata", "ada", "ya ", "ta ", "
in", "ala", "eri", "ia ", "a d", "n k", "am ", "ga ", "at ", "era", "n d", "ter", " ka", "a
p", "ari", "emb", "n m", "ri ", " ba", "aan", "ak ", "ra ", " it", "ara", "ela", "ni ",
"ali", "ran", "ar ", "eru", "lah", "a b", "asi", "awa", "eba", "gan", "n b", " ha", "ini",
"mer", " la", " mi", "and", "ena", "wan", " sa", "aha", "lam", "n i", "nda", " wa", "a i",
"dua", "g m", "mi ", "n a", "rus", "tel", "yak", " an", "dal", "h d", "i s", "ing", "min",
"ngg", "tak", "ami", "beb", "den", "gat", "ian", "ih ", "pad", "rga", "san", "ua ", " de", "a
t", "arg", "dar", "elu", "har", "i k", "i m", "i p", "ika", "in ", "iny", "itu", "mba", "n
t", "ntu", "pan", "pen", "sah", "tan", "tu ", "a k", "ban", "edu", "eka", "g d", "ka ",
"ker", "nde", "nta", "ora", "usa", " du", " ma", "a s", "ai ", "ant", "bas", "end", "i d",
"ira", "kam", "lan", "n s", "uli", "al ", "apa", "ere", "ert", "lia", "mem", "rka", "si ",
"tal", "ung", " ak", "a a", "a w", "ani", "ask", "ent", "gar", "haa", "i i", "isa", "ked",
"mbe", "ska", "tor", "uan", "uk ", "uka", " ad", " to", "asa", "aya", "bag", "dia", "dun",
"erj", "mas", "na ", "rek", "rit", "sih", "us ", " bi", "a h", "ama", "dib", "ers", "g s",
"han", "ik ", "kem", "ma ", "n l", "nit", "r b", "rja", "sa ", " ju", " or", " si", " ti", "a
```

```
y", "aga", "any", "as ", "cul", "eme", "emu", "eny", "epa", "erb", "erl", "gi ", "h m", "i
a", "kel", "li ", "mel", "nia", "opa", "rta", "sia", "tah", "ula", "un ", "unt", " at", "
bu", " pu", " ta", "agi", "alu", "amb", "bah", "bis", "er ", "i t", "ibe", "ir ", "ja ", "k
m", "kar", "lai", "lal", "lu ", "mpa", "ngk", "nja", "or ", "pa ", "pas", "pem", "rak",
"rik", "seb", "tam", "tem", "top", "tuk", "uni", "war", " al", " ga", " ge", " ir", " ja", "
mu", " na", " pr", " su", " un", "ad ", "adi", "akt", "ann", "apo", "bel", "bul", "der",
"ega", "eke", "ema", "emp", "ene", "enj", "esa"};
            tmodels["is"] = new string[]  {"að ", "um ", " að", "ir ", "ið ", "ur ", " ve", "
í ", "na ", " á ", " se", " er", " og", "ar ", "og ", "ver", " mi", "inn", "nn ", " fy", "er
", "fyr", " ek", " en", " ha", " he", "ekk", " st", "ki ", "st ", "ði ", " ba", " me", " vi",
"ig ", "rir", "yri", " um", "g f", "leg", "lei", "ns ", "ð s", " ei", " þa", "in ", "kki", "r
h", "r s", "egi", "ein", "ga ", "ing", "ra ", "sta", " va", " þe", "ann", "en ", "mil",
"sem", "tjó", "arð", "di ", "eit", "haf", "ill", "ins", "ist", "llj", "ndi", "r a", "r e",
"seg", "un ", "var", " bi", " el", " fo", " ge", " yf", "and", "aug", "bau", "big", "ega",
"eld", "erð", "fir", "foo", "gin", "itt", "n s", "ngi", "num", "od ", "ood", "sin", "ta ",
"tt ", "við", "yfi", "ð e", "ð f", " hr", " sé", " þv", "a e", "a á", "em ", "gi ", "i f",
"jar", "jór", "lja", "m e", "r á", "rei", "rst", "rða", "rði", "rðu", "stj", "und", "veg",
"ví ", "ð v", "það", "því", " fj", " ko", " sl", "eik", "end", "ert", "ess", "fjá", "fur",
"gir", "hús", "jár", "n e", "ri ", "tar", "ð þ", "ðar", "ður", "þes", " br", " hú", " kr", "
le", " up", "a s", "egg", "i s", "irt", "ja ", "kið", "len", "með", "mik", "n b", "nar",
"nir", "nun", "r f", "r v", "rið", "rt ", "sti", "t v", "ti ", "una", "upp", "ða ", "óna", "
al", " fr", " gr", "a v", "all", "an ", "da ", "eið", "eð ", "fa ", "fra", "g e", "ger",
"gið", "gt ", "han", "hef", "hel", "her", "hra", "i a", "i e", "i v", "i þ", "iki", "jón",
"jör", "ka ", "kró", "lík", "m h", "n a", "nga", "r l", "ram", "ru ", "ráð", "rón", "svo",
"vin", "í b", "í h", "ð h", "ð k", "ð m", "örð", " af", " fa", " lí", " rá", " sk", " sv", "
te", "a b", "a f", "a h", "a k", "a u", "afi", "agn", "arn", "ast", "ber", "efu", "enn",
"erb", "erg", "fi ", "g a", "gar", "iðs", "ker", "kke", "lan", "ljó", "llt", "ma ", "mið", "n
v", "n í", "nan", "nda", "ndu", "nið", "nna", "nnu", "nu ", "r o", "rbe", "rgi", "slö", "sé
", "t a", "t h", "til", "tin", "ugu", "vil", "ygg", "á s", "ð a", "ð b", "órn", "ögn", "öku",
" at", " fi", " fé", " ka", " ma", " no", " sa", " si", " ti", " ák", "a m", "a t", "a í", "a
þ", "afa", "afs", "ald", "arf"};
            tmodels["it"] = new string[]  {" di", "to ", "la ", " de", "di ", "no ", " co",
"re ", "ion", "e d", " e ", "le ", "del", "ne ", "ti ", "ell", " la", " un", "ni ", "i d",
"per", " pe", "ent", " in", "one", "he ", "ta ", "zio", "che", "o d", "a d", "na ", "ato", "e
s", " so", "i s", "lla", "a p", "li ", "te ", " al", " ch", "er ", " pa", " si", "con",
"sta", " pr", "a c", " se", "el ", "ia ", "si ", "e p", " da", "e i", "i p", "ont", "ano", "i
c", "all", "azi", "nte", "on ", "nti", "o s", " ri", "i a", "o a", "un ", " an", "are",
"ari", "e a", "i e", "ita", "men", "ri ", " ca", " il", " no", " po", "a s", "ant", "il ",
"in ", "a l", "ati", "cia", "e c", "ro ", "ann", "est", "gli", "tà ", " qu", "e l", "nta", "
a ", "com", "o c", "ra ", " le", " ne", "ali", "ere", "ist", " ma", " è ", "io ", "lle", "me
", "era", "ica", "ost", "pro", "tar", "una", " pi", "da ", "tat", " mi", "att", "ca ", "mo ",
"non", "par", "sti", " fa", " i ", " re", " su", "ess", "ini", "nto", "o l", "ssi", "tto", "a
e", "ame", "col", "ei ", "ma ", "o i", "za ", " st", "a a", "ale", "anc", "ani", "i m",
"ian", "o p", "oni", "sio", "tan", "tti", " lo", "i r", "oci", "oli", "ona", "ono", "tra", "
l ", "a r", "eri", "ett", "lo ", "nza", "que", "str", "ter", "tta", " ba", " li", " te",
"ass", "e f", "enz", "for", "nno", "olo", "ori", "res", "tor", " ci", " vo", "a i", "al ",
"chi", "e n", "lia", "pre", "ria", "uni", "ver", " sp", "imo", "l a", "l c", "ran", "sen",
"soc", "tic", " fi", " mo", "a n", "ce ", "dei", "ggi", "gio", "iti", "l s", "lit", "ll ",
"mon", "ola", "pac", "sim", "tit", "utt", "vol", " ar", " fo", " ha", " sa", "acc", "e r",
"ire", "man", "ntr", "rat", "sco", "tro", "tut", "va ", " do", " gi", " me", " sc", " tu", "
ve", " vi", "a m", "ber", "can", "cit", "i l", "ier", "ità", "lli", "min", "n p", "nat",
"nda", "o e", "o f", "o u", "ore", "oro", "ort", "sto", "ten", "tiv", "van", "art", "cco",
"ci ", "cos", "dal", "e v", "i i", "ila", "ino", "l p", "n c", "nit", "ole", "ome", "po ",
"rio", "sa ", " ce", " es", " tr", "a b", "and", "ata", "der", "ens", "ers", "gi ", "ial",
"ina", "itt", "izi", "lan", "lor", "mil"};
            tmodels["kk"] = new string[]  {"ан ", "ен ", "ың ", " қа", " ба", "ай ", "нда",
"ын ", " са", " ал", "ді ", "ары", "ды ", "ып ", " мұ", " бі", "асы", "да ", "най", " жа",
"мұн", "ста", "ған", "н б", "ұна", " бо", "ның", "ін ", "лар", "сын", " де", "аға", "тан", "
кө", "бір", "ер ", "мен", "аза", "ынд", "ыны", " ме", "анд", "ері", "бол", "дың", "қаз",
"аты", "сы ", "тын", "ғы ", " ке", "ар ", "зақ", "ық ", "ала", "алы", "аны", "ара", "ағы",
"ген", "тар", "тер", "тыр", "айд", "ард", "де ", "ға ", " қо", "бар", "ің ", "қан", " бе", "
қы", "ақс", "гер", "дан", "дар", "лық", "лға", "ына", "ір ", "ірі", "ғас", " та", "а б", "гі
```

```
", "еді", "еле", "йды", "н к", "н т", "ола", "рын", "іп ", "қст", "қта", "ң б", " ай", " ол",
" со", "айт", "дағ", "иге", "лер", "лып", "н а", "ік ", "ақт", "бағ", "кен", "н қ", "ны ",
"рге", "рға", "ыр ", " ар", "алғ", "аса", "бас", "бер", "ге ", "еті", "на ", "нде", "не ",
"ниг", "рды", "ры ", "сай", " ау", " кү", " ни", " от", " өз", "ауд", "еп ", "иял", "лты", "н
ж", "н о", "осы", "оты", "рып", "рі ", "тке", "ты ", "ы б", "ы ж", "ылы", "ысы", "і с",
"қар", " бұ", " да", " же", " тұ", " құ", "ады", "айл", "ап ", "ата", "ені", "йла", "н м", "н
с", "нды", "нді", "р м", "тай", "тін", "ы т", "ыс ", "інд", " би", "а ж", "ауы", "деп",
"дің", "еке", "ери", "йын", "кел", "лды", "ма ", "нан", "оны", "п ж", "п о", "р б", "рия",
"рла", "уда", "шыл", "ы а", "ықт", "і а", "і б", "із ", "ілі", "ң қ", " ас", " ек", " жо", "
мә", " ос", " ре", " се", "алд", "дал", "дег", "дей", "е б", "ет ", "жас", "й б", "лау",
"лда", "мет", "нын", "сар", "сі ", "ті ", "ыры", "ыта", "ісі", "ң а", "өте", " ат", " ел", "
жү", " ма", " то", " шы", "а а", "алт", "ама", "арл", "аст", "бұл", "дай", "дық", "ек ",
"ель", "есі", "зді", "көт", "лем", "ль ", "н е", "п а", "р а", "рес", "са ", "та ", "тте",
"тұр", "шы ", "ы д", "ы қ", "ыз ", "қыт", " ко", " не", " ой", " ор", " сұ", " тү", "аль",
"аре", "атт", "дір", "ев ", "егі", "еда", "екі", "елд", "ерг", "ерд", "ияд", "кер", "кет",
"лыс", "ліс", "мед", "мпи", "н д", "ні ", "нін", "п т", "пек", "рел", "рта", "ріл", "рін",
"сен", "тал", "шіл", "ы к", "ы м", "ыст"};
            tmodels["ky"] = new string[]  {"ын ", "ан ", " жа", "ен ", "да ", " та", "ар ",
"ин ", " ка", "ары", " ал", " ба", " би", "лар", " бо", " кы", "ала", "н к", " са", "нда",
"ган", "тар", " де", "анд", "н б", " ке", "ард", "мен", "н т", "ара", "нын", " да", " ме",
"кыр", " че", "н а", "ры ", " ко", "ген", "дар", "кен", "кта", "уу ", "ене", "ери", " ша",
"алы", "ат ", "на ", " кө", " эм", "аты", "дан", "деп", "дын", "еп ", "нен", "рын", " бе",
"кан", "луу", "ргы", "тан", "шай", "ырг", "үн ", " ар", " ма", "агы", "акт", "аны", "гы ",
"гыз", "ды ", "рда", "ай ", "бир", "бол", "ер ", "н с", "нды", "ун ", "ча ", "ынд", "а к",
"ага", "айл", "ана", "ап ", "га ", "лге", "нча", "п к", "рды", "туу", "ыны", " ан", " өз",
"ама", "ата", "дин", "йт ", "лга", "лоо", "оо ", "ри ", "тин", "ыз ", "ып ", "өрү", " па", "
эк", "а б", "алг", "асы", "ашт", "биз", "кел", "кте", "тал", " не", " су", "акы", "ент",
"инд", "ир ", "кал", "н д", "нде", "ого", "онд", "оюн", "р б", "р м", "ран", "сал", "ста",
"сы ", "ура", "ыгы", " аш", " ми", " сы", " ту", "ал ", "арт", "бор", "елг", "ени", "ет ",
"жат", "йло", "кар", "н м", "огу", "п а", "п ж", "р э", "сын", "ык ", "юнч", " бу", " ур", "а
а", "ак ", "алд", "алу", "бар", "бер", "бою", "ге ", "дон", "еги", "ект", "ефт", "из ",
"кат", "лды", "н ч", "н э", "н ө", "ндо", "неф", "он ", "сат", "тор", "ты ", "уда", "ул ",
"ула", "ууд", "ы б", "ы ж", "ы к", "ыл ", "ына", "эке", "ясы", " ат", " до", " жы", " со", "
чы", "аас", "айт", "аст", "баа", "баш", "гар", "гын", "дө ", "е б", "ек ", "жыл", "и б", "ик
", "ияс", "кыз", "лда", "лык", "мда", "н ж", "нди", "ни ", "нин", "орд", "рдо", "сто", "та ",
"тер", "тти", "тур", "тын", "уп ", "ушу", "фти", "ыкт", "үп ", "өн ", " ай", " бү", " ич", "
иш", " мо", " пр", " ре", " өк", " өт", "а д", "а у", "а э", "айм", "амд", "атт", "бек",
"бул", "гол", "дег", "еге", "ейт", "еле", "енд", "жак", "и к", "ини", "ири", "йма", "кто",
"лик", "мак", "мес", "н у", "н ш", "нтт", "ол ", "оло", "пар", "рак", "рүү", "сыр", "ти ",
"тик", "тта", "төр", "у ж", "у с", "шка", "ы м", "ызы", "ылд", "эме", "үрү", "өлү", "өтө", "
же", " тү", " эл", " өн", "а ж", "ады"};
            tmodels["la"] = new string[]  {"um ", "us ", "ut ", "et ", "is ", " et", " in", "
qu", "tur", " pr", "est", "tio", " au", "am ", "em ", "aut", " di", "ent", "in ", "dic", "t
e", " es", "ur ", "ati", "ion", "st ", " ut", "ae ", "qua", " de", "nt ", " su", " si",
"itu", "unt", "rum", "ia ", "es ", "ter", " re", "nti", "rae", "s e", "qui", "io ", "pro",
"it ", "per", "ita", "one", "ici", "ius", " co", "t d", "bus", "pra", "m e", " no", "edi",
"tia", "ue ", "ibu", " se", " ad", "er ", " fi", "ili", "que", "t i", "de ", "oru", " te",
"ali", " pe", "aed", "cit", "m d", "t s", "tat", "tem", "tis", "t p", "sti", "te ", "cum",
"ere", "ium", " ex", "rat", "ta ", "con", "cti", "oni", "ra ", "s i", " cu", " sa", "eni",
"nis", "nte", "eri", "omi", "re ", "s a", "min", "os ", "ti ", "uer", " ma", " ue", "m s",
"nem", "t m", " mo", " po", " ui", "gen", "ict", "m i", "ris", "s s", "t a", "uae", " do", "m
a", "t c", " ge", "as ", "e i", "e p", "ne ", " ca", "ine", "quo", "s p", " al", "e e",
"ntu", "ro ", "tri", "tus", "uit", "atu", "ini", "iqu", "m p", "ost", "res", "ura", " ac", "
fu", "a e", "ant", "nes", "nim", "sun", "tra", "e a", "s d", " pa", " uo", "ecu", " om", "
tu", "ad ", "cut", "omn", "s q", " ei", "ex ", "icu", "tor", "uid", " ip", " me", "e s",
"era", "eru", "iam", "ide", "ips", " iu", "a s", "do ", "e d", "eiu", "ica", "im ", "m c", "m
u", "tiu", " ho", "cat", "ist", "nat", "on ", "pti", "reg", "rit", "s t", "sic", "spe", "
en", " sp", "dis", "eli", "liq", "lis", "men", "mus", "num", "pos", "sio", " an", " gr",
"abi", "acc", "ect", "ri ", "uan", " le", "ecc", "ete", "gra", "non", "se ", "uen", "uis", "
fa", " tr", "ate", "e c", "fil", "na ", "ni ", "pul", "s f", "ui ", "at ", "cce", "dam", "i
e", "ina", "leg", "nos", "ori", "pec", "rop", "sta", "uia", "ene", "iue", "iui", "siu", "t
t", "t u", "tib", "tit", " da", " ne", "a d", "and", "ege", "equ", "hom", "imu", "lor", "m
```

```
m", "mni", "ndo", "ner", "o e", "r e", "sit", "tum", "utu", "a p", "bis", "bit", "cer",
"cta", "dom", "fut", "i s", "ign", "int", "mod", "ndu", "nit", "rib", "rti", "tas", "und", "
ab", "err", "ers", "ite", "iti", "m t", "o p"};
            tmodels["lt"] = new string[]  {"as ", " pa", " ka", "ai ", "us ", "os ", "is ", "
ne", " ir", "ir ", "ti ", " pr", "aus", "ini", "s p", "pas", "ių ", " ta", " vi", "iau", "
ko", " su", "kai", "o p", "usi", " sa", "vo ", "tai", "ali", "tų ", "io ", "jo ", "s k",
"sta", "iai", " bu", " nu", "ius", "mo ", " po", "ien", "s s", "tas", " me", "uvo", "kad", "
iš", " la", "to ", "ais", "ie ", "kur", "uri", " ku", "ijo", "čia", "au ", "met", "je ", "
va", "ad ", " ap", "and", " gr", " ti", "kal", "asi", "i d", "s g", "ta ", "uot", " ja", "
už", "aut", "i s", "ino", "mą ", "oje", "rav", "dėl", "nti", "o a", "toj", "ėl ", " to", "
vy", "ar ", "ina", "lic", "o v", "sei", "su ", " mi", " pi", "din", "iš ", "lan", "si ",
"tus", " ba", "asa", "ata", "kla", "omi", "tat", " an", " ji", "als", "ena", "jų ", "nuo",
"per", "rig", "s m", "val", "yta", "čio", " ra", "i k", "lik", "net", "nė ", "tis", "tuo",
"yti", "ęs ", "ų s", "ada", "ari", "do ", "eik", "eis", "ist", "lst", "ma ", "nes", "sav",
"sio", "tau", " ki", "aik", "aud", "ies", "ori", "s r", "ska", " ge", "ast", "eig", "et ",
"iam", "isa", "mis", "nam", "ome", "žia", "aba", "aul", "ikr", "ką ", "nta", "ra ", "tur", "
ma", "die", "ei ", "i t", "nas", "rin", "sto", "tie", "tuv", "vos", "ų p", " dė", "are",
"ats", "enė", "ili", "ima", "kar", "ms ", "nia", "r p", "rod", "s l", " o ", "e p", "es ",
"ide", "ik ", "ja "};
            tmodels["lv"] = new string[]  {"as ", " la", " pa", " ne", "es ", " un", "un ", "
ka", " va", "ar ", "s p", " ar", " vi", "is ", "ai ", " no", "ja ", "ija", "iem", "em ", "tu
", "tie", "vie", "lat", "aks", "ien", "kst", "ies", "s a", "rak", "atv", "tvi", " ja", " pi",
"ka ", " ir", "ir ", "ta ", " sa", "ts ", " kā", "ās ", " ti", "ot ", "s n", " ie", " ta",
"arī", "par", "pie", " pr", "kā ", " at", " ra", "am ", "inā", "tā ", " iz", "jas", "lai", "
na", "aut", "ieš", "s s", " ap", " ko", " st", "iek", "iet", "jau", "us ", "rī ", "tik",
"ība", "na ", " ga", "cij", "s i", " uz", "jum", "s v", "ms ", "var", " ku", " ma", "jā ",
"sta", "s u", " tā", "die", "kai", "kas", "ska", " ci", " da", "kur", "lie", "tas", "a p",
"est", "stā", "šan", "nes", "nie", "s d", "s m", "val", " di", " es", " re", "no ", "to ",
"umu", "vai", "ši ", " vē", "kum", "nu ", "rie", "s t", "ām ", "ad ", "et ", "mu ", "s l", "
be", "aud", "tur", "vij", "viņ", "āju", "bas", "gad", "i n", "ika", "os ", "a v", "not",
"oti", "sts", "aik", "u a", "ā a", "āk ", " to", "ied", "stu", "ti ", "u p", "vēl", "āci", "
šo", "gi ", "ko ", "pro", "s r", "tāj", "u s", "u v", "vis", "aun", "ks ", "str", "zin", "a
a", "adī", "da ", "dar", "ena", "ici", "kra", "nas", "stī", "šu ", " mē", "a n", "eci", "i
s", "ie ", "iņa", "ju ", "las", "r t", "ums", "šie", "bu ", "cit", "i a", "ina", "ma ",
"pus", "ra ", " au", " se", " sl", "a s", "ais", "eši", "iec", "iku", "pār", "s b", "s k",
"sot", "ādā", " in", " li", " tr", "ana", "eso", "ikr", "man", "ne ", "u k", " tu", "an ",
"av ", "bet", "būt", "im ", "isk", "līd", "nav", "ras", "ri ", "s g", "sti", "īdz", " ai",
"arb", "cin", "das", "ent", "gal", "i p", "lik", "mā ", "nek", "pat", "rēt", "si ", "tra",
"uši", "vei", " br", " pu", " sk", "als", "ama", "edz", "eka", "ešu", "ieg", "jis", "kam",
"lst", "nāk", "oli", "pre", "pēc", "rot", "tās", "usi", "ēl ", "ēs ", " bi", " de", " me", "
pā", "a i", "aid", "ajā", "ikt", "kat", "lic", "lod", "mi ", "ni ", "pri", "rād", "rīg",
"sim", "trā", "u l", "uto", "uz ", "ēc ", "ītā", " ce", " jā", " sv", "a t", "aga", "aiz",
"atu", "ba ", "cie", "du ", "dzi", "dzī"};

            tmodels["ne"] = new string[]  {"को ", "का ", "मा ", "हरु", " ने", "नेप", "पाल",
"ेपा", " सम", "ले ", " प्", "प्र", "कार", "ा स", "एको", " भए", " छ ", " भा", "्रम", " गर",
"रुक", " र ", "भार", "ारत", " का", " वि", "भएक", "ाली", "ली ", "ा प", "ीहर", "ार्", "ो छ",
"ना ", "रु ", "ालक", "्या", " बा", "एका", "ने ", "न्त", "ा ब", "ाको", "ार ", "ा भ", "ाहर",
"्रो", "क्ष", "न् ", "ारी", " नि", "ा न", "ी स", " डु", "क्र", "जना", "यो ", "ा छ", "ेवा", "्ता",
" रा", "त्य", "न्द", "हुन", "ा क", "ामा", "ी न", "्दा", " से", "छन्", "म्ब", "रोत", "सेव", "स्त",
"स्र", "ेका", "्त ", " बी", " हु", "क्त", "त्र", "रत ", "र्न", "र्य", "ा र", "ाका", "ुको", " एक", "
सं", " सु", "बीब", "बीस", "लको", "स्य", "ीबी", "ीसी", "ेको", "ो स", "्यक", " छन", " जन", "
बि", " मु", " स्", "गर्", "ताह", "न्ध", "बार", "मन्", "मस्", "रुल", "लाई", "ा व", "ाई ", "ाल ",
"िका", " त्", " मा", " यस", " रु", "ताक", "बन्", "र ब", "रण ", "रुप", "रेक", "ष्ट", "सम्", "सी ",
"ाएक", "ुका", "ुक्", " अध", " अन", " तथ", " थि", " दे", " पर", " बै", "तथा", "ता ", "दा ",
"द्व", "नी ", "बाट", "यक्", "री ", "रीह", "र्म", "लका", "समस", "ा अ", "ा ए", "ाट ", "िय ", "ो
```

```
प", "ो म", "्न ", "्ने", "्षा", " पा", " यो", " हा", "अधि", "डुव", "त भ", "त स", "था ", "धिक",
"पमा", "बैठ", "मुद", "या ", "युक", "र न", "रति", "वान", "सार", "ा आ", "ा ज", "ा ह", "ुद्",
"ुपम", "ुले", "ुवा", "ैठक", "ो ब", "्तर", "्य ", "्यस", " क्", " मन", " रह", "चार", "तिय",
"दै ", "निर", "नु ", "पर्", "रक्", "र्द", "समा", "सुर", "ाउन", "ान ", "ानम", "ारण", "ाले", "ि
ब", "ियो", "ुन्", "ुरक", "्त्", "्बन", "्रा", "्ष ", " आर", " जल", " बे", " या", " सा",
"आएक", "एक ", "कर्", "जलस", "णका", "त र", "द्र", "धान", "धि ", "नका", "नमा", "नि ", "ममा",
"रम ", "रहे", "राज", "लस्", "ला ", "वार", "सका", "हिल", "हेक", "ा त", "ारे", "िन्", "िस्", "े
स", "ो न", "ो र", "ोत ", "्धि", "्मी", "्रस", " दु", " पन", " बत", " बन", " भन", "ंयु",
"आरम", "खि ", "ण्ड", "तका", "ताल", "दी ", "देख", "निय", "पनि", "प्त", "बता", "मी ", "म्भ", "र
स", "रम्", "लमा", "विश", "षाक", "संय", "ा ड", "ा म", "ानक", "ालम", "ि भ", "ित ", "ी प",
"ी र", "ु भ", "ुने", "े ग", "ेखि", "ेर ", "ो भ", "ो व", "ो ह", "्भ ", "्र ", " ता", "
नम", " ना"};
            tmodels["pl"] = new string[]  {"ie ", "nie", "em ", " ni", " po", " pr", "dzi", "
na", "że ", "rze", "na ", "łem", "wie", " w ", " że", "go ", " by", "prz", "owa", "ię ", "
do", " si", "owi", " pa", " za", "ch ", "ego", "ał ", "się", "ej ", "wał", "ym ", "ani",
"ałe", "to ", " i ", " to", " te", "e p", " je", " z ", "czy", "był", "pan", "sta", "kie", "
ja", "do ", " ch", " cz", " wi", "iał", "a p", "pow", " mi", "li ", "eni", "zie", " ta", "
wa", "ło ", "ać ", "dy ", "ak ", "e w", " a ", " od", " st", "nia", "rzy", "ied", " kt",
"odz", "cie", "cze", "ia ", "iel", "któ", "o p", "tór", "ści", " sp", " wy", "jak", "tak",
"zy ", " mo", "ałę", "pro", "ski", "tem", "łęs", " tr", "e m", "jes", "my ", " ro", "edz",
"eli", "iej", " rz", "a n", "ale", "an ", "e s", "est", "le ", "o s", "i p", "ki ", " co",
"ada", "czn", "e t", "e z", "ent", "ny ", "pre", "rzą", "y s", " ko", " o ", "ach", "am ", "e
n", "o t", "oli", "pod", "zia", " go", " ka", "by ", "ieg", "ier", "noś", "roz", "spo",
"ych", "ząd", " mn", "acz", "adz", "bie", "cho", "mni", "o n", "ost", "pra", "ze ", "ła ", "
so", "a m", "cza", "iem", "ić ", "obi", "ył ", "yło", " mu", " mó", "a t", "acj", "ci ", "e
b", "ich", "kan", "mi ", "mie", "ośc", "row", "zen", "zyd", " al", " re", "a w", "den",
"edy", "ił ", "ko ", "o w", "rac", "śmy", " ma", " ra", " sz", " ty", "e j", "isk", "ji ",
"ka ", "m s", "no ", "o z", "rez", "wa ", "ów ", "łow", "ść ", " ob", "ech", "ecz", "ezy", "i
w", "ja ", "kon", "mów", "ne ", "ni ", "now", "nym", "pol", "pot", "yde", " dl", " sy", "a
s", "aki", "ali", "dla", "icz", "ku ", "ocz", "st ", "str", "szy", "trz", "wia", "y p", "za
", " wt", "chc", "esz", "iec", "im ", "la ", "o m", "sa ", "wać", "y n", "zac", "zec", " gd",
"a z", "ard", "co ", "dar", "e r", "ien", "m n", "m w", "mia", "moż", "raw", "rdz", "tan",
"ted", "teg", "wił", "wte", "y z", "zna", "zło", "a r", "awi", "bar", "cji", "czą", "dow",
"eż ", "gdy", "iek", "je ", "o d", "tał", "wal", "wsz", "zed", "ówi", "ęsa", " ba", " lu", "
wo", "aln", "arn", "ba ", "dzo", "e c", "hod", "igi", "lig", "m p", "myś", "o c", "oni",
"rel", "sku", "ste", "y w", "yst", "z w"};
            tmodels["ps"] = new string[] {" د ", "اؤ ", " اؤ", "نو ", "د ي", "ره ", " په",
"نه ", "چې ", " چې", "په ", "د ه", "ته ", "ا و", "ونو", "د و", " او", "انو", "ونه", "ک ه",
" دا", "ا ه", "دي ", "ښي ", " کي", "ان ", "لو ", "هم ", "م و", "کښي", "م ه", "ا ی", " نو",
" ته", " کښ", "نرو", "کي ", "ده ", "له ", "به ", "رو ", " هم", "و ه", "وی ", "او ", "تون",
"دا ", " کو", " کړ", "قام", " تر", "ران", "پ ه", "و ي", "پ ي", " به", " خو", "تو ", "د د",
"ا د", "ت ه", "پ و", "يا ", " خپ", " دو", " را", " مش", " پر", "ارو", "ري ", "د م", "مشر",
" شو", " ور", "ار ", "دی ", " اد", " دی", " مو", "پ د", " لي", "ک و", " مق", " يو", "د ؤ",
"خپل", "سره", "چ ه", "ور ", " تا", " دي", " رو", " سر", " مل", " کا", "ا ؤ", " اره", "برو",
"مه ", "ب ه", "ت و", "پښت", " با", " دغ", " قب", " له", " وا", " پا", " پښ", "د م", "د ه",
"لي ", "مات", "مو ", "ه ه", "وي ", "ب ي", "ک ي", " ده", " قا", "ال ", "اما", "د ن", "قبر",
"ن ه", "پار", " اث", " بي", " لا ", " لر", "اثا", "خ د", "دار", "ريخ", "شرا", "مقا", " نی ",
"ر ه", "ل ه", "ولو", " يو", "کوم", " دد", " لو", " مح", " مر", " وو", "اتو", "اري", "الو",
"اند", "خان", "ت د", "سي ", "لی ", "نور", "ل و", "چ ي", " ړي", "ښتو", "ل ي", " جو", " سي",
"ام ", "بان", "تار", " تر", "ثار", " خو", " دو", "ک ر", "د ل", "مون", "ندي", "ن و", "ول ",
"وه ", "و ی", "د ي", "ا ي", "ت ي", "ي ي", " حک", " خب", " نه", " پو", "د ا", " تي", "جوړ",
"حکم", "حکو", "خبر", "دان", "د ر", "غه ", "قاف", "محک", "وال", "ومت", "ويل", "د ی", "م ی",
"يره", " پر", "کول", "ه ي", " تي", " خا", " وک", " يا", " ځا", "ق ؤ", "انی", " بی ", "غو ",
"خ ه", "ب و", "ودا", "دوي", " ړي", "کال", " بر", " قد", " مي", " وي", " کر", "ؤ م", "ات ",
"ايي", " تی ", "تيا", "تير", "خوا", "دغو", " دم ", "ديم", "و ر", "قدي", "خ م", "مان", " مي ",
"نيو", " نړ ", "ي ه", "س و", "چ و", "وان", "ورو", "ونړ", "پور", " ړه ", " ړو ", "د ی", "ن ي",
" اه", " زي", " سو", " شي", " هر", " هغ", " ښا", "اتل", "اق ", "اني", "بري", "بي ", "ا ت",
"ب د", "س د", "م ر", "ری ", "عرا", "لا ن", "مى ", "نى ", "و خ", "ئو ", "ورک", "وري", "ون ",
"وکړ", "چ ی", "يمه", "پي ", "ښتن", "که ", "کړي", "ي خ", "ے ش", " تح", " تو", " رد", " دپ",
" صو", " عر", " ول", " يؤ", " پئ", " ځو", "ا ا"};
            tmodels["pt_br"] = new string[]  {"eq ", "ent", " en", "q e", "q i", "g e", "g
i", "ng ", " id", "ida", "nte", "te ", " es", " in", "ade", "ag ", "dad", "de ", "ia ",
"ing", " br", " sa", "est", "inq", "lin", "mo ", "nq ", "o a", "seq", " co", " li", " ni", "
o ", "a a", "a c", "ado", "asi", "bra", "dor", "iq ", "nta", "o b", "or ", "q n", "ras",
"sil", "str", "ta ", "tre", "us ", " a ", " ag", " an", " ca", " e ", " eq", " g ", " i ", "
ir", " nc", " q ", " se", " ve", "ant", "ar ", "cia", "con", "e a", "eir", "el ", "ig ",
"ili", "imo", "io ", "ir ", "nci", "o t", "ro ", "vel", " ap", " bo", " de", " fr", " tr", "a
b", "a e", "a g", "a v", "apo", "as ", "bus", "ca ", "cet", "cin", "des", "e b", "e s",
"eta", "fre", "i a", "ibu", "il ", "iro", "la ", "liq", "nib", "nti", "o c", "o q", "os ",
"ra ", "re ", "req", "s a", "s s", "san", "sim", "tar", "to ", "ult", " ba", " ci", " el", "
em", " fi", " gr", " gu", " ia", " mu", " pe", " po", " re", " ri", " si", " su", " te", "
vi", "a o", "a s", "abe", "alc", "and", "ara", "arg", "ari", "ben", "boc", "car", "co ", "do
", "e f", "e g", "e l", "e o", "em ", "emo", "en ", "es ", "esp", "exe", "fic", "g n", "g s",
"gra", "gua", "ias", "ica", "idi", "ila", "ile", "inh", "l b", "l e", "lei", "loq", "mos",
"mul", "nad", "nio", "nt ", "nto", "o g", "o r", "oce", "ont", "oq ", "pos", "q v", "r b", "r
e", "r i", "r s", "rad", "ran", "rem", "rg ", "ria", "rio", "s e", "s p", "sta", "sti",
"tig", "til", "tra", "ua ", "ue ", "va ", "xeq", " ' ", " ab", " ad", " ae", " al", " am", "
aq", " ar", " b ", " bi", " bl", " bu", " cc", " ch", " di", " et", " ex", " fa", " ic", "
il", " im", " is", " it", " ll", " m ", " me", " na", " ne", " ng", " nu", " ob", " ou", "
pi", " qu", " ss", " st", " ti", " ub", " un", " v ", " x ", "' c" , "a d", "a f", "a i", "a
n", "a p", "a t", "abr", "aci", "ad ", "ada", "adr", "aer", "afe", "alv", "amb", "amp", "an
", "ana", "ang", "anh", "ani", "ano", "anq", "apa", "aq ", "ati", "ato", "azi", "b n", "ban",
"bar", "big", "bil", "biq", "bli", "blu", "bon", "bre", "bri", "bse", "buc", "c f", "c i",
"cad", "caf", "cag", "cal", "can", "cc "};
            tmodels["pt_pt"] = new string[]  {"equ", "ent", "que", "qui", "gui", "uen", "
li", "ngu", "qu ", "uid", " co", " ve", "de ", "gue", "ida", "nte", "o a", "a a", "ade",
"dad", "el ", "ing", "mo ", "nqu", "nta", "seq", "u n", "vel", " de", " o ", " se", "a c",
"ado", "ar ", "est", "ia ", "inq", "io ", "iqu", "lin", "o c", "o p", "ort", "por", "ta ",
"te ", " ag", " eq", " nc", " pi", " po", " sa", "a d", "a e", "ant", "as ", "ca ", "cia",
"des", "do ", "gu ", "imo", "l p", "nci", "ro ", "rtu", "str", "tug", "u s", "ues", "ui ", "
a ", " an", " ap", " ba", " bi", " ca", " fr", " gu", " in", " pe", " qu", "agu", "apa",
"con", "dor", "e f", "e g", "eir", "fre", "ho ", "i a", "ica", "igu", "iro", "liq", "nti", "o
b", "o l", "o s", "or ", "r o", "ra ", "req", "s c", "sim", "tar", "to ", "ue ", "uin",
"ult", " ci", " en", " ho", " mu", " ni", " re", " s ", " si", " su", "a s", "a v", "abe",
"ag ", "al ", "and", "anh", "apo", "ata", "ban", "ben", "bic", "boi", "cap", "car", "cin",
"co ", "com", "cto", "dei", "e b", "e o", "e s", "eca", "en ", "er ", "es ", "exe", "fic",
"for", "gal", "gra", "ias", "ich", "ico", "idi", "ili", "ir ", "ira", "isb", "la ", "lis",
"mbo", "mul", "na ", "nho", "nio", "nt ", "o q", "o r", "o t", "oc ", "oio", "omb", "oo ",
"os ", "par", "pe ", "r b", "r e", "r s", "ran", "re ", "rec", "s a", "s s", "san", "sbo",
"so ", "sta", "tan", "tra", "tre", "u v", "uga", "ugu", "xeq", " ab", " ad", " al", " am", "
aq", " ar", " au", " b ", " bo", " c ", " ch", " ct", " cu", " el", " es", " ex", " fa", "
fi", " ga", " gr", " id", " ir", " ne", " ng", " nu", " ob", " oo", " pa", " ps", " pt", " r
", " ra", " ri", " ss", " st", " ta", " te", " tr", " ub", " un", " vi", " vo", "a f", "a i",
"a l", "a m", "a o", "a r", "ach", "aci", "act", "ad ", "afa", "age", "agr", "alf", "alh",
"am ", "amb", "ami", "ana", "ang", "anq", "aqu", "ara", "arb", "arc", "arg", "ari", "arr",
"asa", "ati", "aut", "azi", "b n", "bar", "bat", "ber", "big", "bil", "biq", "bli", "boa",
"boe", "bor", "bse", "c i", "c l", "c p", "cad", "cam", "cas", "ch ", "cha", "che", "chi",
"cio", "coc", "coi", "cou", "ctr", "cue", "cul"};
            tmodels["ro"] = new string[]  {" de", " în", "de ", " a ", "ul ", " co", "în ",
"re ", "e d", "ea ", " di", " pr", "le ", "şi ", "are", "at ", "con", "ui ", " şi", "i d",
"ii ", " cu", "e a", "lui", "ern", "te ", "cu ", " la", "a c", "că ", "din", "e c", "or ",
"ulu", "ne ", "ter", "la ", "să ", "tat", "tre", " ac", " să", "est", "st ", "tă ", " ca", "
ma", " pe", "cur", "ist", "mân", "a d", "i c", "nat", " ce", "i a", "ia ", "in ", "scu", "
mi", "ato", "aţi", "ie ", " re", " se", "a a", "int", "ntr", "tru", "uri", "ă a", " fo", "
pa", "ate", "ini", "tul", "ent", "min", "pre", "pro", "a p", "e p", "e s", "ei ", "nă ",
"par", "rna", "rul", "tor", " in", " ro", " tr", " un", "al ", "ale", "art", "ce ", "e e", "e
î", "fos", "ita", "nte", "omâ", "ost", "rom", "ru ", "str", "ver", " ex", " na", "a f",
"lor", "nis", "rea", "rit", " al", " eu", " no", "ace", "cer", "ile", "nal", "pri", "ri ",
"sta", "ste", "ţie", " au", " da", " ju", " po", "ar ", "au ", "ele", "ere", "eri", "ina", "n
a", "n c", "res", "se ", "t a", "tea", " că", " do", " fi", "a s", "ată", "com", "e ş",
"eur", "guv", "i s", "ice", "ili", "na ", "rec", "rep", "ril", "rne", "rti", "uro", "uve", "ă
p", " ar", " o ", " su", " vi", "dec", "dre", "oar", "ons", "pe ", "rii", " ad", " ge", "a
m", "a r", "ain", "ali", "car", "cat", "ecu", "ene", "ept", "ext", "ilo", "iu ", "n p",
"ori", "sec", "u p", "une", "ă c", "şti", "ţia", " ch", " gu", "ai ", "ani", "cea", "e f",
"isc", "l a", "lic", "liu", "mar", "nic", "nt ", "nul", "ris", "t c", "t p", "tic", "tid", "u
a", "ucr", " as", " dr", " fa", " nu", " pu", " to", "cra", "dis", "enţ", "esc", "gen", "it
", "ivi", "l d", "n d", "nd ", "nu ", "ond", "pen", "ral", "riv", "rte", "sti", "t d", "ta ",
"to ", "uni", "xte", "ând", "îns", "ă s", " bl", " st", " uc", "a b", "a i", "a l", "air",
"ast", "bla", "bri", "che", "duc", "dul", "e m", "eas", "edi", "esp", "i l", "i p", "ica",
"ică", "ir ", "iun", "jud", "lai", "lul", "mai", "men", "ni ", "pus", "put", "ra ", "rai",
"rop", "sil", "ti ", "tra", "u s", "ua ", "ude", "urs", "ân ", "înt", "ţă ", " lu", " mo", "
s ", " sa", " sc", "a u", "an ", "atu"};
",
           tmodels["sk"] = new string[]  {" pr", " po", " ne", " a ", "ch ", " na", " je",
"ní ", "je ", " do", "na ", "ova", " v ", "to ", "ho ", "ou ", " to", "ick", "ter", "že ", "
st", " za", "ost", "ých", " se", "pro", " te", "e s", " že", "a p", " kt", "pre", " by", " o
", "se ", "kon", " př", "a s", "né ", "ně ", "sti", "ako", "ist", "mu ", "ame", "ent", "ky ",
"la ", "pod", " ve", " ob", "om ", "vat", " ko", "sta", "em ", "le ", "a v", "by ", "e p",
"ko ", "eri", "kte", "sa ", "ého", "e v", "mer", "tel", " ak", " sv", " zá", "hla", "las",
"lo ", " ta", "a n", "ej ", "li ", "ne ", " sa", "ak ", "ani", "ate", "ia ", "sou", " so",
"ení", "ie ", " re", "ce ", "e n", "ori", "tic", " vy", "a t", "ké ", "nos", "o s", "str",
"ti ", "uje", " sp", "lov", "o p", "oli", "ová", " ná", "ale", "den", "e o", "ku ", "val", "
am", " ro", " si", "nie", "pol", "tra", " al", "ali", "o v", "tor", " mo", " ni", "ci ", "o
n", "ím vá", "bez", "byv", "cho"};
            tmodels["sl"] = new string[]  {"je ", " pr", " po", " je", " v ", " za", " na",
"pre", "da ", " da", "ki ", "ti ", "ja ", "ne ", " in", "in ", "li ", "no ", "na ", "ni ", "
bi", "jo ", " ne", "nje", "e p", "i p", "pri", "o p", "red", " do", "anj", "em ", "ih ", "
bo", " ki", " iz", " se", " so", "al ", " de", "e v", "i s", "ko ", "bil", "ira", "ove", "
br", " ob", "e b", "i n", "ova", "se ", "za ", "la ", " ja", "ati", "so ", "ter", " ta", "a
s", "del", "e d", " dr", " od", "a n", "ar ", "jal", "ji ", "rit", " ka", " ko", " pa", "a
b", "ani", "e s", "er ", "ili", "lov", "o v", "tov", " ir", " ni", " vo", "a j", "bi ",
"bri", "iti", "let", "o n", "tan", "še ", " le", " te", "eni", "eri", "ita", "kat", "por",
"pro", "ali", "ke ", "oli", "ov ", "pra", "ri ", "uar", "ve ", " to", "a i", "a v", "ako",
"arj", "ate", "di ", "do ", "ga ", "le ", "lo ", "mer", "o s", "oda", "oro", "pod", " ma", "
mo", " si", "a p", "bod", "e n", "ega", "ju ", "ka ", "lje", "rav", "ta ", "a o", "e t", "e
z", "i d", "i v
"gay", "guu", "h e", "hal", "iga", "ihi", "iri", "iye", "ken", "lad", "lid", "lsh", "mag",
"mun", "n h", "n i", "na ", "o n", "o w", "ood", "oor", "ora", "qab", "qor", "rab", "rit",
"rta", "s o", "sab", "ska", "to ", "u a", "u h", "u u", "ud ", "ugu", "uls", "uud", "waa",
"xus", "y b", "y q", "y s", "yad", "yay", "yih", " aa", " bo", " br", " go", " ji", " mi", "
of", " ti", " um", " wi", " xo", "a x"};
            tmodels["sq"] = new string[]  {"të ", " të", "në ", "për", " pë", " e ", "sht", "
në", " sh", "se ", "et ", "ë s", "ë t", " se", "he ", "jë ", "ër ", "dhe", " pa", "ë n", "ë
p", " që", " dh", "një", "ë m", " nj", "ësh", "in ", " me", "që ", " po", "e n", "e t",
"ish", "më ", "së ", "me ", "htë", " ka", " si", "e k", "e p", " i ", "anë", "ar ", " nu",
"und", "ve ", " ës", "e s", " më", "nuk", "par", "uar", "uk ", "jo ", "rë ", "ta ", "ë f",
"en ", "it ", "min", "het", "n e", "ri ", "shq", "ë d", " do", " nd", "sh ", "ën ", "atë",
"hqi", "ist", "ë q", " gj", " ng", " th", "a n", "do ", "end", "imi", "ndi", "r t", "rat", "ë
b", "ëri", " mu", "art", "ash", "qip", " ko", "e m", "edh", "eri", "je ", "ka ", "nga", "si
", "te ", "ë k", "ësi", " ma", " ti", "eve", "hje", "ira", "mun", "on ", "po ", "re ", " pr",
"im ", "lit", "o t", "ur ", "ë e", "ë v", "ët ", " ku", " së", "e d", "es ", "ga ", "iti",
"jet", "ndë", "oli", "shi", "tje", " bë", " z ", "gje", "kan", "shk", "ënd", "ës ", " de", "
kj", " ru", " vi", "ara", "gov", "kjo", "or ", "r p", "rto", "rug", "tet", "ugo", "ali",
"arr", "at ", "d t", "ht ", "i p", "ipë", "izi", "jnë", "n n", "ohe", "shu", "shë", "t e",
"tik", "a e", "arë", "etë", "hum", "nd ", "ndr", "osh", "ova", "rim", "tos", "va ", " fa", "
fi", "a s", "hen", "i n", "mar", "ndo", "por", "ris", "sa ", "sis", "tës", "umë", "viz",
"zit", " di", " mb", "aj ", "ana", "ata", "dër", "e a", "esh", "ime", "jes", "lar", "n s",
"nte", "pol", "r n", "ran", "res", "rrë", "tar", "ë a", "ë i", " at", " jo", " kë", " re", "a
k", "ai ", "akt", "hë ", "hën", "i i", "i m", "ia ", "men", "nis", "shm", "str", "t k", "t
n", "t s", "ë g", "ërk", "ëve", " ai", " ci", " ed", " ja", " kr", " qe", " ta", " ve", "a
```

```
p", "cil", "el ", "erë", "gji", "hte", "i t", "jen", "jit", "k d", "mën", "n t", "nyr",
"ori", "pas", "ra ", "rie", "rës", "tor", "uaj", "yre", "ëm ", "ëny", " ar", " du", " ga", "
je", "dës", "e e", "e z", "ha ", "hme", "ika", "ini", "ite", "ith", "koh", "kra", "ku ",
"lim", "lis", "qën", "rën", "s s", "t d", "t t", "tir", "tën", "ver", "ë j", " ba", " in", "
tr", " zg", "a a", "a m", "a t", "abr"};
  tmodels["ss"] = new string[]  {" ku", " le", "la ", "eku", "a k", "nga", " ng", "a n",
"nge", "a l", "lo ", " ne", "eti", "kwe", "ndz", "e n", "o l", " lo", "ela", "ema", "ent",
"si ", " kw", "tsi", "i l", "wa ", "lel", "kut", "e k", "und", "ni ", "elo", "fun", "esi", "
si", "ele", "tin", "tfo", " ti", "le ", "kha", "tse", "e l", "pha", "ung", "i k", " em", "ti
", "sa ", " um", "isa", "eli", "ndl", "ing", "set", "we ", "ise", "na ", "ang", "etf", "khe",
"and", "o n", " we", "nti", "nye", "tfu", "ben", "a e", "uts", "let", "dza", "imi", "sek",
"ko ", "lok", "eni", "ye ", "ba ", "nkh", "ebe", "alo", "o k", "lan", "ga ", "aba", "seb", "
ye", "he ", "lwa", "kel", " te", " la", "kus", "wem", "ati", "ikh", "nek", "ala", "kuf", "i
n", "oku", "ats", "mts", "hla", "wen", "a t", " na", "gek", "uhl", "kub", "ngu", "ka ",
"aka", "fut", "kan", "kwa", " li", "kuc", "onk", "ban", "ana", "ulu", " se", " im", "akh",
"ume", "a i", "les", "tim", "ula", "ini", "lwe", "za ", "fo ", "hul", " no", "han", "li ",
"iph", "a s", "tis", "khu", "ta ", "dzi", "be ", "emi", "ma ", "end", "o t", "eke", "a u", "
ka", "ane", "lek", "mel", "elw", "kun", "sis", "lon", "utf", "any", "kho", "kul", "hlo", "
ba", "ufu", "aph", "lun", "e s", "hal", "ind", "isw", "o s", "use", "ekw", "me ", "ndv",
"eng", "uph", "hat", "ne ", "so ", "lul", "nom", "te ", "lol", "awu", "nel", "lu ", "ha ",
"wat", "men", "ete", " lw", "nem", "ako", "zin", "kuh", "sha", "bha", "gab", " in", "ale",
"mis", "tem", "e e", "o e", "e t", "wek", "dze", "ome", "wel", " lu", "emb", "nis", " ek",
"tsa", "u l", "o y", "dle", "ute", "len", "swa", "phe", "mkh", "ntf", "uke", "sit", "iny", "e
i", "wo ", "ani", "phi", "wet", "sin", "nhl", "mal", "mba", "mfu", "fu ", "lab", "sik",
"taw", "no ", "hle", "e u", "eki", "ase", "ali", "ulw", "ve ", "eka", "zel", "nta", "bon",
"tek", "bo ", "sig", "ama", "ile", "ule", "tfw", "mph", "uma", "kup", "emt", "asi", "dlu",
"ish", "umt", "gen", "o w", "ike", "iga", "hak", "abe", "net", "gan", "kis", "nde", "ngi",
"ukh", "bek", "mo ", "phu", "sel", "elu", "i t", "ant", "dvo", "a y", "vo ", "hum", "lis",
"dla", "gam", "ive", "jen", "ket"};
"ist", "ldı", "lu ", "mek", "mle", "n ç", "onu", "opl", "ran", "rat", "rdı", "rke", "siy",
"son", "ta ", "tçı", "tın"};
      tmodels["ts"] = new string[]  {" ku", "ku ", "na ", "ka ", "wa ", "a n", " sw",
"a m", "ya ", "a k", " ti", "swi", "hi ", "la ", " ya", " le", " hi", "a t", "a v", " va",
"ni ", " na", "ndz", " ma", "a h", " xi", "a s", "i n", "ele", "i k", "ana", "a l", "nga",
"lo ", "va ", "le ", "aka", "ela", "irh", "eka", " vu", "iwa", "a x", " ka", "yi ", " wa",
"isa", "sa ", "ko ", "ta ", "ga ", "wu ", "wi ", "tir", " ek", " mi", " ni", "o y", "elo",
"awu", "isi", "swa", "i t", "hla", "a e", " ta", " ng", " la", "a y", "ri ", "eri", " ri",
"rhi", "eke", "umb", "u t", "ndl", " yi", "lan", "i v", "esw", "mbe", "i l", "a r", "e k",
"ang", "les", "ula", "ti ", "yon", "o w", "ona", "law", "xa ", "nel", "yo ", "lel", "iko", "
lo", "amb", " a ", "i y", " xa", "ane", "wan", "i s", "ond", "fan", "end", "i h", "o l", "u
k", "mbi", "n'w" , "ke ", "dyo", " fa", "lam", "nhl", "o s", "ong", "no ", " ko", "u n", "
ha", "ho ", "oko", "u h", "i m", "o n", " yo", "ngu", "o k", "u y", "ati", "u l", "van",
"ulu", "and", "", "e h", "ari", "imi", "i e", "ind", "vum", "nts", "ime", "kom", "mfu",
"ise", " mf", "hin", "dla", "vut", "gan", "i r", "ban", "bya", "mil", "int", "ats", " dy", "u
s", "e x", "ile", "kel", "kwa", " no", "i f", "asi", "za ", "uri", "o m", "rha", "e l", "in'"
, "eta", "von", "i a", "kho", " wo", "iki", " ra", "u e", "o e", "zo ", "yin", "ink", "any",
"ket"};

      //build trigram model objects from these data
      foreach(string key in tmodels.Keys)
      {
        string[] t = (string[])tmodels[key];
        models[key] = new TrigramModel(t);
      }
      tmodels = null;//release temp data structure
    }

  }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using TranslationD.Models;
using SpeechLib;

namespace TranslationD.Controllers
{
    public class SpeechController : Controller
    {
       [HttpPost]
        public ActionResult TextToSpeech(TranslateSummary ts)
        {
            TranslateSummary TranslateSummaryObj = new TranslateSummary();


            SpVoice Voice = new SpVoice();

            string SreviceResult;
            string strTranslatedText = (ts.Text == null ? "User is currently only selected 
English" : ts.Text);
            if (strTranslatedText != null)
            {
                SreviceResult = strTranslatedText;
            }
            else
            {
                SreviceResult = "Unable to process this Sorry.";
            }
            SpeechVoiceSpeakFlags SpFlags = SpeechVoiceSpeakFlags.SVSFlagsAsync;
            Voice.Speak(SreviceResult,SpFlags);
            return View("Index", TranslateSummaryObj);
        }

        // GET: Speech
        public ActionResult Index()
        {
            return View();
        }

        // GET: Speech/Details/5
        public ActionResult Details(int id)
        {
            return View();
        }

        // GET: Speech/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: Speech/Create
        [HttpPost]
        public ActionResult Create(FormCollection collection)
        {
            try
            {
                // TODO: Add insert logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }

        // GET: Speech/Edit/5
        public ActionResult Edit(int id)
        {
            return View();
        }

        // POST: Speech/Edit/5
        [HttpPost]
        public ActionResult Edit(int id, FormCollection collection)
        {
            try
            {
                // TODO: Add update logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }

        // GET: Speech/Delete/5
        public ActionResult Delete(int id)
        {
            return View();
        }

        // POST: Speech/Delete/5
        [HttpPost]
        public ActionResult Delete(int id, FormCollection collection)
        {
            try
            {
                // TODO: Add delete logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }
    }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Speech.Recognition;
using TranslationD.Models;


namespace TranslationD.Controllers
```

```
{
    public class StPController : Controller
    {


        [HttpPost]
        public  ActionResult SpeechToText()
        {
            return View();
        }

        // GET: StP
        public ActionResult Index()
        {


            return View();
        }

        // GET: StP/Details/5
        public ActionResult Details(int id)
        {
            return View();
        }

        // GET: StP/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: StP/Create
        [HttpPost]
        public ActionResult Create(FormCollection collection)
        {
            try
            {
                // TODO: Add insert logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }

        // GET: StP/Edit/5
        public ActionResult Edit(int id)
        {
            return View();
        }

        // POST: StP/Edit/5
        [HttpPost]
        public ActionResult Edit(int id, FormCollection collection)
        {
            try
            {
                // TODO: Add update logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }

        // GET: StP/Delete/5
        public ActionResult Delete(int id)
        {
            return View();
        }

        // POST: StP/Delete/5
        [HttpPost]
        public ActionResult Delete(int id, FormCollection collection)
        {
            try
            {
                // TODO: Add delete logic here

                return RedirectToAction("Index");
            }
            catch
            {
                return View();
            }
        }
    }
}


namespace Transusing LiteMiner.classes;
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using System.Xml.Linq;
using TranslationD.Models;
using SpeechLib;
using System.Xml;
using System.Web.Mvc.Html;
using TranslationD.DAL;
using System.DlationD.Controllers
{
    public class TranslateController : Controller
    {
        // GET: Translate

        public ActionResult Index()
        {

        return View("Index");
        }

        [HttpPost]
        public ActionResult TranslateText(TranslateSummary ts)
        {
            string strTranslatedText = null;
            TranslateSummary TranslateSummaryObj = new TranslateSummary();
            string SreviceResult; SpVoice Voice = new SpVoice();
            string thirdLanguage1 = (ts.Text == null ? "User is currently only selected
English" : ts.Text);
            if (ts.TextCulture == "hiEng")
            {
                //Do the code
                try
                {
                    DictionaryDb ddb = new DictionaryDb();
                    if (ddb.HindiDict == null)
                    {
                        ddb.prepareEDb();
                        ddb.prepareHdb();
                    }
                    string Rec = ddb.SplitFunc(thirdLanguage1);
                    if (Rec != null)
                    {
                        Voice.Speak(Rec);
                        TranslateSummaryObj.Chi = Rec;
                        TranslateSummaryObj.Eng = Rec;
                    }
                    else
                    {
                        string _atch = "";
                        Voice.Speak(_atch);
                        TranslateSummaryObj.Chi = thirdLanguage1;
                        TranslateSummaryObj.Eng = _atch;
                    }
                }
                catch (Exception ex) { }
            }
            else
            {
                try
                {
                    string fromCulture1 = ts.TextCulture.ToString(); // "zh-cn";
                    string toCulture1 = "en";
                    string appId1 = "78280AF4DFA1CE1676AFE86340C690023A5AC139";
                    //string translation = null;

                    TranslatorService.LanguageServiceClient client = new
TranslatorService.LanguageServiceClient();
                    client = new TranslatorService.LanguageServiceClient();
                    strTranslatedText = client.Translate(appId1, thirdLanguage1,
fromCulture1, toCulture1);
                    // txtTranslatedText.Text = strTranslatedText;

                    //create voice object



                    if (strTranslatedText != null)
                    {
                        SreviceResult = strTranslatedText;
                    }
                    else
                    {
                        SreviceResult = "Unable to process this Sorry.";
                    }
                    Voice.Speak(SreviceResult);
                    TranslateSummaryObj.Chi = thirdLanguage1;
                    TranslateSummaryObj.Eng = SreviceResult;
                    //Response.Write("<h1>" + translatedText + "</h1>");
                    if (string.IsNullOrEmpty(SreviceResult))
                    {
                        //this.ErrorMessage = 
Westwind.Globalization.Resources.Resources.InvalidSearchResult;
                        return null;
                    }
                }
                catch (Exception ex)
                {
                    // MessageBox.Show(ex.Message);
                }
            }
            SreviceResult = null;
            return View("Index", TranslateSummaryObj);
        }





        public ActionResult DetectMethod(TranslateSummary tsD) {

            LanguageDetector ld = new LanguageDetector();
            string lanCode = ld.Detect(tsD.TextToIdentift);
            if (lanCode == null) throw new Exception("Cannot detect language");
            string languageNaturalName = ld.GetLanguageNameByCode(lanCode);
            TranslateSummary TranslateSummaryObjL = new TranslateSummary();
            TranslateSummaryObjL.Language = languageNaturalName;
            return View("Index", TranslateSummaryObjL);
        }
    }
}

namespace TranslationD.DAL
{
       using System.Data.Linq;
       using System.Data.Linq.Mapping;
       using System.Data;
       using System.Collections.Generic;
       using System.Reflection;
       using System.Linq;
       using System.Linq.Expressions;
       using System.ComponentModel;
       using System;


       [global::System.Data.Linq.Mapping.DatabaseAttribute(Name="HinglishSlang")]
       public partial class DictionaryDataContext : System.Data.Linq.DataContext
       {

              private static System.Data.Linq.Mapping.MappingSource mappingSource = new 
AttributeMappingSource();

    #region Extensibility Method Definitions
    partial void OnCreated();
    #endregion
```

```
        public DictionaryDataContext() : 

    base(global::System.Configuration.ConfigurationManager.ConnectionStrings["HinglishSlang
ConnectionString"].ConnectionString, mappingSource)
        {
            OnCreated();
        }

        public DictionaryDataContext(string connection) : 
                base(connection, mappingSource)
        {
            OnCreated();
        }

        public DictionaryDataContext(System.Data.IDbConnection connection) : 
                base(connection, mappingSource)
        {
            OnCreated();
        }

        public DictionaryDataContext(string connection, 
System.Data.Linq.Mapping.MappingSource mappingSource) : 
                base(connection, mappingSource)
        {
            OnCreated();
        }

        public DictionaryDataContext(System.Data.IDbConnection connection, 
System.Data.Linq.Mapping.MappingSource mappingSource) : 
                base(connection, mappingSource)
        {
            OnCreated();
        }

        public System.Data.Linq.Table<Dictionary> Dictionaries
        {
            get
            {
                return this.GetTable<Dictionary>();
            }
        }
    }

    [global::System.Data.Linq.Mapping.TableAttribute(Name="dbo.Dictionary")]
    public partial class Dictionary
    {

        private string _Hindi;

        private string _English;

        private long _Id;

        public Dictionary()
        {
        }

        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Hindi", 
DbType="VarChar(2000)")]
        public string Hindi
        {
            get
```

```
                     {
                            return this._Hindi;
                     }
                     set
                     {
                            if ((this._Hindi != value))
                            {
                                   this._Hindi = value;
                            }
                     }
              }

              [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_English",
DbType="VarChar(MAX)")]
              public string English
              {
                     get
                     {
                            return this._English;
                     }
                     set
                     {
                            if ((this._English != value))
                            {
                                   this._English = value;
                            }
                     }
              }

              [global::System.Data.Linq.Mapping.ColumnAttribute(Storage="_Id",
AutoSync=AutoSync.Always, DbType="BigInt NOT NULL IDENTITY", IsDbGenerated=true)]
              public long Id
              {
                     get
                     {
                            return this._Id;
                     }
                     set
                     {
                            if ((this._Id != value))
                            {
                                   this._Id = value;
                            }
                     }
              }
       }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace TranslationD.DAL
{
    public class DictionaryDb
    {
        public Dictionary<int,string> HindiDict;
        public Dictionary<int,string> dict;

        public void prepareEDb()
          {
            dict = new Dictionary<int, string>();
```

```
           }

        public void prepareHdb()
           {
            HindiDict = new Dictionary<int, string>();
            }

      Dictionary<int, string> getEngDb()
        {
            return HindiDict;
        }
        public Dictionary<int, string> getHinDb() {
            return dict;
        }
        public string SplitFunc(string thirdLanguage1)
        {
            string _atch = "";
            int key = 0;
            string Rec = "";
            thirdLanguage1 = thirdLanguage1.Trim();

            key = HindiDict.FirstOrDefault(hword => hword.Value == thirdLanguage1).Key;
            Rec = dict.FirstOrDefault(word => word.Key == key).Value;
            if (Rec != null)
            {
                return Rec;
            }
            else
            {
                string s = thirdLanguage1.Split(' ')[0];
                string ss = thirdLanguage1.Substring(s.Length).Trim();
                key = HindiDict.FirstOrDefault(hword => hword.Value == s).Key;
                Rec = dict.FirstOrDefault(word => word.Key == key).Value;
                if (Rec != null){if (ss == "")
                        _atch += Rec;
                    else
                        _atch += String.Format("{0} {1}", Rec, SplitFunc(ss));}

                else
                {
                    if (ss == "")
                        _atch += s;
                    else
                        _atch += String.Format("{0} {1}", s, SplitFunc(ss));
                }

                return _atch;
            }
        }
    }
}
```