Speech recognition and synthesis

Introduction

Speech technology for "disadvantaged" languages

- Language barriers limit access to digital resources
- Speech technology needed for access to services, eg, phone services
- Language often part of national, cultural, and political identity
- Lack of Language and Speech technology will put communities at a disadvantage
- Many speech technology projects for "minority" languages started by single "students" of the language

Introduction

Basic speech technology projects

- Demonstration TTS or ASR systems *can* be build by a single person
- All tools available on the internet for free
- Basic systems for a new language take around 3-6 person months
- Systems and work are modular
- Systems should be constructed iteratively
- Start with an existing system, and change it gradually
- If digital resources are available, *use them!*

See `http://www.fon.hum.uva.nl/IFA-publications/Others/Other_papers.html`

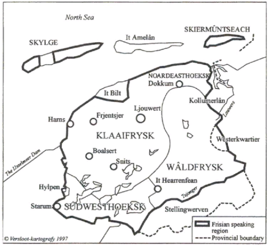
Example 1: A basic Frisian TTS

Master's thesis

- No speech technology available for Frisian
- Language community is organized scientifically
- There is "political" demand for Frisian Language Technology
- Student is a native speaker
- 4 Month thesis project
- Dutch diphones (no time to create Frisian set)
- Aim: "bootstrap" the development of a TTS system

[Dijkstra et al.(2005)Dijkstra, Pols, and van Son]

Example 1: A basic Frisian TTS: West Frisian dialects in the Netherlands



Map 1: Dialect map of Fryslân (Versloot cartography 1997, in: Visser, 1997)

West Germanic language (Indo-European)

- Main dialects: Klaaifrysk, Wàldfrysk, and Sûd-Westhoeksk
- Standard Frisian based on Klaaifrys
- Official status since 1970

[Dijkstra et al.(2005)Dijkstra, Pols, and van Son]

Example 1: A basic Frisian TTS

Total population of *Friesland* $> 634,000$ [Gorter and Jonkman(1995)]

- 55% native speakers (350,000)
- 74% understands Frisian (470,000)
- 65% reads Frisian (410,000)
- 17% writes in Frisian (110,000)

[Dijkstra et al.(2005)Dijkstra, Pols, and van Son]

Example 1: A basic Frisian TTS

Start with an existing system, and change it gradually

- *Frisian* is close to *Dutch* in many respects
- *Nextens* and those that build it were available
- Contacts with the *Fryske Akademy* could supply language help
- A digital pronunciation lexicon could be "borrowed"
- Technical and community support were available

Example 1: A basic Frisian TTS

Language resources and tools

- *Fryske Akademy*
- MBROLA [MBROLA(2005)]
- Nextens [Nextens(2003)]
- Festival [Black and Lenzo(2003a)]
- Pre-publication of "Frysk Hânwurdboek" (Concise dictionary)
- Worldbet [Hieronymus(1994)]
- Enthusiasm from everyone

Example 1: A basic Frisian TTS: Nextens

The architecture of NeXTeNS (Festival):

- **Token Module**: Tokenization
- **POS Module**: Part-Of-Speech tagging
- **Syntactic Module**: Syntax parsing (*disabled*)
- **Phrasing Module**: Phrase break prediction
- **Intonation Module**: Sentence accents
- **Tune Module**: Tune choice needed for ToDI
- **Word Module**: Grapheme-to-phoneme conversion
- **Pauses Module**: Insertion of pause segments
- **Postlexical Module**: Anything left over
- **Duration Module**: Segment and pause-durations
- **Fundamental frequency control**: ToDI ⇒ utterance
- **Waveform synthesis**

Example 1: A basic Frisian TTS

Building a Frisian TTS

- Construct Frisian Worldbet phonetic alphabet [Hieronymus(1994)]
- Convert pronunciation lexicon to Worldbet
- Phrasing, Tune, Pause: Use Dutch (small adaptations)
- Tokenization: Enter Frisian numbers and abbreviations
- POS: Translated Dutch *Function* wordlist
- POS: Use only *Content/Function* word difference
- Intonation: Accent every other *Content* word

Example 1: A basic Frisian TTS

Building a Frisian TTS: Word module

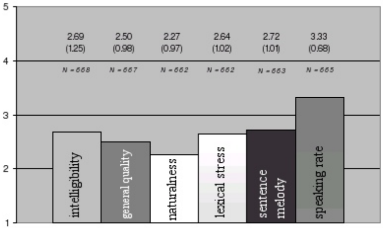
- Pronunciation lexicon
- Letter-to-Sound rules, eg,
  `(VOWEL [- g] VOICEDC = - G)`
- Syllable stress rules, i.e. strong/weak syllables
- Map complex sounds, eg, nasalized vowels and triphthongs

Example 1: A basic Frisian TTS

Building a Frisian TTS: Other modules

- Postlexical: Adapted Dutch rules
- Postlexical: Map Frisian worldbet to Dutch SAMPA symbols
- Duration: Shorten schwa, change duration long vowels
- Fundamental frequency: Adapt Dutch ToDI module
- Waveform synthesis: Map each "Frisian" phone to the "nearest" Dutch MBROLA phone

Example 1: A basic Frisian TTS: Evaluation



Mean judgments for 20 test sentences

- End evaluation over WWW with 32 native subjects
- 10 short ($<$ 13 words) 10 long ($\geq$ 13 words)
- Example of short and long sentence
- 6 qualities on a 5 point scale (higher is better/more rapid)

Example 1: A basic Frisian TTS

	short $N \approx 331$	long $N \approx 331$	total $N \approx 662$
intelligibility	2.57 (1.25)	2.80 (1.24)	2.69 (1.25)
quality	2.51 (0.99)	2.50 (0.97)	2.50 (0.98)
naturalness	2.31 (0.97)	2.22 (0.97)	2.27 (0.97)
lexical stress	2.67 (1.05)	2.58 (0.99)	2.64 (1.02)
sentence melody	2.79 (0.99)	2.64 (1.02)	2.72 (1.01)
speaking rate	3.30 (0.65)	3.35 (0.71)	3.33 (0.68)

Mean judgments (standard deviation)

- Mean ratings *below 3* (mid-point)
- *Naturalness rated* lowest
- Sentence length did not change ratings
- Ratings were above *1!*
- Note: This was done using a Dutch diphone set

Judgments on a 5 point scale, higher is better. For speaking rate higher is more rapid

Example 1: A basic Frisian TTS

	short $N \approx 331$	long $N \approx 331$	total $N \approx 662$
intelligibility	2.57 (1.25)	2.80 (1.24)	2.69 (1.25)
quality	2.51 (0.99)	2.50 (0.97)	2.50 (0.98)
naturalness	2.31 (0.97)	2.22 (0.97)	2.27 (0.97)
lexical stress	2.67 (1.05)	2.58 (0.99)	2.64 (1.02)
sentence melody	2.79 (0.99)	2.64 (1.02)	2.72 (1.01)
speaking rate	3.30 (0.65)	3.35 (0.71)	3.33 (0.68)

Mean judgments (standard deviation)

- Mean ratings *below 3* (mid-point)
- *Naturalness rated* lowest
- Sentence length did not change ratings
- Ratings were above *1!*
- Note: This was done using a Dutch diphone set

Judgments on a 5 point scale, higher is better. For speaking rate higher is more rapid

Example 1: A basic Frisian TTS

	short $N \approx 331$	long $N \approx 331$	total $N \approx 662$
intelligibility	2.57 (1.25)	2.80 (1.24)	2.69 (1.25)
quality	2.51 (0.99)	2.50 (0.97)	2.50 (0.98)
naturalness	2.31 (0.97)	2.22 (0.97)	2.27 (0.97)
lexical stress	2.67 (1.05)	2.58 (0.99)	2.64 (1.02)
sentence melody	2.79 (0.99)	2.64 (1.02)	2.72 (1.01)
speaking rate	3.30 (0.65)	3.35 (0.71)	3.33 (0.68)

Mean judgments (standard deviation)

- Mean ratings *below 3* (mid-point)
- *Naturalness rated* lowest
- Sentence length did not change ratings
- Ratings were above *1!*
- Note: This was done using a Dutch diphone set

Judgments on a 5 point scale, higher is better. For speaking rate higher is more rapid

Example 1: A basic Frisian TTS

	short $N \approx 331$	long $N \approx 331$	total $N \approx 662$
intelligibility	2.57 (1.25)	2.80 (1.24)	2.69 (1.25)
quality	2.51 (0.99)	2.50 (0.97)	2.50 (0.98)
naturalness	2.31 (0.97)	2.22 (0.97)	2.27 (0.97)
lexical stress	2.67 (1.05)	2.58 (0.99)	2.64 (1.02)
sentence melody	2.79 (0.99)	2.64 (1.02)	2.72 (1.01)
speaking rate	3.30 (0.65)	3.35 (0.71)	3.33 (0.68)

Mean judgments (standard deviation)

- Mean ratings *below 3* (mid-point)
- *Naturalness rated* lowest
- Sentence length did not change ratings
- Ratings were above *1!*
- Note: This was done using a Dutch diphone set

Judgments on a 5 point scale, higher is better. For speaking rate higher is more rapid

Example 1: A basic Frisian TTS

	short $N \approx 331$	long $N \approx 331$	total $N \approx 662$
intelligibility	2.57 (1.25)	2.80 (1.24)	2.69 (1.25)
quality	2.51 (0.99)	2.50 (0.97)	2.50 (0.98)
naturalness	2.31 (0.97)	2.22 (0.97)	2.27 (0.97)
lexical stress	2.67 (1.05)	2.58 (0.99)	2.64 (1.02)
sentence melody	2.79 (0.99)	2.64 (1.02)	2.72 (1.01)
speaking rate	3.30 (0.65)	3.35 (0.71)	3.33 (0.68)

Mean judgments (standard deviation)

- Mean ratings *below 3* (mid-point)
- *Naturalness rated* lowest
- Sentence length did not change ratings
- Ratings were above *1!*
- Note: This was done using a Dutch diphone set

Judgments on a 5 point scale, higher is better. For speaking rate higher is more rapid

Example 2: Digit recognition in two languages

Kinyarwanda: Official language of Rwanda

- Niger-Congo Language
  `http://www.nvtc.gov/lotw/months/september/niger.html`
- 7-8 million native speakers
- Many Rwandese are monolingual
- Recognizer build by *Muhirwe Jackson* for his Master of Science thesis [Jackson(2005)]
- Computer Science of Makerere University, Kampala, Uganda
- Implements the tutorial digit recognizer from HTK

[Jackson(2005)]
[Young et al.(2004)Young, Evermann, Hain, Kershaw, Moore, Odell, Ollason, Povey, Valtchev, and Woodland]

[HTK(2002)]

Example 2: Digit recognition in two languages

Dutch: Official language of the Netherlands

- West Germanic language, 21 million native speakers
- Masters of Science course for AI students
- University of Amsterdam
- Speech Technology project
- 1 month, 6 students [Adriaans et al.(2004)Adriaans, Heukelom, Koolen, Lentz, de Rooij, and Vreeswijk]
- Implements the tutorial telephone application from HTK

[Adriaans et al.(2004)Adriaans, Heukelom, Koolen, Lentz, de Rooij, and Vreeswijk]
[Young et al.(2004)Young, Evermann, Hain, Kershaw, Moore, Odell, Ollason, Povey, Valtchev, and Woodland]

[HTK(2002)]

Building a basic ASR system

Tasks

1. Building the task grammar
2. Constructing a dictionary for the models
3. Recording the data.
4. Creating transcription files for training data
5. Encoding the data (feature processing)
6. (Re-) training the acoustic models
7. Evaluating the recognizers against the test data
8. Reporting recognition results

Building a basic ASR system

```
# Kinyarwanda
$digit=RIMWE | KABIRI | GATATU | KANE | GATANU | GATANDATU | KARINDWI | UMUNANI
| ICYENDA | ZERO;
(SENT-START [ $digit ] SENT-END)

# Dutch
$digit = EEN | TWEE | DRIE | VIER | VIJF | ZES | ZEVEN | ACHT | NEGEN | NUL;
$name = [ ROB ] (VAN SON) | [ FRANS ] ADRIAANS | [ TOM ] LENTZ | [ MARIJN |
MARINUS ] KOOLEN | [ ORK ] (DE ROOIJ) | [ MARKUS ] HEUKELOM | [ DAAN ]
VREESWIJK;
( SENT-START ( DRAAI <$digit> | BEL $name) SENT-END )
```

Task Grammars

- Define digits and names
- Define grammar on vocabular
- Square brackets enclose optional items

Building a basic ASR system

```
# Kinyarwanda
$digit=RIMWE | KABIRI | GATATU | KANE | GATANU | GATANDATU | KARINDWI | UMUNANI
| ICYENDA | ZERO;
(SENT-START [ $digit ] SENT-END)

# Dutch
$digit = EEN | TWEE | DRIE | VIER | VIJF | ZES | ZEVEN | ACHT | NEGEN | NUL;
$name = [ ROB ] (VAN SON) | [ FRANS ] ADRIAANS | [ TOM ] LENTZ | [ MARIJN |
MARINUS ] KOOLEN | [ ORK ] (DE ROOIJ) | [ MARKUS ] HEUKELOM | [ DAAN ]
VREESWIJK;
( SENT-START ( DRAAI <$digit> | BEL $name) SENT-END )
```

Task Grammars

- Define digits and names
- Define grammar on vocabular
- Square brackets enclose optional items

Building a basic ASR system

```
# Kinyarwanda
$digit=RIMWE | KABIRI | GATATU | KANE | GATANU | GATANDATU | KARINDWI | UMUNANI
| ICYENDA | ZERO;
(SENT-START [ $digit ] SENT-END)

# Dutch
$digit = EEN | TWEE | DRIE | VIER | VIJF | ZES | ZEVEN | ACHT | NEGEN | NUL;
$name = [ ROB ] (VAN SON) | [ FRANS ] ADRIAANS | [ TOM ] LENTZ | [ MARIJN |
MARINUS ] KOOLEN | [ ORK ] (DE ROOIJ) | [ MARKUS ] HEUKELOM | [ DAAN ]
VREESWIJK;
( SENT-START ( DRAAI <$digit> | BEL $name) SENT-END )
```

Task Grammars

- Define digits and names
- Define grammar on vocabular
- Square brackets enclose optional items

Building a basic ASR system

Construct pronunciation dictionary

- Make a word list of all words in the training corpus or a suitable text corpus
- Transcribe the words by hand or use a TTS system (eg, *Nextens*)
- Feed the lexicon to HTK [HTK(2002)]

Building a basic ASR system

Generate prompts and record utterances

- Use task grammar to generate random prompts
- Record as many users as possible reading the prompts
- Better, subjects repeat synthesized (TTS) prompts
- Transcribe all prompts and all sentences in the corpus

Building a basic ASR system

Training

- Transcribe and (feature) encode utterances
- Feed as much speech as possible to the HTK training
- Kinyarwanda uses 3 male and 3 female speakers, 150 sentences
- Words were hand-labeled
- Dutch uses 1000 labeled sentences from the IFAcorpus (4 male, 4 female speakers)
- Dutch recorded 150 task sentences from 4 male speakers (total 600)
- Recorded utterances were transcribed automatically
- Put all files in correct format and fire up HTK training [HTK(2002)]

ASR evaluation: Kinyarwanda

Subject	Words correct	Substitution errors	Percentage
Subject 1	9	1	90%
Subject 2	8	2	80%
Subject 3	8	2	80%
Subject 4	8	2	80%

Live data recognition results

- 4 New subjects
- Read out all 10 numbers
- HTK self-test results (*not* live):
- Sentence Recognition Rate: 92.00% (N=50)
- Word Recognition Rate: 94.87% (N=156)

ASR evaluation: Dutch

TRAINED ON	TESTED ON
IFA + Domain	Domain, training speakers
IFA + Domain	Domain, 'unknown' speaker
IFA + Domain	New sentences, training speakers
IFA + Domain	New sentences, new speaker

Testing procedures

- Two corpora: IFA corpus and Domain corpus
- Testing using randomly selected sentences
- Test set not used during training

ASR evaluation: Dutch

Left Out %	WORD RECOGNITION (%)	SENTENCE RECOGNITION (%)
10	99.71	91.38
20	99.46	92.31
50	99.67	89.93
80	99.66	89.18

Testing on random sentences

- Leave out random sentences and train
- Test randomly selected sentences
- Smaller training set affects Sentence Recognition most

ASR evaluation: Dutch

LEFT OUT SPEAKER	WORD RECOGNITION (%)	SENTENCE RECOGNITION (%)
Tom	99.57	85.71
Markus	99.78	72.60
Ork	99.43	89.13
Frans	99.78	81.63
LEFT OUT PERCENTAGE	WORD RECOGNITION (%)	SENTENCE RECOGNITION (%)
12	99.41	92.86
25	99.80	90.57
50	99.84	89.35

Top: Testing on a new speaker, Bottom: Testing on new sentences

New speakers are worse than new sentences

- More speakers needed for independence
- Sentence recognition drops sharply
- New speaker *and* new sentences
  Recognition: Word - 99.57%, Sent - 84.35%

Conclusion

Simple TTS and ASR can be done in a few months

- Free tools are available
- People like it when their language is used
- Recording speech is the most laborous step
- More speech is better, as is more text
- Pronunciation dictionaries are crucial

Further Reading I

Frans Adriaans, Markus Heukelom, Marijn Koolen, Tom Lentz, Ork de Rooij, and Daan Vreeswijk.
Speech Technology Project 2004 Building an HMM Speech Recogniser for Dutch.
Technical report, Masters of AI, Faculty of Science, University of Amsterdam, 9 July 2004.
URL http://www.fon.hum.uva.nl/IFA-publications/Others/Other_papers.html.

Guillermo Aradilla, Jithendra Vepa, and Hervé Bourlard.
Improving speech recognition using a data-driven approach.
IDIAP-RR 66, IDIAP, Martigny, Switzerland, 2005.

Alan W. Black and Kevin A. Lenzo.
Festvox.
Web, 2003.
URL http://festvox.org/.
Festival speech synthesis.

P. Boersma.
Praat, a system for doing phonetics by computer.
Glot International, 5:341–345, 2001.
URL http://www.Praat.org/.

P. Boersma and D. Weenink.
Praat 4.2: doing phonetics by computer.
Computer program: http://www.Praat.org/, 2004.
URL http://www.Praat.org/.

Further Reading II

M. De Wachter, M. Matton, K. Demuynck, P. Wambacq, R. Cools, and D. Van Compernolle.
Template-Based Continuous Speech Recognition.
IEEE TRANSACTIONS ON AUDIO, SPEECH & LANGUAGE PROCESSING, 15(4):1377–1390, 2007.
ISSN 1558-7916.

Jelske Dijkstra, Louis C.W. Pols, and R.J.J.H. van Son.
Frisian TTS, an example of bootstrapping tts for minority languages.
In *Proceedings of the 5th ISCA Speech Synthesis Workshop - Pittsburgh*, pages 97–102, 2005.
URL `http://www.ssw5.org/`.

FSF.
GNU General Public License.
Web, June 1991.
URL `http://www.gnu.org/licenses/gpl.html`.

D. Gorter and R.J. Jonkman.
Taal yn Fryslân op 'e nij besjoen.
Fryske Akademy, Ljouwert, 1995.

James L. Hieronymus.
Ascii phonetic symbols for the world's languages: Worldbet.
Web, 1994.
URL `http://www.ling.ohio-state.edu/~edwards/worldbet.pdf`.

Further Reading III

HTK.
Hidden Markov Model Toolkit.
Web, December 2002.
URL http://htk.eng.cam.ac.uk.
Toolkit distribution.

Muhirwe Jackson.
Automatic speech recognition: Human computer interface for Kinyarwanda language.
Master's thesis, Faculty of computing and Information Technology, Makerere University, August 2005.
URL http://www.fon.hum.uva.nl/IFA-publications/Others/Other_papers.html.

MBROLA.
The MBROLA Project.
Web, 2005.
URL http://tcts.fpms.ac.be/synthesis/.
Synthesis.

Nextens.
NeXTeNS: Open Source Text-to-Speech for Dutch.
Web, 2003.
URL http://nextens.uvt.nl/index.html.

Keiichi Tokuda, Heiga Zen, and Alan W. Black.
An HMM-based speech synthesis system applied to English, 2002.
URL http://www.cs.cmu.edu/~awb/papers/IEEE2002/hmmenglish.pdf.

Further Reading IV

S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland.
The HTK Book.
Cambridge University Engineering Department, December 2004.
URL http://htk.eng.cam.ac.uk.
Part of the HTK distribution.

Appendix A

Copyright License

The GNU General Public License I

Version 2, June 1991
Copyright © 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

The GNU General Public License II

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

Terms and Conditions For Copying, Distribution and Modification

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".
   Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

The GNU General Public License III

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.
   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
   1. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
   2. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
   3. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

The GNU General Public License IV

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
    1. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
    2. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
    3. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The GNU General Public License V

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

The GNU General Public License VI

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
   If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
   It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
   This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
   Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

The GNU General Public License VII

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

No Warranty

11. Because the program is licensed free of charge, there is no warranty for the program, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holders and/or other parties provide the program "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the program is with you. Should the program prove defective, you assume the cost of all necessary servicing, repair or correction.
12. In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may modify and/or redistribute the program as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.

End of Terms and Conditions

The GNU General Public License VIII

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

> *one line to give the program’s name and a brief idea of what it does.*
> *Copyright (C) yyyy name of author*
> *This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.*
>
> *This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.*
>
> *You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.*

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

> *Gnomovision version 69, Copyright (C) yyyy name of author*
> *Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type ‘show w’.*
> *This is free software, and you are welcome to redistribute it under certain conditions; type ‘show c’ for details.*

The GNU General Public License IX

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items—whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

> *Yoyodyne, Inc., hereby disclaims all copyright interest in the program*
> *'Gnomovision' (which makes passes at compilers) written by James Hacker.*
> *signature of Ty Coon, 1 April 1989*
> *Ty Coon, President of Vice*

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.