Lexical Tools
ASCII Conversion

Dr. Chris J. Lu

The Lexical Systems Group
NLM. LHNCBC. CGSB

March, 2011
Table of Contents

• Introduction
• ASCII conversion
  ▪ Character
  ▪ Document
  ▪ Corpus
  ▪ Software/APIs
  ▪ Example
• Questions
ASCII Character Set

• ** ASCII:** [American Standard Code for Information Interchange](http://example.com)
• Contains 128 7-bit coded characters
• Value range: U+0000 ~ U+007F
• Includes:
  • alphabetic characters: A, B, C, …
  • numeric characters: 0, 1, 2, 3, …
  • control characters: ESC, FS, CR, …
  • graphic characters: #, $, %, &, *, (, ), ..
• The most common used standard code (before Unicode)
Unicode

• A character encoding specification published by the Unicode Consortium
• Includes all of the major world’s writing systems
• Becomes the industry standard
• Allows data to be transported through different systems
• Very useful when dealing with multilingual NLP
• Latest version Unicode 6.0.0, 2011
Unicode Transformation Format

• Unicode Encoding
  ▪ Including UTF-7, UTF-8, UTF-16, UTF-32

• UTF-8 has become the dominant character encoding
  ▪ Backward-compatible with ASCII
  ▪ Avoiding the complications of endianness
  ▪ No need to use byte order marks (BOM)
Lexicon & Lexical Tools

• Released in UTF-8 format since 2006
• Provides functions to convert UTF-8 to ASCII
  ▪ Character
  ▪ Text
  ▪ Document
Why ASCII Conversion?

• Non-ASCII Unicode are commonly seen even in English documents, such as “Déjà Vu “, “Café”, “resumé”, etc.

• Some NLP projects still only deal with ASCII
The Challenges

• Not one-to-one mapping:
  ▪ Many to one: å, â, ã, á, à, ä to a
  ▪ One to many: © to ![COPYRIGHT SIGN]!, (c), or just simply removed
  ▪ One to none: French borrowing “divorcé” means a man who is divorced. This word has no pure ASCII spelling variant in Webster’s Dictionary, while the converted ASCII word, “divorce”, is another closely related word

• Misused Unicode characters (before the conversion)
  ▪ µ (mu, U+03BC) and μ (micro sign, U+00B5)
  ▪ β (Sharp S, U+00DF) and β (beta, U+03B2)
  ▪ ¶ (Pilcrow Sign, U+00B6) and π (PI, U+03C0)

• Wrong conversions (meaning changed)
  ▪ © to (c): copyright or cellular phone number?
  ▪ divorcé to divorce
Conversion Guidelines

- Preserve semantic and/or graphic representation
- Example ™:
  - Graphic: TM
  - Semantic: ![TRADE MARK SIGN]!
  - Graphic and Semantic: (TM), or (tm)
  - NLP: empty string, consider ™ as a stopword

- Different NLP applications might apply different methods due to different requirements and objectives
- There is no best method for ASCII conversion
Character Conversion

- Strip diacritics:
  å, â, ã, á, à, ä, ê, é, è, ë, î, í, ì, ï, ô, õ, ó, ø, ò, ö, û, ú, ü, ý, ç, ñ, etc.

- Split ligatures:
  Æ, æ, Ø, œ, ff, fl, ffi, etc.

- Punctuation mapping:
  “double quotation”, ‘single quotation’, –, -, etc.

- Symbols mapping:
  ©, ®, ™, °, ÷, ≤, ≥, etc.

- Combinations:
  æ [U+01FD], Dž [U+01C5], ¾ [U+00BE], etc

- Others:
  α, β, etc
### Lexical Tools

- **Unicode related functions (flow components)**

<table>
<thead>
<tr>
<th>LVG Flow</th>
<th>Description</th>
<th>Input (UTF-8)</th>
<th>Output (ASCII)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-f:q</td>
<td>Strips diacritic</td>
<td>Déjà Vu</td>
<td>Deja Vu</td>
</tr>
<tr>
<td>-f:q0</td>
<td>Symbols &amp; punctuation</td>
<td>“Quote”</td>
<td>&quot;Quote&quot;</td>
</tr>
<tr>
<td>-f:q1</td>
<td>Unicode mapping</td>
<td>⅔</td>
<td>2/3</td>
</tr>
<tr>
<td>-f:q2</td>
<td>Splits ligatures</td>
<td>spælsau</td>
<td>spaelsau</td>
</tr>
<tr>
<td>-f:q3</td>
<td>Unicode names</td>
<td>©</td>
<td>![COPYRIGHT SIGN]!</td>
</tr>
<tr>
<td>-f:q4</td>
<td>Unicode Synonym</td>
<td>μ (mu, U+03BC)</td>
<td>μ (Micro sign, U+00B5)</td>
</tr>
<tr>
<td>-f:q5</td>
<td>Normalize Unicode</td>
<td>UMLS®</td>
<td>UMLS![REGISTERED SIGN]!</td>
</tr>
<tr>
<td></td>
<td>(-f:q7:q3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-f:q6</td>
<td>Normalize Unicode w Synonyms</td>
<td>UMLS®</td>
<td>UMLS![REGISTERED SIGN]!</td>
</tr>
<tr>
<td></td>
<td>(-f:q4:q7:q3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-f:q7</td>
<td>Core Norm</td>
<td>Æ</td>
<td>AE</td>
</tr>
<tr>
<td></td>
<td>(recursive -f:q0:q1:q2:q)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-f:q8</td>
<td>Strip or Map (not ICU)</td>
<td>Zadaxin™</td>
<td>Zadaxin</td>
</tr>
<tr>
<td>-f:q8</td>
<td>Strip or Map (not ICU)</td>
<td>α</td>
<td>alpha</td>
</tr>
</tbody>
</table>
Lexical Tools (Cont.)

- Pure ASCII conversion

<table>
<thead>
<tr>
<th>LGF Flow(s)</th>
<th>Desc.</th>
<th>Pure ASCII</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>-f:q5</td>
<td>Normalize Unicode</td>
<td>Yes</td>
<td>Single</td>
</tr>
<tr>
<td>-f:q6</td>
<td>Normalize Unicode with Synonyms</td>
<td>Yes</td>
<td>Single</td>
</tr>
<tr>
<td>-f:N</td>
<td>Normalize</td>
<td>Yes</td>
<td>Multiple</td>
</tr>
<tr>
<td>-f:N3</td>
<td>Lui-Norm</td>
<td>Yes</td>
<td>Single</td>
</tr>
<tr>
<td>-f:q7:q8</td>
<td>Serial Flows</td>
<td>Yes</td>
<td>Single</td>
</tr>
<tr>
<td>ToAscii</td>
<td>ASCII conversion</td>
<td>Yes</td>
<td>Single</td>
</tr>
</tbody>
</table>
Text Conversion

• Many different ways for ASCII conversion
• The SPECIALIST Lexical Tools
  ▪ Provides various powerful functions
  ▪ Is configurable according to the specifications
  ▪ Use ToAscii

Free Text (Unicode) \rightarrow Lexical Tools (ToAscii) \rightarrow Free Text (ASCII)
Corpus Conversion

- ToAscii
- Algorithm from domain experts

Corpus (Unicode) → Corpus (ASCII)
Corpus Conversion - Lexicon

Conversion Algorithm

- ToAscii
- Delete if it is new
- Delete if it is duplicated
- Delete if it has a different meaning
Delete: If New

- Delete the conversion if it is new (not known to Lexicon)
  - Theoretically, the ASCII Lexicon is a subset of Unicode Lexicon since ASCII is a subset of Unicode
  - All converted bases should be known to (contained inside) Lexicon
- Example - Müthing” [E0573093]:
  - The record is deleted ("Muthing" is not know to Lexicon)

```json
{base=Müthing
 entry=E0573093
  cat=noun
  variants=reg
  variants=uncount
  proper
}
```

```json
{base=Muthing
 entry=E0573093
  cat=noun
  variants=reg
  variants=uncount
  proper
}
```
Delete: If Duplicated

- Delete the conversion if it is a duplication
- Example – resume [E0053099]
  - Spelling variants are removed

```json
{base=resume
 spelling_variant=résumé
 spelling_variant=resumé
 entry=E0053099
   cat=noun
   variants=reg
}

{base=resume
 spelling_variant=resume
 spelling_variant=resume
 entry=E0053099
   cat=noun
   variants=reg
}
```
Delete: If Meaning Changed

- Delete the conversion if it has a different meaning
- Example – mu [E0041164]:
  - Spelling variant “μm” is deleted because its ASCII conversion, “mum” [E0041369], is a different record

```json
{base=mu
 spelling_variant=μ
 spelling_variant=μm
 entry=E0041164
  cat=noun
  variants=inv
  variants=metareg
  abbreviation_of=micrometer|E0040123
 }
```

```json
{base=mum
 spelling_variant=mum
 entry=E0041369
  cat=noun
  variants=reg
 }
```
NLP Software Conversion

ASCII NLP Project
Software Components
- Data out (ASCII)
  ...
- Data in for further process

NLP Software/APIs (Unicode)
- Algorithm
- Unicode Data

Results from APIs (Unicode)

• Traditional approach
• Interface approach
Traditional Approach

- This traditional approach is tedious and not practical
Interface Approach

• The interface approach is easy and generic
Application Example

Traditional Approach
Lexical Tools APIs (Unicode)
- Algorithm
- ASCII data (Db tables)

Results from APIs (ASCII)

Interface Approach
Lexical Tools API (Unicode)
- Algorithm
- Unicode data

Results from Lexical Tools
- ToAscii
  - Remove unknown conversions
  - Remove duplicated conversions

ASCII NLP Project (MetaMap)
Software Component
- Data out (ASCII)
...
- Data in for further process
...

• Identical results from both approaches over 0.5M test cases for 2010 release
References

• Unicode Consortium - [http://www.unicode.org](http://www.unicode.org)
• ICU (International Components for Unicode) - [http://site.icu-project.org](http://site.icu-project.org)
Questions