NLP와 기계번역: 통계적 기법과 머신러닝

2017년 6월 26일

강승식
국민대학교 소프트웨어학부
What is Machine Translation?

Source Input

I'm originally from Dublin but now live in Edinburgh.

Target Output

मैं डबलिन से मूल हूँ, लेकिन अब एडिनबर्ग में रहते हैं.

Machine Translation
Topics

• MT history and NLP
• Intro. to NLP techniques
• Intro. to Machine Translation
• Statistical MT
• Machine Learning Approach
History of MT and NLP

• 7 January 1954
  • The first public demonstration of a Russian-English MT in New York, IBM
  • Having just 250 words and translating just 49 Russian sentences into English.
  • Rough translation of Russian scientific journals in order to intercept secret information.

• Early 1970s
  • Russian-English project called SYSTRAN
  • An attempt to translate a vast body of terminology connected with the military
A Critical Problem of MT

• The spirit is willing, but the flesh is weak

• The vodka is good, but the steak is lousy
The Goal of Machine Translation

- Automatic translation of all kinds of documents
- At a quality of the best human translators
- In fact, this goal was impossible!
기계번역 vs. 자동통역

• 문어체 vs. 대화체
  • 문서번역 vs. 대화통역(동시 통역, 실시간 통역)

• 기계번역의 유형
  • Fully Automatic MT
  • Human-Assisted MT (HAMT)
  • Machine-Assisted Human Translation (MAHT)
    • MT Workbench
기계번역 필요성

출처: ETRI, 전자통신동향분석, 제20권 제5호, 2005년 10월.
Who is winning the race in translation?

- Google Translate
  https://translate.google.co.kr/

- Babylon
  http://translation.babylon-software.com/english/to-korean/

- Jibbigo
  http://jibbigo-translator-2-0.soft112.com/

- iLingual: French, German, Spanish, Arabic
If it is an online translator you need, you have just found the best and it is free! Babylon, the world’s leading provider of language solutions, puts at your disposal an automatic translator for translating single words, full texts, phrases and more. Search for literally millions of terms in Babylon Software’s database of over 1,700 dictionaries, glossaries, thesauri, encyclopedias and lexicons covering a wide range of subjects, all in more than 77 languages.
Translation Examples

• If it is an online translator you need, you have just found the best and it is free!

• 그것은 필요한 온라인 Translator 경우 찾았을지도 최고의 무료!

• 당신이 필요로하는 온라인 번역가 인 경우에, 당신은 지금 베스트를 찾아 내고 자유 롭다!

• 만약에 이것이 당신이 필요로 하는 온라인 번역기라면, 당신은 바로 가장 좋은 것을 찾았고 이것은 무료입니다.
• Babylon, the world's leading provider of language solutions,

• 바벨론, 세계 언어 솔루션을 공급하는 선도 업체로서

• 세계 최고의 언어 솔루션 제공 업체인 바빌론 (Babylon)은

• 세계의 주도적인 언어 솔루션 공급자인 바빌론은
  (국제적으로 언어 솔루션을 주도적으로 공급하는 바빌론은)
• puts at your disposal an automatic translator for translating single words, full texts, phrases and more.

• 고객의 편의대로 이용하실 단일 단어를 번역하는 자동 번역, 전체 글귀, 구절 등을 배치합니다.

• 한 단어, 전문을 번역하는 자동 번역기를 제공합니다.

• 한 단어, 전문, 구 등을 번역하기 위한 자동번역기를 당신이 처분할 수 있게 해줍니다.
• Search for literally millions of terms in Babylon Software’s database of over 1,700 dictionaries, glossaries, thesauri, encyclopedias and lexicons covering a wide range of subjects; all in more than 77 languages.

• 바빌론에서 소프트웨어의 1700여 사전, 또는 메뉴별, thesauri, 백과사전 신민들의 광범위한 용어 데이터베이스 용어 말 그대로 수백만 검색, 모두 77개 이상의 언어로.

• 바빌론 소프트웨어의 1,700 개가 넘는 사전, 용어집, 시소러스, 백과 사전 및 광범위한 주제를 다루는 어휘집으로 이루어진 수백만 단어를 문자 그대로 검색하십시오. 모두 77 개 이상의 언어로 제공됩니다.

• 넓은 범위의 주제들을 포괄하는 1,700개 이상의 사전과 용어집, 시소러스, 백과사전, 어휘사전을 보유하고 있는 바빌론 소프트웨어 데이터베이스에서 수백만개의 용어들을 검색해 보세요. 모두 77개 이상의 언어로.
How many languages? 104
기계 번역 방법 예제

나는 뉴턴을 읽었다

대명사 조사
주어

명사 조사
목적어

동사

솔어

대명사 동사
명사

I read Newton
M.T. Approaches

• Direct Translation
• Rule-Based M.T.
  • Transfer-based Approach
  • Interlingua/Pivot Approach
• Corpus-Based M.T.
  • Statistical M.T. (SMT)
  • Example-Based M.T. (EBMT)
• Knowledge-Based M.T.
• Neural Network Approach
Traditional MT approaches

- Transfer-based
- Interlingua
- Example-based (EBMT)
- Statistical MT (SMT)
- Hybrid approach
Direct Translation

- In direct translation - as shown in the diagram: 도우치 준이치 지음, 최기선 옮김, 미래사, 1992, Page 129~141
Transfer Approach

• Number of translators: $N \times N$
• **Analysis, transfer, generation:**
  1. Parse the source sentence
  2. Transform the parse tree with transfer rules
  3. Translate source words
  4. Get the target sentence from the tree

• **Resources required:**
  • Source parser
  • A translation lexicon
  • A set of transfer rules
Example: Korean-to-English

Diagram 7: downloadable file
Issues in Transfer-based MT

- **Parsing:** linguistically motivated grammar or formal grammar?
- **Transfer:**
  - context-free rules? A path on a dependency tree?
  - Apply at most one rule at each level?
  - How are rules created?
- **Translating words:** word-to-word translation?
- **Generation:** using LM or other additional knowledge?

- How to create the needed resources automatically?
- For n languages, we need n(n-1) MT systems!
Interlingua Approach

• Language-independent representation of a sentence

• We only need \( n \) analyzers, and \( n \) generators.

• Resource needed:
  • A language-independent representation
  • Sophisticated analyzers
  • Sophisticated generators
Interlingua/Pivot Approach

• Esperanto like intermediate representation
Analysis & Generation

- Number of translators: N + N
Interlingua: Pivot Approach
Direct, Transfer, and Interlingua
Issues in Interlingua

• Language-independent meaning representation really exist? If so, what does it look like?
• It requires deep analysis: how to get such an analyzer: e.g., semantic analysis
• It requires non-trivial generation: How is that done?
• It forces disambiguation at various levels: lexical, syntactic, semantic, discourse levels.
NLP and Machine Translation is to
Analysis and Generation
NLP issues and applications
NLP Basics

• Morphological analysis(형태소 분석)
  • Word-level

• Syntactic analysis(구문 분석)
  • Sentence-level

• Semantic analysis(의미 분석)
  • Word-sense disambiguation

• Natural Language Generation(자연어 생성)

• Language Resources(언어 자원)
  • 말뭉치, WordNet, 온톨로지 등
NLP Applications

• Machine Translation, 1950’s-now

• Information Retrieval, 1980’s-now
  • Text Classification, Information Extraction
  • Text Summarization
  • Text Mining, Opinion Mining
  • Sentiment Classification (감성 분류)

• Natural Language Understanding, 1960-70, 2000’s
  • ELIZA: Doctor, Joseph Weizenbaum, MIT, 1965
  • SHRDLU: Robot arm, Terry Winograd, MIT, 1971
  • LUNAR
  • Ask Jeeves (ask.com), 1996
  • Wolfram alpha, 2009
• Speller and grammar checker

• Spam mail filtering, Spam 문자 filtering

• Sentiment analysis(감성 분석)

• 아이폰 시리, IBM 화슨, 자동통역 시스템

• 텍스트 마이닝, 빅데이터 분석
NLP Resources and NLTK in Python
NLP resources in http://nlp.stanford.edu/

- Stanford CoreNLP
- Stanford Parser
- Stanford POS Tagger
- Stanford Named Entity Recognizer
- Stanford RegexNER
- Stanford Word Segmenter
- Stanford Classifier
- Stanford EnglishTokenizer
- Stanford TokensRegex
- Stanford Temporal Tagger (SUTime)
- Stanford Pattern-based Information Extraction and Diagnostics (SPIED)
- Stanford Relation Extractor

- Stanford Neural Machine Translation
- Stanford Natural Language Inference Corpus (SNLI)
- Semantic Parsing with Execution (SEMPRE)
  SEMPRE is a toolkit for training semantic parsers, which map natural language utterances to denotations (answers) via intermediate logical forms.
- Stanford Open Information Extraction
  A tool for extracting open domain relation triples; e.g., "cats play with yarn" yields (cats, play with, yarn).
- GloVe: Global Vectors for Word Representations
- Deep Learning for Sentiment Analysis
- Tregex, Tsurgeon, and Semgrep
  Tools for matching patterns in linguistic trees (following the tgrep/tgrep2 tradition), a GUI for this, and a tree-transformation utility built on top of this matching language. Also, a similar utility for matching patterns in dependency graphs.
- Phrasal
  A state-of-the-art phrase-based machine translation system.
The strongest rain ever recorded in India shut down the financial hub of Mumbai, snapped communication lines, closed airports and forced thousands of people to sleep in their offices or walk home during the night, officials said today.
(ROOT (S (S (NP (DT The) (JJS strongest) (NN rain)) (VP (ADVP (RB ever)) (VBN recorded) (PP (IN in) (NP (NNP India)))))) (VP (VBD shut) (PRT (RP down)) (NP (NP (DT the) (JJ financial) (NN hub)) (PP (IN of) (NP (NNP Mumbai))))))

S (VP (TO to) (VP (VB sleep) (FP (IN in) (NP ((PRP$ their) (NNS offices))))))

(CC and) (VP (VB forced) (NP (NP (NNS thousands)) (PP (IN of) (NP (NNS people)))))

(S (VP (VBD snooped) (NP (NNS communication) (NNS lines)) (,,))) (VP (VBD closed) (NP (NNS airports)) (CC and)) (VP (VBD forced) (NP (NP (NNS thousands)) (PP (IN of) (NP (NNS people))))))

This output was generated with the command:

```java -mx200m edu.stanford.nlp.parser.lexparser.LexicalizedParser -retainTMPSubcategories -outputFormat "wordsAndTags,penn,typedDependencies" englishPCFG.ser.gz mumbai.txt
```
President Xi Jinping of China, on his first state visit to the United States, showed off his familiarity with American history and pop culture on Tuesday night.
NLTK: NLP Took Kit

• Natural Language Toolkit

• Suite of classes for several NLP tasks
  • Parsing, POS tagging, classifiers...

• Easy-to-use interfaces to over 50 corpora and lexical resources
  • [http://www.nltk.org/nltk_data/](http://www.nltk.org/nltk_data/)
Installing NLTK

• [http://www.nltk.org/install.html](http://www.nltk.org/install.html)

• Mac/Unix
  1. Install Setuptools
  2. Install Pip
  3. Install Numpy(optional)
  4. Install PyYAML and NLTK
  5. Test installation

• Windows
  1. Install Python
  2. Install Numpy(optional)
  3. Install Setuptools
  4. Install Pip
  5. Install PyYAML and NLTK
  6. Test installation
Modules

• The NLTK modules include:
  • `nltk.token` : processing individual elements of text, such as words or sentences
  • `nltk.tagger` : tagging tokens with supplemental information, such as POS or wordnet sense tags
  • `nltk.parser` : high-level interface for parsing texts
  • `nltk.classify` : classify text into categories
  • `nltk.corpus` : access (tagged)corpus data
  .......

• [http://www.nltk.org/py-modindex.html](http://www.nltk.org/py-modindex.html)
Example: POS tagging

```python
from nltk import pos_tag, word_tokenize
sent1 = "this is a demo that will show you how to detects parts of speech with little effort using NLTK!"

tokenized_sent = word_tokenize(sent1)
print pos_tag(tokenized_sent)
```

```
alicia ambiguity anger bathos blunder boards commoners correspondence
doors english france incubi it james organ oriental ours outspoken
paper perpetration
```

```
[('this', 'DT'), ('is', 'VBZ'), ('a', 'DT'), ('demo', 'NN'), ('that', 'WDT'), ('will', 'MD'), ('show', 'VB'), ('you', 'PRP'), ('how', 'WRB'), ('to', 'TO'), ('detects', 'NNS'), ('parts', 'NNS'), ('of', 'IN'), ('speech', 'NN'), ('with', 'IN'), ('little', 'JJ'), ('effort', 'NN'), ('using', 'VBG'), ('NLTK', 'NNP'), ('!', '.'), ('"", ""')]```
Example: Parsing

```python
from nltk.chunk import *
from nltk.chunk.util import *
from nltk.chunk.regexp import *
from nltk import word_tokenize
from nltk import pos_tag

text = '''Jack and Jill went up the hill to fetch a pail of water'''

tokens = pos_tag(word_tokenize(text))

chunk = ChunkRule("<.*>+", "Chunk all the text")
chink = ChinkRule("<VBD|IN>|\>", "Leave verbs and prepositions out of this")
split = SplitRule("<DT><NN>", "<DT><NN>", "Chunk on sequences of determiner+noun phr:"

chunker = RegexpChunkParser([chunk, chink, split],chunk_node='NP')
chunked = chunker.parse(tokens)
chunked.draw()
```
Example: WordNet
For more details

• NLTK
  • http://www.nltk.org/index.html

• NLTK demo site
  • http://text-processing.com/demo/
NLP Generation

• **Robot Journalism**: 스포츠, 지진, 교통, 일기예보
  - [https://automatedinsights.com/](https://automatedinsights.com/)
  - [https://www.narrativescience.com](https://www.narrativescience.com/)
NLP Generation (cont)

• **ChatBot**: dialogue analysis and generation

• Pattern match in the new programming languages
  • Scala, Swift, and Wolfram Language
NLP, Machine Learning, and Machine Translation
Machine Learning for NLP

- HMM, MEM (Maximum Entropy Model)
- kNN (k-Nearest Neighbor)
- Naïve Bayse
- SVM (Support Vector Machine)
- CRF++ (Conditional Random Field)
- Neural Network
Support Vector Machine (SVM)

- Support Vector Machine (SVM)
  - 이원(binary) 패턴 인식 문제를 해결하기 위해 제안된 학습 방법
  - 두 클래스 사이에 가장 최적의 결정면(벡터 평면)을 찾는 것이 목적

smaller margin vs maximal margin
SVM: binary classifier

• SVM light
  • Thorsten Joachims <thorsten@joachims.org>
    • Cornell University Department of Computer Science
  • An implementation of the SVMs in C.

• SVM 엔진 다운로드
  • [http://svmlight.joachims.org/](http://svmlight.joachims.org/)
  • source code:
  • Binary versions are also available for the various systems.
SVM: Install and compile

• Create a new directory
  • $ mkdir svm_light

• Move svm_light.tar.gz into svm_light and decompress
  • $ tar xzf svm_light.tar.gz

• Compile
  • $ make

• Two executables will be created.
  • svm_learn (learning module)
  • svm_classify (classification module)
Learning Module

• **svm_learn [options] example_file model_file**
  - options: Refer help messages using “-?” option
  - example_file: Input file for training examples.
    - Format for classification mode
      - <Target> <Feature1>:<Value1> <F2>:<V2>…<Fn>:<Vn>
        - Target: +1 | -1 | 0
        - Feature: <integer>, Value: <float>
          - Feature/value pairs MUST be ordered by increasing feature number.
        - For example
          - -1 1:0.43 3:0.12 9284:0.2 --- Negative example
          - 1 1:0.1 10:0.45 --- Positive example
          - 0 1:0.34 5:0.13 189:0.5 --- Unknown example
      - model_file: Result of svm_learn is the model which is learned from the training examples.
Classification Module

• `svm_classify [options] example_file model_file output_file`
  • options: Refer help messages using “-?” option
  • example_file: Test examples in the same format as the training examples.
  • model_file: The model_file from svm_learn.
  • output_file
    • The result of svm_classify which has the predicted values.
    • The predicted values are result of the decision function for each examples.
    • The sign of the predicted value is the predicted class.
    • The zero indicates unknown
The task is to learn which Reuters articles are about "corporate acquisitions".
9947 features: Each feature corresponds to a word stem.
- train.dat: 1000 positive and 1000 negative examples
- test.dat: 600 test examples
- words: A set of word stems. Features correspond to the line numbers. (9947 lines)
CRF++

- CRF++-0.58.tar.gz -- Source
- CRF++-0.58.zip
  - Binary for MS-Windows
CRF 통합 가능한 언어

- C++, Java, Python, Perl, Ruby 등

<table>
<thead>
<tr>
<th>언어</th>
<th>설치 Directory</th>
<th>설명</th>
<th>비고</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>CRF++-0.58/sdk</td>
<td>C++에서 CRF++라이브러리 연동 방법 제공</td>
<td></td>
</tr>
<tr>
<td>JAVA</td>
<td>CRF++-0.58/java</td>
<td>JAVA에서 CRF++라이브러리 연동 방법 제공</td>
<td>swig를 이용한스크립트언어 C++ 라이브러리 인터페이스</td>
</tr>
<tr>
<td>Python</td>
<td>CRF++-0.58/python</td>
<td>Python에서 CRF++라이브러리 연동 방법 제공</td>
<td></td>
</tr>
<tr>
<td>Perl</td>
<td>CRF++-0.58/perl</td>
<td>Perl에서 CRF++라이브러리 연동 방법 제공</td>
<td></td>
</tr>
<tr>
<td>Ruby</td>
<td>CRF++-0.58/ruby</td>
<td>Ruby에서 CRF++라이브러리 연동 방법 제공</td>
<td></td>
</tr>
</tbody>
</table>
CRF++-0.58/example/basenp/

[taeseok@localhost CRF++-0.58]$ cd example/basenp/
exec.sh template test.data train.data
[taeseok@localhost python]$ ../../crf_learn -c 10.0 template train.data model
...
iter=33 terr=0.00000 serr=0.00000 act=32970 obj=19.70277 diff=0.00019
iter=34 terr=0.00000 serr=0.00000 act=32970 obj=19.70237 diff=0.00002
iter=35 terr=0.00000 serr=0.00000 act=32970 obj=19.70003 diff=0.00012
iter=36 terr=0.00000 serr=0.00000 act=32970 obj=19.69958 diff=0.00002
iter=37 terr=0.00000 serr=0.00000 act=32970 obj=19.69887 diff=0.00004
iter=38 terr=0.00000 serr=0.00000 act=32970 obj=19.69855 diff=0.00002
Done!0.15 s

[taeseok@localhost python]$ ../../crf_test -m model test.data > output.txt
...
of IN O O
Columbus NNP B B
, . O O
Ohio NNP B B
, . O O
grew VBD O O
3.8 CD B B
% NN I I
. . O O

[taeseok@localhost python]$ ./conlleval.pl -d "\t" < output.txt
processed 19172 tokens with 5051 phrases; found: 4978 phrases; correct: 4285.
accuracy: 93.67%; precision: 86.08%; recall: 84.83%; FB1: 85.45
: precision: 86.08%; recall: 84.83%; FB1: 85.45 4978
: precision: 86.08%; recall: 84.83%; FB1: 85.45 4978

# Unigram
U00:%x[-2,0]
U01:%x[-1,0]
U02:%x[0,0]
U03:%x[1,0]
U04:%x[2,0]
U05:%x[-1,0]/%x[0,0]
U06:%x[0,0]/%x[1,0]
U10:%x[-2,1]
U11:%x[-1,1]
U12:%x[0,1]
U13:%x[1,1]
U14:%x[2,1]
U15:%x[-2,1]/%x[-1,1]
U16:%x[-1,1]/%x[0,1]
U17:%x[0,1]/%x[1,1]
U18:%x[1,1]/%x[2,1]
U20:%x[-2,1]/%x[-1,1]/%x[0,1]
U21:%x[-1,1]/%x[0,1]/%x[1,1]
U22:%x[0,1]/%x[1,1]/%x[2,1]
U23:%x[0,1]

# Bigram
B

AI, ML, NN, and Deep Learning

• **AI**
  - 지식표현, game theory
  - NLP, Q&A, M.T., pattern recognition, expert system, etc

• **Machine Learning**
  - Decision tree, **Neural Net**, SVM, Naïve Bayes, Ada boost

• **Deep Learning** (Deep Neural Network)
  - Convolutional Neural Network (CNN)
  - Recurrent Neural Network (RNN)
  - Restricted Boltzmann Machine (RBM)
SMT and NMT
Example-based MT

• Basic idea: translate a sentence by using the closest match in parallel data.
• First proposed by Nagao (1981)
• Ex:
  • Training data:
    • w1 w2 w3 w4 \(\Rightarrow\) w1’ w2’ w3’ w4’
    • w5 w6 w7 \(\Rightarrow\) w5’ w6’ w7’
    • w8 w9 \(\Rightarrow\) w8’ w9’
  • Test sent:
    • w1 w2 w6 w7 w9 \(\Rightarrow\) w1’ w2’ w6’ w7’ w9’
• Types of EBMT:
  • Lexical (shallow)
  • Morphological / POS analysis
  • Parse-tree based (deep)

• Types of data required by EBMT systems:
  • Parallel text
  • Bilingual dictionary
  • Thesaurus for computing semantic similarity
  • Syntactic parser, dependency parser, etc.
• Word alignment: using dictionary and heuristics
  ➔ exact match

• Generalization:
  • Clusters: dates, numbers, colors, shapes, etc.
  • Clusters can be built by hand or learned automatically.

• Ex:
  • Exact match: 12 players met in Paris last Tuesday ➔
    12 Spieler trafen sich letzten Dienstag in Paris
  • Templates: $num players met in $city $time ➔
    $num Spieler trafen sich $time in $city
Progress in M.T.

A brief history of MT...

- Rule Based
- Statistical
- Neural

MT Quality


Statistical MT

• Basic idea: learn all the parameters from parallel data
• Major types: Word-based, Phrase-based

• Strengths:
  • Easy to build, and it requires no human knowledge
  • Good performance when a large amount of training data is available

• Weaknesses:
  • How to express linguistic generalization?
Hybrid MT

• Basic idea: combine different approaches

• Types of hybrid HT:
  • Borrowing concepts/methods:
    • SMT from EBMT: phrase-based SMT; Alignment templates
    • EBMT from SMT: automatically learned translation lexicon
    • Transfer-based from SMT: automatically learned translation lexicon, transfer rules; using LM
  • Using two MTs in a pipeline:
    • Using transfer-based MT as a preprocessor of SMT
  • Using multiple MTs in parallel, then adding a re-ranker
Statistical M.T. with Bilingual(Parallel) Corpus
SMT Model

\[ e_{best} = \arg\max_e P(e|f) = \arg\max_e P(f|e) P(e) \]

- Decoding Algorithm
- Translation Model
- Language Model
Neural Machine Translation

• Demo -- http://104.131.78.120/

History Google Translator

• 2006, SYSTRAN

• 2007, SMT

• 2016, Google’s Multilingual Neural M.T.
Traditional vs. Google Translate

• **Traditional M.T. system**
  • Break sentences into words and phrases
  • Translate each individually

• **Google Translate, 2016/09**
  • Neural translation system
  • Neural network to work on entire sentences at once
  • Multiple language combinations
    • Eng <-> Japanese & Eng <-> Korean  →  Kor <-> Japanese
    • By Cho Kyunghyun, New York Univ.
Learning the lingo: Google Translate

• gathers from across the internet
• community input
• the Bible for obscure languages
Imprisoned American Student, 22, Sent Home From North Korea Amid Reports He's in a Coma
An American student who hasn't been seen since North Korea convicted him of crimes against the country is reportedly on his way back home, but he's believed to be in a troubled condition.

Otto Warmbier made international headlines in 2016 after North Korea claimed he attempted to steal a propaganda poster led to a sentence of 15 years hard labor. The 22-year-old hadn't been seen by American representatives in Pyongyang since.

Watch: Friends Shocked After 'Well-Liked' American Student is Arrested in North Korea for Alleged 'Hostile Act' on Tuesday, but the already strange tale took some even stranger turns when it was reported that the Cincinnati native is on his way back to Ohio in a coma, one his parents have been told Warmbier has been in for nearly his entire incarceration.

Pyongyang officials have reportedly claimed Warmbier contracted botulism shortly after the conclusion of the hours-long trial in which he was found guilty of "hostile acts against the state."
The Warmbiers are told he was subsequently "given a sleeping pill, from which he never woke up," they told The Washington Post.

Swedish diplomats who represent American interests in the Hermit Kingdom, with which the U.S. has no ties, say they haven't been given access to Warmbier since the trial and there is no way to know whether North Korea's account is true.

Warmbier is being transported back via Japan, where State Department officials were slated to meet him for the journey back to Ohio.
In what an official has called a "bizarre coincidence," that journey is happening the same day that retired NBA star Dennis Rodman began his fifth-high-profile visit to Pyongyang.

Weird still. The Washington Post reports that the official believes the Basketball Hall of Fame may be visiting as part of an attempt by Pyongyang to distract from Warmbier's condition.

Either way, more likely be known about Warmbier's physical state once he's back on American soil Tuesday evening.

Watch: Dennis Rodman Charged With Hit-and-Run After Allegedly Driving on Wrong Side of Freeway.
빅데이터 활용 예: 구글 번역

- 기존의 기계 번역 방식
  - 변환(transfer) 방식과 피봇(pivot) 방식의 자동 번역 기법
  - 컴퓨터가 명사, 형용사, 동사 등 단어와 어문의 문법적 구조를 인식하여 번역하는 방식

- 구글이 제공하는 자동 번역 서비스인 구글 번역의 특징
  - 통계적 방식: 빅데이터를 활용하는 방법으로 구현
  - 수억 건의 문장과 번역문을 데이터베이스화
  - 번역시 유사한 문장과 어구를 기존에 촉적된 데이터를 바탕으로 추론
  - 구글은 수억 건의 자료를 활용하여 전 세계 58개 언어 간의 자동 번역 프로그램 개발에 성공

- 데이터 양의 측면에서의 엄청난 차이가 자동 번역 프로그램의 번역의 질과 정확도에 영향을 미침
GNMT: Google’s Multilingual Neural Machine Translation System

• Zero-Shot Translation
Zero-Shot Translation

The stratosphere extends from about 10km to about 50km in altitude.
Part (a) shows an overall geometry of these translations.

• The points in this view are colored by the meaning; a sentence translated from **English to Korean with the same meaning** as a sentence translated from **Japanese to English share the same color**.

• From this view we can see distinct groupings of points, each with their own color.

Part (b) zooms in to one of the groups.

Part (c) colors by the source language.

• Within a single group, we see a sentence with the same meaning but from three different languages.

• This means the network must be encoding something about the semantics of the sentence rather than simply memorizing phrase-to-phrase translations.

• We interpret this as a sign of existence of an interlingua in the network.
Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, yonghui, schuster, zhifeng, qvl, mnorouzi@gmail.com
Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingener, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hitode Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, Jeffrey Dean

Neural Machine Translation (NMT) is an end-to-end learning approach for automated translation, with the potential to overcome many of the weaknesses of conventional phrase-based translation systems. Unfortunately, NMT systems are known to be computationally expensive both in training and in translation inference – sometimes prohibitively so in the case of very large data sets and large models. Several authors have also charged that NMT systems lack robustness, particularly when input sentences contain rare words. These issues have hindered NMT’s use in practical deployments and services, where both accuracy and speed are essential. In this work, we present GNMT, Google’s Neural Machine Translation system, which addresses many of these issues. Our model consists of a deep LSTM network with 8 encoder and 8 decoder layers using residual connections as well as attention connections from the decoder network to the encoder. To improve parallelism and therefore decrease training time, our attention mechanism connects the bottom layer of the decoder to the layer of the encoder. To accelerate the final translation speed, we employ low-precision arithmetic during inference computations. To improve handling of rare words, we divide words into a limited set of common sub-word units (“wordpieces”) for both input and output. This method provides a good balance between the flexibility of “character”-delimited models and the efficiency of “word”-delimited models, naturally handles translation of rare words, and ultimately improves the overall accuracy of the system. Our beam search technique employs a length-normalization procedure and uses a coverage penalty, which encourages generation of an output sentence that is most likely to cover all the words in the source sentence. To directly optimize the translation BLEU scores, we consider refining the models by using reinforcement learning, but we found that the improvement in the BLEU scores did not reflect in the human evaluation. On the WMT’14 English-to-French and English-to-German benchmarks, GNMT achieves competitive results to state-of-the-art. Using a human side-by-side evaluation on a set of isolated simple sentences, it reduces translation errors by an average of 60% compared to Google’s phrase-based production system.
References

- Austermühl, Frank (2001) Electronic Tools For Translators
- **Free-to-use machine translation on the web:**
  - [http://www.translatorsbase.com/](http://www.translatorsbase.com/) (Free human translation service)
  - [http://www.systranet.com/](http://www.systranet.com/) (the Systran site)
  - [http://www.reverso.net/textonly/default_ie.asp](http://www.reverso.net/textonly/default_ie.asp)
  - [http://translate.google.com/](http://translate.google.com/)
마지막으로...
• 한국어 형태소 분석
• 구문 분석
• 색인어 추출 및 가중치 계산

• 복합명사 분해
• 맞춤법 검사 및 교정

• 자동 문서 분류
• 자동 띄어쓰기 등

http://nlp.kookmin.ac.kr/
http://cafe.daum.net/nlpk

Demo Site: Korean Language Processing System

Natural Language Processing Laboratory, Kookmin University

• Korean Morphological Analyzer(형태소 분석)
• Term Extraction for Information Retrieval(색인어 추출)
• Hangul Speller and Spell Correction(맞춤법 검사/교정)
• Decomposition of Compound Nouns(복합명사 분해)
• Automatic Segmentation of Korean Sentence(자동 띄어쓰기)
• Automatic K-E Conversion(한영 자동전환)
• Normalization of Arabic/Numerical Words(수식 어절 정규화)
• Korean Dependency Parser(한국어 구문분석)
• 고비도 어절 조합(18만 어절)의 적용을 테모
다
• 한글 어절의 출현 환율
• 한글 음절의 출현 확률
• 자동 문서 분류
형태소 분석과 구문분석

한국어 구문 분석 시스템

입력: 문장은 입력한 후에 실행버튼을 누르세요.
여기에 한국어 문장을 입력한 후에 실행버튼을 누르세요.

입력

P.I.
F: 누락된 V 누락 된 V
O: 입력된 값 입력된 값 J
T: 입력된 값 T
K: 입력된 V 입력된 V
L: 입력된 값 입력된 값 J
N: 입력된 값 입력된 값 J
문서에서 키워드 추출

한글 자동 띄어쓰기 시스템

입력: 여기에 한글 문장들을 붙여서 입력한 후에 실행 버튼을 눌러보세요.

실행

출력: 여기에 한글 문장들을 모두 붙여서 입력하고 실행 버튼을 눌러 보세요. 실행 버튼을 누르면 자동으로 띄어쓰기를 하여 공백을 삽입하고 그 결과를 아래 출력 부분에 보여줍니다.
감사합니다!

sskang@kookmin.ac.kr