Linguistic, computational and operational architectures of MT systems (Part 1)

- Fundamental facts about HT & MT
  - Nature of MT
    - scientific technology, not a science
    - variety of goals & associated evaluation methods
    - misconceptions about MT
  - Independence of linguistic and computational architectures of MT systems
    - Linguistic: objects (representations)
    - Computational: processes (RBMT - SMT - EBMT - hybrid...)
    - Types & size of resources w.r.t. MT architectures
  - Operational architectures (time permitting)
  - Current evolution, recap & first conclusions

I. Fundamental facts about MT

- Varieties, difficulties, evaluation in HT (Human Translation)
  - Translation is multiple!
  - Translation is difficult!
  - Evaluation of HT is also multiple and difficult!
- The place of MT in a translation process
  - More or less automation: MT#1 > [MT#2, HT >...] typology
    - Human Translation workflow
    - Automated Translation workflow
  - What can be automated outside of the "MT module"
  - What can be done by humans to help the MT module
- The CxAxQ theorem
  - and ways of measuring Coverage, Automaticity, & Quality
- Dream & reality: misconceptions & achievements
Human Translation is multiple!

- Translating & interpreting are TRADES
  speak (P1, (L1, L2)) \(\iff\) able (P1, translate (L1, L2))
- Variety of competences & tasks

Evaluation of HT is multiple and difficult

- Grading by professors, no MCQ

Human Translation is difficult!

- Sources of difficulties
  - underspecification
    L_S vs. L_T
    ex: countability,
    determination
  - different views
    of the world
  - ambiguity,
    fuzziness,
    (probable)
    unaxiomatizibility

What can be automated?

- The term MT is ambiguous
  In French: MT\#1 = TAO, MT\#2 = TA
  - MT\#1 — Automated Translation
  - MT\#2 — Automatic Translation
  - FAMT — Fully Automatic
  - HAMT — Human Aided Translation
  - DBMT — Dialogue-Based MT
  - IMT — Interactive Machine Translation
  - MAHT — Machine-Aided
  - FHT — Fully Human
Professional Human Translation Workflow

• Only 50% time spent on "translation proper"
• Documents parts, translation units, segments

Automated Translation Workflow

• MT is only one module
• Human interaction possible, but translation proper is done by the system

Automation outside of MT module

• Writing control
  • language appropriateness
    1. lexicon (unknowns -L_S+L_T-L_S)
    2. grammar (anaphor, ellipsis, extraposition, coordination, comb)
  • complexity
    1. simplification (split long sentences, use simple constructions)
    2. de-underspecification (fill ellipsis, 0-pronouns)

• Pre-edition
  • mark segments, special chunks, groups (attachment)
  • mark POS, Word sense saw_V#eye

• Preparation
  • Complete/revise segmentation

• Use of results
  Modify choices
  • rescoring, filters, tournament

During post-edition
  • adapt to user choices
  • integrate new equivalents & recompute translation

Human help to MT module

• Typical pipeline process
  IRs - intermediate representations (data) and phases (processes)
  R0 = input → P1 → R1 → P2 → R2 → Pk → Rk → Pf → Rf = output

• Steps = 1 or more phases (Pi)
  |← Analysis |← Transfer |← Generation |←|
  |← Analysis |← Transfer+Generation |←|
  |← Enconversion |← Deconversion |←|
  (2 or 3 lexical spaces)

• Interaction possible
  during the phases ITS-1, TAO, TransActive, ITS-2
  on multiple/factorizing IRs KBMT/KANT, JETS, Systran-5...
The CxAxQ MT theorem

- Coverage x Automaticity x Quality << 100%
- Quality here means linguistic quality (judged by translators)
- That limitation is in the nature of the problem

⇒ FAHQMT for all domains is impossible

- But 2 factors can be ≈100% if compromising on 3rd
  - C x A ≈100% web translation
  - A x Q ≈100% METEO, ALTFLASH
  - C x Q ≈100% DBMT (JETS, CATALYST, LIDIA)

What are C, A, Q & how to measure them?

- Automaticity (MT module only)
  \[ A = 1 - \frac{T(\text{human\_interaction})}{T(\text{human\_first\_draft})} \]
- Quality (wrt HT)
  \[ Q = (100 - 2 \times T(\text{post\_edition\_MT}))\% \]

Coverage computation

Estimate coefficients, then use normalized sums

<table>
<thead>
<tr>
<th></th>
<th>FL frequency + importance</th>
<th>GM general meanings ((\times))</th>
<th>SM specific meanings ((\times))</th>
</tr>
</thead>
<tbody>
<tr>
<td>W, words &amp; compounds</td>
<td>FI_W_i</td>
<td>GMW_i_1_1...</td>
<td>SMW_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMW_i_g</td>
<td>SMW_i_s</td>
</tr>
<tr>
<td>T, terms</td>
<td>FI_T_i</td>
<td>GMT_i_1_1...</td>
<td>SMT_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMT_i_g</td>
<td>SMT_i_s</td>
</tr>
<tr>
<td>C, collocations</td>
<td>FI_C_i</td>
<td>GMC_i_1_1...</td>
<td>SMC_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMC_i_g</td>
<td>SMC_i_s</td>
</tr>
<tr>
<td>S, standard constructions</td>
<td>FI_S_i</td>
<td>GMS_i_1_1...</td>
<td>SMS_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMS_i_g</td>
<td>SMS_i_s</td>
</tr>
<tr>
<td>P, particular constructions</td>
<td>FI_P_i</td>
<td>GMP_i_1_1...</td>
<td>SMS_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMP_i_g</td>
<td>SMS_i_s</td>
</tr>
</tbody>
</table>

Coverage computation

Dream & reality: misconceptions & achievements

- The dream of FAHQMT with large coverage (from 1951!)
- Fully Automatic High Quality Machine Translation
- Proof of impossibility by Bar-Hillel (1960)
- ... impossibility of an angelic goal, of course
- Concrete operations, useful systems (since 1967!)
- 1972, Systran, Ispra, Euratom: 20% ling. quality, 80% usefulness
- FAHQMT on sublanguages does exist
- CETA 1965-70, METEO 1977—, GETA-Ariane 78—, METAL, ENGSPLAN/SPANAM (PAHOMTS), ALTFLASH... (many)
- You don’t see them because they are tailored!
- Rough MT exists (web translators...)
- and SMT has introduced Machine Learning in MT enlarging the Coverage (and # languages), but not the Quality

Estimate coefficients, then use normalized sums

<table>
<thead>
<tr>
<th></th>
<th>FL frequency + importance</th>
<th>GM general meanings ((\times))</th>
<th>SM specific meanings ((\times))</th>
</tr>
</thead>
<tbody>
<tr>
<td>W, words &amp; compounds</td>
<td>FI_W_i</td>
<td>GMW_i_1_1...</td>
<td>SMW_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMW_i_g</td>
<td>SMW_i_s</td>
</tr>
<tr>
<td>T, terms</td>
<td>FI_T_i</td>
<td>GMT_i_1_1...</td>
<td>SMT_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMT_i_g</td>
<td>SMT_i_s</td>
</tr>
<tr>
<td>C, collocations</td>
<td>FI_C_i</td>
<td>GMC_i_1_1...</td>
<td>SMC_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMC_i_g</td>
<td>SMC_i_s</td>
</tr>
<tr>
<td>S, standard constructions</td>
<td>FI_S_i</td>
<td>GMS_i_1_1...</td>
<td>SMS_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMS_i_g</td>
<td>SMS_i_s</td>
</tr>
<tr>
<td>P, particular constructions</td>
<td>FI_P_i</td>
<td>GMP_i_1_1...</td>
<td>SMS_i_1_1...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GMP_i_g</td>
<td>SMS_i_s</td>
</tr>
</tbody>
</table>
Next...

- The various tasks of MT and their difficulty

- Independence of linguistic and computational architectures of MT systems
  - Linguistic: objects (representations)
  - Computational: processes (RBMT - SMT - EBMT - hybrid...)
  - Types & size of resources w.r.t. MT architectures

- Example of a heterogeneous, IL-based system: UNL
  - Introduction to UNL hypergraphs
  - An experiment to evaluate UNL real potential
  - Current developments & perspectives

Main goals of MT & associated measures

- 4 main translational situations, in increasing order of difficulty of automation:
  1. production of high-quality (HQ) translation by bilinguals
     dissemination by bilinguals $\rightarrow$ human time needed
  2. understanding text or speech in an unknown language
     assimilation $\rightarrow$ adequacy / # buying acts (e-commerce)
  3. production of HQ translation from an unknown language
     HQ assimilation by monolinguals $\rightarrow$ human time needed
  4. production of HQ translation into unknown languages
     HQ dissemination by monolinguals $\rightarrow$ human time needed

Misconceptions about MT

- "Pivot MT" $\rightarrow$ "Rule-Based MT"
- "Transfer MT" + N languages $\rightarrow$ N(N-1) transfers
- "Rule-Based MT" $\rightarrow$ high cost
- "Statistical MT" $\rightarrow$ low cost
- "Statistical MT" $\rightarrow$ majority of operational systems
- "Statistical MT" $\leftrightarrow$ "Rule-Based MT"
- BLEU measures the quality of translations
- Linguistic quality of MT outputs has increased with SMT
- Adequacy should be measured by a positive number
- "MT with interlingual pivot (IL) cannot work and scale up": But,
  - ATLAS-II (Fujitsu) has been the best for J$\rightarrow$E since 20 years
  - It has more than 7M dictionary entries (v.14, Dec. 2008)

To sum up about MT, MT systems, variety

- MT is not a science, but a scientific technology
  - deep and hard problems come from the confrontation with reality
- MT without concrete goals is not MT
  - $\rightarrow$ automating various translation tasks for various users
  - USSR in the 70's: MT without machines and without translations!
- There is no such thing as ONE unique goal / task for MT
  - analogy with transportation
    - translational situations
      - web surfing
      - diffusion by bilinguals (general)
      - technical translation
      - restricted translation (ALTFLASH)
    - transport situations
      - bicycle
      - car
      - train
      - automatic train
    - (difficulties are not parallel!)
- The SAME system can be judged very good/bad
  - if used when a user wants (a help) to understand a content
  - if "pushed" on the user by the originator of a content
3 types of MT architectures

- Linguistic
- Computational
- Operational

Linguistic vs. computational architectures

**Linguistic architecture**
- Objects: see Vauquois' triangle
- Intermediate representations:
  - Direct, semi-direct, transfer (≥ 7 variants)
  - 2 lexical spaces
  - IL (≥ 2 variants)
  - 3 lexical spaces

**Computational architecture**
- Automatic processes
- Human interaction, if any
- Programming paradigms:
  - Direct programming
  - RBMT (rules, automata...)
  - Corpus-based:
    - SMT, PSMT (unsupervised)
    - EBMT (≥ 3 variants)
    - ± supervised
    - Hybrid

Linguistic architectures in MT: Vauquois' triangle

- Deep understanding level
- Interlingual level
- Logico-semantic level
- Mixing levels
- Syntactico-functional level
- Syntagmatic level
- Morpho-syntactic level
- Graphemic level

- Conceptual transfer
- Semantic transfer
- SPA-structures (semantic & predicate-argument)
- Multilevel transfer
- Multilevel description
- Syntactic transfer (deep)
- F-structures (functional)
- C-structures (constituent)
- Semi-direct translation
- Descending transfers
- Tagged text
- Direct translation
- Text

Computational architectures

- Intermediate Representation i
- Intermediate Representation i+1
- Computational type of the phase
- Expert
- Rule-based
- Statistical (probabilistic)
- Example-based
- EBMT
- Well-formedness grammar rules
- Translations of transduction automata
- Rewriting rules
- Transformational grammars
- Annotated parallel corpora
- S-SSTC
- Raw parallel corpora
- Analyse-based MT
- ABMT
Operational architectures for various translational situations

- **Users / tasks**
  - help bilinguals to produce good translations
  - help people understand an unknown or little known language
  - help people to communicate (chat, spoken translation...)

- **Language pairs / volumes / kinds**
  - 1→1 (ALT/JE)
  - 1→N (MedSLT), N→1
  - 1→1 (Converser for Healthcare)
  - 1→N (US Army)
  - N→N (debates, chat, multilingual peace forces)

- **Possible involvement of humans**
  - authors (controlled language, rewriting, interactive disambiguation)
  - professional/occasional translators (post-editing)
  - readers (guessing from multiple factorized output)

- **Available resources**
  - data: huge parallel corpora necessary for SMT
  - humans: computational linguists, lexicographers needed for "expert" MT

LT architectures on which we work

---

Direct translation systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Steps</th>
<th>Method</th>
<th>Comments</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT 1975</td>
<td>segmentation word for word translation</td>
<td>FST (rules + dict.) rules</td>
<td>OK for very near languages Japanese → Korean Hindi → Urdu</td>
<td>ATLAS-I Fujitsu, 1976-78</td>
</tr>
<tr>
<td>SMT 1980</td>
<td>segmentation, reordering...</td>
<td>alignment + &quot;decoding&quot; statistical</td>
<td>SMT = first idea about MT from war cryptographers (W. Weaver 1949)</td>
<td>Many SMT IBM... 1980-</td>
</tr>
<tr>
<td>EBM 2000</td>
<td>no preprocessing &quot;pure&quot; EBM</td>
<td>analogy resolution + n-gram filtering analogical</td>
<td>Results = those of SMT Nagao 1984 (similarity MT) Lepage 2000 (real analogy)</td>
<td>ALEPH ATR 2000-</td>
</tr>
</tbody>
</table>
**Conceptual transfer systems** (IL with separate lexicon)

<table>
<thead>
<tr>
<th>Interlingual level</th>
<th>Conceptual transfer</th>
<th>Semantico-linguistic interlingua</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
<td><strong>Enconversion</strong></td>
<td><strong>Conc. transfer</strong></td>
</tr>
<tr>
<td>RBMT 1980—</td>
<td>Lemmatization</td>
<td>in principle none</td>
</tr>
<tr>
<td></td>
<td>direct or rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td>string-graph</td>
<td></td>
</tr>
<tr>
<td></td>
<td>transformations</td>
<td></td>
</tr>
<tr>
<td>RBMT 1980—</td>
<td>DCG (?) rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBMT 1997—</td>
<td>depending on partners rules</td>
<td>depending on partners rules</td>
</tr>
</tbody>
</table>

**Graphemic level**

**Text**

---

**Size & cost of resources / MT architectures**

<table>
<thead>
<tr>
<th>Type</th>
<th>Sentences</th>
<th>6.5 word/sentence</th>
<th>25 word/sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMT</td>
<td>0.9—3 Mw</td>
<td>BTEC, Meteo</td>
<td>50—200 Mw</td>
</tr>
<tr>
<td>PSMT</td>
<td>3.6—12 K pages</td>
<td></td>
<td>200—800 K pages</td>
</tr>
<tr>
<td>analogical EBMT</td>
<td>0.15—0.5 M</td>
<td></td>
<td>2—8 M sentences</td>
</tr>
<tr>
<td></td>
<td>2.4—8 m day</td>
<td></td>
<td>100—400 m day</td>
</tr>
<tr>
<td></td>
<td>(already done!)</td>
<td></td>
<td>(available?)</td>
</tr>
<tr>
<td>EBMT with trees</td>
<td>N/A for short sent.</td>
<td>Supervised learning</td>
<td>4—12.5 Mw</td>
</tr>
<tr>
<td>MST</td>
<td>1h/page?</td>
<td></td>
<td>15—50 K pages</td>
</tr>
<tr>
<td>Master-1</td>
<td></td>
<td></td>
<td>0.15—0.5 M sentences</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10—40 m day</td>
</tr>
<tr>
<td>EBMT with trees and S-SSTCs</td>
<td>N/A for short sent.</td>
<td>Supervised learning</td>
<td>4—12.5 Mw</td>
</tr>
<tr>
<td>SiSTeC-ebmt (USM)</td>
<td>15 h/page !</td>
<td></td>
<td>0.6—1 K pages</td>
</tr>
<tr>
<td></td>
<td>dict. (50 K)</td>
<td></td>
<td>0.006—0.01 M sentences</td>
</tr>
<tr>
<td></td>
<td>available</td>
<td></td>
<td>6—10 m day</td>
</tr>
<tr>
<td>RBMT</td>
<td>DICT. 3-10 K</td>
<td>0.6—2 m day</td>
<td>15—150 m day</td>
</tr>
<tr>
<td></td>
<td>Total 1—3 m y</td>
<td>(to do!)</td>
<td>Total 40—175 m y</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Knowledge-based systems: explicit understanding, IL linked with an ontology**

<table>
<thead>
<tr>
<th>Deep understanding level</th>
<th>Ontological interlingua</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
<td><strong>Enconversion</strong></td>
</tr>
<tr>
<td>KBMT 1980—</td>
<td>lemmatization &amp;</td>
</tr>
<tr>
<td></td>
<td>EPSG+F-structures</td>
</tr>
<tr>
<td></td>
<td>+pseudo-unification</td>
</tr>
<tr>
<td></td>
<td>rules (using UP)</td>
</tr>
<tr>
<td>RBMT 1997—</td>
<td>dictionary + FST</td>
</tr>
<tr>
<td></td>
<td>rules</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>SMT 2003—</td>
<td>learned from</td>
</tr>
<tr>
<td></td>
<td>(string,IF) KB</td>
</tr>
<tr>
<td></td>
<td>statistical</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Operational architectures**

- **Translational situations**
  - *Graph of needed translations*, with weights (Flows, Importance)
  - Possibility/necessity of human help to MT
  - Possibility/necessity of pre-edition or post-edition
  - Time constraints
  - Available resources (no SMT if no corpora, no EMT if no experts)

- **They may influence the linguistic & computational archit.**
  - Reusing existing steps to build new translation pairs
  - Ex: transfer approach (if N languages, #transfers still linear)
Current evolution

- Deeper linguistic architectures
- Hybrid computational architectures,
  - Coupled with user involvement
- Use more Machine Learning
  - Use it also with deep transfer or pivot architectures (through an IL)
    - (More an that in lecture 3)

Recap & first conclusions

- The distinctions RBMT, EBMT, an-EBMT, (P)SMT... concern the computational architecture only (processes)
- The rawer the corpora, the larger they must be
  - SMT/PSMT is for niches for the rich (languages, texts)
    - Few parallel corpora of 200—800 K pages
    - To build them from scratch is 2 to 3 times more expensive than to build a classical large RBMT system
- IL-based MT can use any computational framework
  - Statistical, analogical, rule-based, hybrid
    - All depends on available corporal / linguistic / human resources
- Many applications need an adequate IL
  - All applications needing to manipulate content in a strongly multilingual setting

Introductive words

- No time to show details of various linguistic architectures in part 1 (P1)
  - But we went into new territory and detailed usually implicit notions
    - Workflow in HT and MT (MT#1 or automated MT)
    - How to measure C, A, Q (ling. quality) for the "C.A.Q MT theorem"
- The goals of part 2 will be to:
  - Make explicit the basis of linguistic architectures in MT
  - Justify independence of linguistic and computational architectures
  - At the same time, present various intermediate structures
  - Speak about their pros and cons.
Outline of part 2

- Kinds of representations *usable in principle*
  - Linguistic level: monolevel and multilevel structures
  - Scope (what are *units of translation*?)
    - segments, infrasegments, supersegments, whole documents?
  - Geometry: strings, trees, charts/lattices, (hyper)graphs, log. forms
  - Abstractness: type of <string, structure> correspondence
- Various kinds of representations *really used in MT*
  using existing systems as examples
  - morphosyntactic structures
  - syntactic structures: c-structure, f-structure
  - logico-semantic structures: spa-structure
- Different sorts of *deep pivots*
  using UNL as an example
  - hybrid, semantico-linguistic, semantico-pragmatic
  - UNL (semantico Linguistic)
- Recap and conclusions
  - Pros and cons of various linguistic architectures w.r.t. translational situations

---

Formalized representations of texts

<table>
<thead>
<tr>
<th>Linguistic levels</th>
<th>Main linguistic organisation</th>
<th>Geometrical structure</th>
<th>Algebraic structure</th>
<th>Text -- Structure Correspondence</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surface Deep</td>
<td>Syntagms (constituents)</td>
<td>String / Graph of strings (chart)</td>
<td>Label / Structured label</td>
<td>Sentence (= all) Page (Ariane-G5, Sygmart) Document</td>
<td></td>
</tr>
<tr>
<td>1-stratal monolevel</td>
<td>Dependencies</td>
<td>Tree</td>
<td>Boolean features / Structured attributes (vectors)</td>
<td>totally abstract (ex. UNL graph)</td>
<td></td>
</tr>
<tr>
<td>n-stratal multilevel</td>
<td>Logical and semantic relations</td>
<td>Hypergraph</td>
<td>Feature structures (+ typed)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Kinds of representations *usable in principle*

- **Linguistic level**
  - levels
    - monolevel and multilevel structures
- **Scope (what are *units of translation*?)**
  - segments, infrasegments, supersegments, whole documents?
- **Geometry**
  - strings, trees (many), charts/lattices (SMT: Pharaoh, Moses, Josuah) (hyper)graphs (UNL), logical forms (Microsoft Research)
- **Abstractness**
  - type of <string, structure> correspondence
    - concrete <-> abstract ≠ surface <-> deep

---

Linguistic architectures and intermediate representations

- Rule-based MT (symbolic) for sub-languages
- Translation Memory based MAHT
- All-domain MT via UNL
- Ontological interlingua
- Semantico-linguistic interlingua
- SPA-structures (semantic & predicate-argument)
- Multilevel description
- Multistructural transfer
- Syntactic transfer (deep)
- Syntactic transfer (surface)
- F-structures (function)
- C-struct
- Analogy-based MT
- Example-based MT (S-SSTC)
Structures in direct translation systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Steps</th>
<th>Method</th>
<th>Comments</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT</td>
<td>segmentation word for word translation</td>
<td>FST (rules + dict.) rules</td>
<td>OK for very near languages</td>
<td>ATLAS-I Fujitsu, 76-78</td>
</tr>
<tr>
<td>SMT</td>
<td>segmentation, reordering...</td>
<td>alignment + &quot;decoding&quot; statistical</td>
<td>SMT = first idea about MT from war cryptographers (W. Weaver 1949)</td>
<td>Many SMT IBM... 1980-</td>
</tr>
<tr>
<td>EBMT</td>
<td>no preprocessing &quot;pure&quot; EBMT</td>
<td>analogy resolution + n-gram filtering analogical</td>
<td>Results = those of SMT Nagao 1984 (similarity MT) Lepage 2000 (real analogy)</td>
<td>ALEPH ATR 2000-</td>
</tr>
</tbody>
</table>

Graphemic level → Direct translation → Text

Structures in semi-direct translation systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1G-MT 1950</td>
<td>Program-based segmentation &amp; lemmatization procedural</td>
<td>dictionary consult. + reordering &quot;macros&quot; procedural</td>
<td>tables + string macros procedural</td>
<td>GAT (Georgetown) Isra, 1965-69 SPANAM-1, PAHO, 1975? LMT, English, French...</td>
</tr>
<tr>
<td>SMT 1990</td>
<td>segmentation &amp; lemmatization statistical</td>
<td>alignment + &quot;decoding&quot; statistical</td>
<td>language model statistical</td>
<td>Candide IBM, 1980— Many SMT systems NIST, IWSLT Google (1)</td>
</tr>
</tbody>
</table>

Morpho-syntactic level → Semi-direct translation → Tagged text

Graphemic level → Text

Structures in descending surface syntactic transfer systems: RBMT (+SMT / LanguageWeaver?)

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer + syntactic generation</th>
<th>Morphological generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMT 1980</td>
<td>ECFG +decorations gram. rules</td>
<td>recursive descent procedural often in LISP</td>
<td>grammar+dict. rules</td>
<td>AS-Transac Toshiba, 1982—</td>
</tr>
<tr>
<td>RBMT 1984</td>
<td>Lemmatization + Slot-grammars</td>
<td>recursive descent logic programming</td>
<td>dictionary + tables+ prog.</td>
<td>METAL Austin, 1982— Duett-2 Sharp, 1984—</td>
</tr>
</tbody>
</table>

Syntagmatic level: C-structures (constituents)

Morpho-syntactic level: Tagged text

Graphemic level: Text
Structures

- **Syntactic trees**
  - **Information on nodes**
    - simple labels METEO (Q-systems)
    - main label + boolean attributes (TAUM-aviation)
    - main label + typed attributes (Ariane, METAL & many others)
  - **Concrete trees (projective)**
    - constituents: frontier (leaves) is a prototype of the utterance
    - dependencies: in-order (infix) traversal is a prototype...
  - **Abstract trees (not projective)**
    - normalized trees with semantic constituents (Colmerauer, METEO)
    - sometime necessary because of
      1. verbs with separable particles (German, English...)
      2. comb-like constructions
        - A, B & C gave A', B' & C' to A", B" & C"
        - \((\text{give}(A, A', \text{to}(A'')), \text{give}(B, B', \text{to}(B'')), \text{give}(C, C', \text{to}(C'')))\)

---

**Concrete syntagmatic (constituent) tree**

```
PHVB

NP
  N
  Coo
  NP
  V
  give
  cod
  .
  N
  Emil

NP
  N
  donut
  to
  Prep
  N
  Fathia
```

---

**Concrete dependency tree**

```
give V

subj
Ann N
cod
N
to
Prep
Bob

obj1

cod
N
gov
Coo
donut
N
gov

obj2

```

---

**Abstract dependency tree**

```
give V

subj
Ann N
cod
N
to
Prep
Bob

obj1

```
Examples : MT for access, web (1)

ENGLISH (human version)  FRENCH (human version)  ENGLISH (Systran FRE-ENG version)
The European-Heritage.net thesaurus covers the fields of archaeology and architecture as defined in the Council of Europe conventions signed in Granada (1985) and Malta (1992).
The European-Heritage.net thesaurus covers the fields of archaeology and architecture within the meaning of conventions of the Council of Europe of Grenade (1985) and Malta (1992).

It encompasses information ranging from other fields involved, categories of cultural assets and legislation, to activities, skills and finance supplemented by a number of specific thesauri composed by each member state on a particular topic, such as the thesaurus on Andalusian heritage or the architectural thesaurus from the Mémorial database in France.
Il prend en compte des aspects aussi variés que les acteurs, les catégories de biens culturels, la législation ou encore les interventions, les métiers et les financements. Il est complété et prolongé par des thesauri spécifiques développés par chaque Etat membre sur tel ou tel sujet spécifique, comme le thesaurus du patrimoine andalou ou le thesaurus d'architecture de la base de données documentaire Mémorial en France.

This new, open-ended search tool will come on line shortly, together with a management and administration system shared among the various contributors.
Cet instrument de recherche, open ended, sera mis prochainement en ligne accompagné d'un dispositif de gestion et d'administration réparti entre les différents contributeurs.

Comparison : rough vs. raw MT results

Reverso rough Spanish-English output
Message of the Chief Operating officer of the World Organization of the Health

From its discovery, the antibiotics have transformed completely the perspective of the humanity with regard to the infectious diseases. Today the use of the antibiotics, cocktail' with improvements in the reparation, the housing and the nutrition, together with the advent of the programs of widespread vaccination, have given place to a notable decrease of infectious diseases that before were common and were annihilating entire populations.

Scourges that terrified million persons, as the pest, the scourge cough, the poliomyelitis and the scarlatina, they have been controlled or are on the verge of being controlled. Now, in the dawn of a new millennium, the humanity faces another crisis. Diseases before curable as the gonorrhea and the fever typhoida they are becoming rapidly difficult to treat, whereas killer old men as the tuberculosis and the malaria are armed(“assembled”) now with the increasing impenetrable resistance the antimicrobial ones.

This phenomenon is potentially controllable. The problem is increasingly deep and complex, accelerated by the abuse of the antibiotics in the developed countries and the paradoxical satiuration of the antimicrobial ones of quality in the countries in development due to the poverty and the resultant shortage of an attention of effective health.

Systran raw Spanish-English output
Message of the Director-General of the World Health Organization

From its discovery, antibiotics have completely transformed the perspective of humanity with respect to infectious diseases. Today the use of antibiotics, combined with improvements in nutrition, housing, and nutrition, together with the advent of the vaccination programs generalized, have caused a notable reduction of infectious diseases that previously were common and annihilated entire populations.

Scourges that terrified millions of people, as plague, whooping cough, poliomyelitis, and the scarlet fever, have been controlled or are on the verge of being controlled. Now, in the dawn of a new millennium, humanity faces another crisis. Previously curable diseases as the gonorrhea and typhoid fever are becoming rapidly difficult to treat, while old assassins tuberculosis and malaria now are armed of the increasingly impenetrable resistance to the antimicrobial drugs.

This phenomenon is potentially controllable. The problem is increasingly profound and complex, accelerated by the abuse of antibiotics in the developed countries and the paradoxical underutilization of the quality antimicrobial drugs in the developing countries due to the poverty and the scarcity resulting from an effective health care.

Examples : MT for access, web (2)

• FE is easy compared to EG and even more FG

GERMAN (Systran ENG-GER version)  GERMAN (Systran FRE-GER version)

Es berückichtige Aspekte der Bedeutung, daß die Beteiligten, die Kategorien kultureller Güter, die Gesetzgebung oder noch die Interventionen, die Berufe und die Finanzierungen. Er wird vervollständigt und wird durch ein spezifische Thesaurus entwickelt durch jeden Mitgliedstaat über das einer anderen spezifische Themen verlängert, als der Thesaurus des andalusischen historischen Kultururses oder des Thesaurus der Architektur der urkundlichen Datenbank Mémorial in Frankreich.

Dieses neue, offene Suchhilfsmittel kommt auf Zeits kur, zusammen mit einem Management- und Leitungssystem, das unter den verschiedenen Mitwirkenden geteilt wird.
Dieses notgedrungen entwicklungsfähige Forschungsinstrument wird gestellt demnächst online begleitet von einer Verwaltungs- und Verwaltungsvorrichtung, die aufgeteilt der den verschiedenen Beiträgern.

Descending deep syntactic transfer systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer + synt. generation</th>
<th>Morphological generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT</td>
<td>1985—</td>
<td>recursive</td>
<td>grammar+dict.</td>
<td>JETS</td>
</tr>
<tr>
<td></td>
<td>Legmatization</td>
<td>procedural</td>
<td>rules</td>
<td>(IBM-Jp, 1985-90)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>tables+prog.</td>
<td></td>
</tr>
<tr>
<td>1.5G-MT</td>
<td>1990—</td>
<td>Lemmatization</td>
<td>recursive</td>
<td>Systran</td>
</tr>
<tr>
<td></td>
<td></td>
<td>graph procedural</td>
<td>grammar+dict. rules</td>
<td>1990—</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>tables+prog.</td>
<td></td>
</tr>
</tbody>
</table>

F-structures (functional) often dependency structures

Graphemic level

© Ch. Boitet — MT lecture, part 2 1/12/2011

14

15

16

17
Horizontal surface syntactic transfer systems: RBMT & Phrase-Based SMT

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT 1992—</td>
<td>lemmatization +</td>
<td>tree flattening</td>
<td>tree flattening</td>
<td>TDMT (for SLT)</td>
</tr>
<tr>
<td></td>
<td>linear patterns rules</td>
<td>grammar+dct.</td>
<td>grammar+dct.</td>
<td>ATR, 1992—1998</td>
</tr>
<tr>
<td>RBMT 1995—</td>
<td>lemmatization + Slot Grammars prolog</td>
<td>rec. descent prolog</td>
<td>PT (from LMT)</td>
<td>Linguatech, 1995—</td>
</tr>
<tr>
<td>EBMT 2000—</td>
<td>Initial data: bilingual</td>
<td>Translation: A/T/T/G</td>
<td>EBMT (Banturjah)</td>
<td>UTMK, USM, 2000—</td>
</tr>
<tr>
<td></td>
<td>chunking</td>
<td>postprocessing</td>
<td>+WU, Melamed 1997, 2004</td>
<td></td>
</tr>
</tbody>
</table>

Syntactic transfer (surface) C-structures (constituent) Tagged text Text

Horizontal deep syntactic transfer systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT 1975—</td>
<td>grammar + dct.</td>
<td>tree transformations</td>
<td>tree flattening</td>
<td>ETAP-2, ETAP-3</td>
</tr>
<tr>
<td></td>
<td>dependency anal. rules</td>
<td>rules</td>
<td>grammar+dct.</td>
<td>IPPL, Moscow, 1977—</td>
</tr>
<tr>
<td>RBMT 1995—</td>
<td>lemmatization + Slot Grammars prolog</td>
<td>rec. descent prolog</td>
<td>PT (from LMT)</td>
<td>Linguatech, 1995—</td>
</tr>
<tr>
<td>RBMT+ SMT 1999—</td>
<td>MSR (Microsoft ) analyzers</td>
<td>Learned from pairs</td>
<td>Microsoft generators</td>
<td>MTS-1 (on tech. documents)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(if_s, if_t) statistical</td>
<td>rules(in G)</td>
<td></td>
</tr>
</tbody>
</table>

Ascending multilevel transfer systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT 1970—</td>
<td>Lemmatization</td>
<td>tree transformations</td>
<td>tree flattening</td>
<td>Ariane-G5-based</td>
</tr>
<tr>
<td></td>
<td>ECFG (gvt &amp; binding)</td>
<td>rules</td>
<td>grammar+dct.</td>
<td>fr-de-ru, en—my-th 80-87</td>
</tr>
<tr>
<td></td>
<td>gram. rules</td>
<td></td>
<td>rules</td>
<td>fr—en (BV/aero) 85-92</td>
</tr>
<tr>
<td></td>
<td>interactive disamb.</td>
<td></td>
<td>rules</td>
<td>fr—en-de-ru (LIDIA) 90-96</td>
</tr>
<tr>
<td></td>
<td>if not enough space</td>
<td></td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MG: dict. + gram.</td>
<td>IPPI, Moscow, 1977—</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>rules</td>
<td>HICATS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(in G)</td>
<td>Hitachi (1990—-)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Jemah USM, NUS (1990—-))</td>
</tr>
</tbody>
</table>

Horizontal multilevel transfer systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT 1990—</td>
<td>Lemmatization</td>
<td>tree transformations</td>
<td>tree flattening</td>
<td>TDMT (for SLT)</td>
</tr>
<tr>
<td></td>
<td>ECFG (gvt &amp; binding)</td>
<td>rules</td>
<td>grammar+dct.</td>
<td>ATR, 1992—1998</td>
</tr>
<tr>
<td></td>
<td>gram. rules</td>
<td></td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td>interactive disamb.</td>
<td></td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td>if not enough space</td>
<td></td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MG: dict. + gram.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(in G)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Examples: raw MT for revision...

<table>
<thead>
<tr>
<th>ORIGINAL</th>
<th>RAW TRANSLATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Après essai, s’assurer du fonctionnement correct de l’ensemble raccord.</td>
<td>After test, check that the coupling assembly works correctly.</td>
</tr>
<tr>
<td>Porter sur celle-ci la date de la dernière réception ou révision.</td>
<td>Write on this one the date of the last reception or of service.</td>
</tr>
<tr>
<td>Effectuer la vidange générale et la purge du carburant (voir chapitre 12).</td>
<td>Drain in a general manner and bleed fuel (see chapter 12).</td>
</tr>
<tr>
<td>Le bouchon a pour but d’assurer la protection d’un raccord auto-obturant lorsque celui-ci n’est pas utilisé au sol ou en vol.</td>
<td>The trap is used for carrying out the self-sealing coupling protection when this one is not used at the ground or in flight.</td>
</tr>
</tbody>
</table>

Semantic transfer systems

<table>
<thead>
<tr>
<th>Type</th>
<th>Analysis</th>
<th>Transfer</th>
<th>Generation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT</td>
<td>1982—</td>
<td>segmentation lemmatization direct programming tree transformations rules</td>
<td>dictionary + tree transformations rules</td>
<td>tree transformations rules</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MG: dict. + gram. rules</td>
<td>MU Kyodai, 82-87</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MAJESTIC JICT, 87—</td>
</tr>
</tbody>
</table>

French-Korean by IF (1)

...with BV-aero/FE (2)

| Enclouer légèrement le joint neuf de liquide d’utilisation. | Slightly coat the new joint with operating fluid. |
| Ouvrir progressivement le robinet (3), appliquer une pression jusqu’à 1,5 bar jusqu’à l’allumage du voyant lumineux DS2 et l’extinction du voyant DS1. | Gradually open tap (3), apply a pressure up to 1.5 bar until the light DS2 switching on (ignition) and the signal lamp DS1 extinction. |
| Ouvrir progressivement le robinet (3) jusqu’à obtenir une pression de 9 bars. | Gradually open tap (3) until a pressure of 9 bars is obtained. No explicit transformation is performed. |
| Procéder à la dépose des panneaux. | Remove the panels. |

IMPORTANT: avant de déposer ou de reposer le panneau central intrados de voiture, il est nécessaire de procéder à certaines modifications.

IMPORTANT: before removing or reinstalling the lower central wing panel, it is necessary to proceed with some modifications.
French-Korean by IF (2)

Conceptual transfer systems (IL with separate lexicon)

Interlingual level

- Conceptual transfer
- Semantico-linguistic interlingua

<table>
<thead>
<tr>
<th>Type</th>
<th>Enconversion</th>
<th>Conc. transfer</th>
<th>Deconversion</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBMT</td>
<td>Lemmatization</td>
<td></td>
<td>in principle none</td>
<td>graph-string transformations rules</td>
</tr>
<tr>
<td>1980—</td>
<td>direct or rules</td>
<td></td>
<td></td>
<td>ATLAS-II</td>
</tr>
<tr>
<td></td>
<td>string-graph</td>
<td></td>
<td></td>
<td>Fujitsu, 1980—</td>
</tr>
<tr>
<td></td>
<td>transformations rules</td>
<td></td>
<td></td>
<td>PIVOT Nec, 1983—</td>
</tr>
<tr>
<td>RBMT</td>
<td></td>
<td></td>
<td></td>
<td>ULTRA NMSU, 89-95</td>
</tr>
<tr>
<td>1980—</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBMT</td>
<td></td>
<td></td>
<td>navigation in set of UWs</td>
<td>depending on partners rules</td>
</tr>
<tr>
<td>1997—</td>
<td></td>
<td></td>
<td>UNL lexicon</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UNL 1996—</td>
</tr>
</tbody>
</table>

Graphemic level

- Text

Knowledge-based systems: explicit understanding, IL linked with an ontology

Deep understanding level

- Ontological interlingua

<table>
<thead>
<tr>
<th>Type</th>
<th>Enconversion</th>
<th>Deconversion</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>KBMT</td>
<td>lemmatization &amp;</td>
<td>planning deep-str</td>
<td>KBMT-89</td>
</tr>
<tr>
<td>1980—</td>
<td>EPSG+f-structures +pseudo-unification rules</td>
<td>rec. descent rules</td>
<td>ONU, 1989—91</td>
</tr>
<tr>
<td></td>
<td>(using UP)</td>
<td></td>
<td>KANT/Catalyst</td>
</tr>
<tr>
<td></td>
<td>all but discourse</td>
<td></td>
<td>ONU+Caterpillar</td>
</tr>
<tr>
<td></td>
<td>elements + dict+rules</td>
<td></td>
<td>en--fr-sp-de?</td>
</tr>
<tr>
<td></td>
<td>+ interactive</td>
<td></td>
<td>1992--</td>
</tr>
<tr>
<td></td>
<td>desambiguation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBMT</td>
<td>dictionary + FST</td>
<td>dictionary + FST</td>
<td>CSTAR-II &amp; Nespole!</td>
</tr>
<tr>
<td>1997—</td>
<td>rules</td>
<td>rules</td>
<td>GETA 97-03</td>
</tr>
<tr>
<td></td>
<td>IF is only a pragmato-</td>
<td></td>
<td>ETRI (Korea) 97-99</td>
</tr>
<tr>
<td></td>
<td>semantic representation no mapping to $\Omega$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMT</td>
<td>learned from (string,IF) KB statistical</td>
<td>no mapping to $\Omega$</td>
<td>CSTAR-II &amp; Nespole!</td>
</tr>
<tr>
<td>2003—</td>
<td></td>
<td></td>
<td>first 98-03</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Master-1 IBM 2003</td>
</tr>
</tbody>
</table>

Graphemic level

- Text

What kind of IL to choose?

- IL+ontology
- restricted domain
- high precision applications
  - cf. CLang (Mooney)
  - beware, Q costlier than gram+dict!
  - machine learning possible

- Pragmatico-semantic IL
- IF of CSTAR/Nespole!
- task- and domain-related reservations in tourism
- medical assistance
- both MUST be restricted
- works very well then
- machine learning possible

- Semantico-linguistic IL
- ATLAS-II, PIVOT better: UNL
- all domains/tasks: IL has to be grounded on a NL understandable
- by most developers anywhere
- amenable to machine learning
UNL: an *anglo-semantic interlingua*

see slides on UNL (previous session)

Conclusions of part 1 (1)

- The distinctions RBMT, EBMT, an-EBMT, (P)SMT... concern the computational architecture only (PROCESSES)
- *The rawer the corpora, the larger they must be*
- SMT/PSMT is for *niches for the rich* (languages, texts)
  - few parallel corpora of 200—800 K pages
  - to build them from scratch is 2 to 3 times more expensive than to build a classical large RBMT system
- IL-based MT can use any computational framework
  - statistical, analogical, rule-based, hybrid
  - all depends on available corporal / linguistic / human resources

Conclusions of part 1 (2)

- Many applications need an *adequate IL*
  - all applications needing to manipulate content in a strongly multilingual setting
- That is possible,
  - with empirical as well as expert / hybrid architectures
- The best structures for HQ transfer and autonomous pivot
  - spa-structures
  - UNL

Machine Translation, part 3
Computational architectures of MT systems

Christian BOITET
GETALP-LIG-UJF, Grenoble, France
Christian.Boitet@imag.fr
Introductive words

- No time to present spa-structures and UNL in lecture 2 so we will cover them in the first half
- The goals of L3 will be to:
  - give an example of a pivot with autonomous lexicon
  - introduce methods to solve the fundamental problems of NL analysis
    1. high intrinsic non-determinism,
    2. fuzziness,
    3. size
  - show how these methods are embedded in both empirical and expert computational MT architectures
The remaining goals should wait until L4:
  - analyze the computational models behind rule systems
  - introduce to 3 main kinds of rule-based SLLPs
  - give examples of some

Outline of part 3

- Recap: Taxonomy of linguistic architectures (direct… deep pivot)
- The best structures for HQ transfer and autonomous pivot spa-structures UNL
- Taxonomy of algorithmic methods
- Taxonomy of computational architectures
- What are rules, and ≠ types of rule-based specialized languages

Recap: linguistic vs. computational architectures

<table>
<thead>
<tr>
<th>Linguistic architecture objects see Vauquois' triangle</th>
<th>Computational architecture automatic processes human interaction, if any</th>
</tr>
</thead>
<tbody>
<tr>
<td>intermediate representations direct, semi-direct, transfer (≥ 7 variants) 2 lexical spaces IL (≥ 2 variants) 3 lexical spaces</td>
<td>programming paradigms direct programming RBMT (rules, automata...) corpus-based SMT, PSMT (unsupervised) EBMT (≥ 3 variants) ± supervised hybrid</td>
</tr>
</tbody>
</table>

Recap: linguistic architectures in MT: Vauquois' triangle
Computational architectures

The best structures for HQ transfer: spa-structures embedded in multilevel trees

(Vauquois 1974)

- based on lexicalized abstract constituent trees
  - articles, auxiliaries removed and replaced by attributes
  - same for presentation tags (italics, bold...)
  - non-connex compounds regrouped (give... back \rightarrow give-back)
- however, order is kept as much as possible
  - hence, some control is possible during transfer
- multilevel: on the same node,
  - ARG arg0, arg1, arg2, arg3, trl10, trl21
  - SR agent, patient, benefit, cause, concession, time, space...
  - SF subject, obj1, obj2, agcomp, complement, epith, circumstancial...
  - and other surface or deep actualization attributes (determination...)

Computational architectures: methods and tools

- Taxonomy of algorithmic methods
  - how to fight non-determinism, fuzziness and noise of correspondences between successive levels of description
- Taxonomy of computational architectures
- What are rules,
  - and \# types of rule-based specialized languages

Taxonomy of algorithmic methods

- Combinatorial methods
  - Compute all solutions
    - if possible using dynamic programming to factorize subproblems
      - first example: Cocke’s algorithm for CNF (binary normal form) grammars [also CJK algo]
  - Then use successive filters to reduce the set of outputs
    - when obtaining empty set, retract 1 step
    - produce a set of outputs ordered by chance only
      - ex: ETAP-3 system (Moscow)
- Same, with scores
  - Order final output set by scores
    - scores can be used effectively even if intuitively set [Slocum 1984]
      - ex: NEON system (Shi X.D., Xiamen), METAL (Slocum), LMT (McCord, IBM)
  - BUT
    - these methods do not handle ambiguities explicitly
    - they are often quite slow
Taxonomy of algorithmic methods (2)

• **Heuristic methods**
  - Simple back-tracking à la Prolog
    - possibility to ask for "the next one"
    - order of outputs depends on precise order of rules & matching algorithm
    - usually no dynamic programming (or "memoization")
    1. fast if producing only the k-first (k=1, 2…10 or so)
    2. slow to very slow if asked to produce all results
• **General heuristic methods (if using scores)**
  - A*, etc. used in empirical and expert approaches
• **Particular heuristic methods**
  - Tournament (ex: CETA [Berlioux 1963])
  - Specific heuristic functions
    - ARRET-, -STOP-, -INIT-, -FINAL-, -ARF-, -ARD- in morphological analysis (ATEF)
  - Deterministic nodes in ATNs
  - Reinitialization nodes in control graph (ROBRA)
• **Representations containing disjunctions**
  - Trees with dotted lines
  - LPNLP at IBM then G at MicroSoft [Heidorn]
  - to factorize several attachment possibilities
  - Vauquois' m-structures: use of tactical attributes to encode some ambiguity types

Characteristics of empirical approaches

• **statistical (rather, probabilistic) phases**
  - Preparation: use of statistics to produce estimates of probabilities of
    - monolingual events (n-grams, treelets, POS ambiguities…)
    - bilingual events (string alignments, phrase alignments, bi-treelets…)
  - typically: distortion model, alignment model, target language model
  - Execution: Viterbi-like dynamic programming algorithm
    - statistical decoding
• **example-based phases**
  - Using pre-annotated examples
    - pairs of trees (tree_S, tree_T) —> triples (treelet_S, treelet_T, score)
    1. MSR 1998-2002, using a MINDNET
    2. Kurohashi (J-E, J-C)
  - Need of a parser and a generator for all languages in the system
  - S-SSTC (USM, Penang)
• **Using raw examples**
  - analogical MT (ALEPH, Y. Lepage, ATR, 2005)

How to distinguish?

• **Example-based approaches**
  - Examples are used during the computation
  - possibility to trace the examples actually used in a given translation
• **Statistical (probabilistic) approaches**
  - Examples disappear
    - they are replaced by a very large number of small factoids [E. Hovy]
    - factoids = very elementary rules?
  - All necessitate ToolKits (TKs) but
• **For methods using raw examples**
  - Tools for managing / testing... very large corpora
• **For methods using prepared (annotated) examples**
  - Annotation / revision tools
What are rules, and ≠ types of rule-based specialized languages

• Rules
  - Rules of well-formedness (grammars),
  - Transition rules (automata),
  - Rewriting rules (on strings or trees).
• There must be a separate engine that
  looks for possible occurrences of rules
  solves conflicts if any
  applies the rules
    - Systran is not a RBMT system because analysis is not rule-based
• SLLPs (Specialized Languages for Linguistic Programming)
  3 types: creation, addition, substitution
  Examples
• Ensuring and relaxing decidability of SLLPs
  Examples
  - ATNs, Q-systems, ATEF, ROBRA, GRADE

Rules of well-formedness (grammars)

• Context-free grammars
  \[ G = (N, T, P, S) \]
  - Non-terminals, Terminals, Productions, Axiom \( S \in N \)
    - Rule \( P_i: A \rightarrow X_1 \ldots X_p \)
      equivalent to elementary tree \( ET_i = A (X_1 \ldots X_p) \)
  Syntax trees \( G \) defined inductively:
    - Base = \( T \cup \{ \$ \} \)
    - Rules = \{insertion\}
  Insertion:
    - combines an elementary tree and a previously constructed tree
    - \( A (X_1 \ldots B \ldots X_p) \), \( B (\$F) \rightarrow A (X_1 \ldots B(\$F) \ldots X_p) \rightarrow F = a \ forest \)
  Well-formedness:
    - \( w \in L(G) \) iff there is a syntax tree \( \Delta \) such that
      1. Root \( \Delta = S \)
      2. Leaves \( \Delta = w \)
      3. or equivalently:
        1. \( S \) is at the root
        2. \( w \) is the LR concatenation of terminal symbols (or \( \$ \)) on the leaves
        3. A grammar rule is "realized" in any internal node

Insertion

• Many similar formalisms define structures by constraints
  dependency grammars
    - traditional (with trees): insertion only
    - from the 60s (Courtin): definition of legitimate arcs:
      [Category_1, Category_2, dependency relation, relative position]
  semantic nets (Sowa)
  TAG (tree adjunction grammars)
    - insertion and adjunction in trees
• Very rich set of formalisms adding attributes to trees
  Unification grammars
    - FUG (M. Kay 1978), GPSG (Gazdar & Pullum 1982), LFG, HPSG…
    - U-TAG
  Attributed dependency grammars
    - Moscow ETAP systems (Apresyan, Boguslavskij, Iomdin…)
    - Prague tectogrammar (Sgall, Hajic’ova)
Transition rules (automata)

- **Simplest example: FST**
  Finite-State Transducers
  - Compute *functions*, not only yes-no decisions (acceptance/rejection)
  - Often used in morphological analysis (ex: Systran, Ariane…)

- **Oldest examples:**
  CNF-based dependency parsing:
  - First CETA system 1962-70 [Vauquois 1975]
  - Compute (abstract) trees from strings
  - Analogous to YACC for NL (handles non-determinism)
  - Used in analysis step in MT:
    AS-Transac (Toshiba), ProMT & Reverso, PAHOMT

Rewriting rules (on strings or trees)

- **Oldest example: transformational grammars**
  Chomsky’s Standard Model 1 (≈1960)
  - Deep structures → Surface structures
  - Rule: pattern on (recursive) cut of current tree → replacement
  - Complex control mechanism(s)
  **Implementation by Joice Friedmann**
  - OK for testing the theory in generation, impossible in analysis (Petrick)
  **Independent tree-transformational systems**
  - Metalanguage of CETA, Grenoble [Vauquois, Veillon, Veyrunes 1967]
  - CETA & ROBRA languages [Chauché 1974, Boitet 1978]
  - GRADE [Nakamura 1985], GWS in Tapestry [Tong LC 1989], & Hitachi (name?)
  - Extension to multidimensional trees TELESI in SYGMART system [Chauché 1982]
  **Current research at ISI**
  - SMT framework, 2005—
  - Attempt to learn tree automata à la Thatcher & Wright [1972]

Recap and what should follow

- **Taxonomy of algorithmic methods**
  - How to fight non-determinism, fuzziness and noise of correspondences between successive levels of description
- **Taxonomy of computational architectures**
  - Empirical computational architectures (statistical, example-based with and without annotations)
  - Expert computational architectures (“procedural” methods, “rule-based” methods)
- **What are rules, and ≠ types of rule-based specialized languages**
  - Rules of well-formedness (grammars), transition rules (automata), rewriting rules (on strings or trees)
- **Next:**
  - Examples of SLLPs (Specialized Languages for Linguistic Programming) of 3 types (creation, addition, substitution)
  - Ensuring and relaxing decidability of SLLPs
    - Examples: ATNs, Q-systems, ATEF, ROBRA, GRADE

Machine Translation, Part 4
Engineering of MT and CAT systems

Christian BOITET
GETALP-LIG-UJF, Grenoble, France
Christian.Boitet@imag.fr
Introduction

- SLLPs (Specialized Languages for Linguistic Programming)
  so we will cover them in the first half:
    Examples of SLLPs
    - SLLP = Specialized Languages for Linguistic Programming
    - of 3 types: creation, addition, substitution
  Ensuring and relaxing decidability of SLLPs
    - Examples: ATNs, Q-systems, ATEF, ROBRA, GRADE

- Goals of part 4:
  show the convergence of MT & TA
  introduce the notion of EDL for MT
  - Environments for Developing Lingware
  introduce to software engineering for lingware engineering
  - Principles of SLLP development
  - Principles of Lexical Data Base development

Outline of part 4

- Convergence/integration of MT & TA (with TM) → CAT
- From homogeneous MT systems to heterogeneous CAT systems.
  Convergence of evolutions since 1980
- Environments for Developing Lingware (EDL)
  EDL specific to 1 MT system (MTS)
  Meta-EDL for heterogeneous MTS, towards Integrating EDL
- Good practices in implementing specialized languages (SLLPs)
- Design of Lexical Data Bases for MT systems
  PIVAX for heterogeneous MT systems
  sharing a common lexical pivot

SLLP (specialized language for linguistic programming)

- Almost always rule-based
  underlying models can be
  - usual grammar, transducer (automaton), rewriting system
  - working on various data structures (strings, decorations, trees, graphs)

- Almost always structured in components
  basic types
  - definition of attributes/variables
    ✓ -EXC- SF == (SUBJ, OBJ1, OBJ2, AGCP, NCP, EPIT, DET,...). ** cmt.
    ✓ -NEX- SEM2 == (CONCR, ABSTR, TIME, SPACE, SOLID...). ** comment.
  - definition of templates/classes
    ✓ TRANS/RB == TRANS=1, VAL1=GN. ** comment.
  - definition of dictionaries
    ✓ ‘PRIX_N’ == / cond1 /         / ‘PRICE’,...
                     /         / 0(1, 2) / 0: $AMBIG; 1: ‘PRICE_N’,…; 2: ‘PRIZE_N’…
  - definition of grammar rules
    ✓ and their organization (ex: control graph in ROBRA)

3 basic types of rule-based SLLPs

- Creation
  R/O input structure, R/W output structure
  - transduction-based model
  - examples: FST, ATN (on the whiteboard)
  Both structures can be different

- Addition
  R/W input structure, R/O output structure
  - rule firing triggers addition of new elements
  - when no rule can fire, do a cleaning act
    ✓ remove partial constructions used in building more complete ones
  - examples: Q-systems, chart parsers
  Both structures can be different

- Replacement
  R/W input/output structure
  - rewriting here is replacement
  - necessity of conflict resolution, more difficult programming
ATN (linear writing)

(S/ (PUSH NP/ T)
  (SETR SUBJ *)
  (SETR TYPE (QUOTE DCL))
  (TO Q1))
(CAT AUX T
  (SETR AUX *)
  (SETR TYPE (QUOTE Q))
  (TO Q2)))
(Q1 (CAT Y T
  (SETR AUX NIL)
  (SETR Y *)
  (TO Q4))
  (CAT AUX T
  (SETR AUX *)
  (TO Q3)))
(Q2 (PUSH NE/ T
  (SETR SUBJ *)
  (TO Q3)))
(Q3 (CAT Y T
  (SETR V *)
  (TO Q4)))
(Q4 (POP (BUILDQ (S+++(VP+)) TYPE SUBJ AUX V) T)
  (PUSH NP/ T
   (SETR VP (BUILDQ (VP (V+) *) V))
   (TO Q5)))
(Q5 (POP (BUILDQ (S++++) TYPE SUBJ AUX VP) T)
  (PUSH PP/ T
   (SETR VP (APPEND (GETR VP) (LIST *))
   (TO Q5)))

Q-systems: a q-graph, and addition of rhs of a rule

- Q-graph

- Rule & its application
  \[ a_1 + a_2 + \ldots + a_p = - b_1 + b_2 + \ldots + b_q \]

- Excerpt of a French morphological generation
  For articles:
  - ART(U*)  == ART1(U*)  / "PL-HORS-U".
  - ART(U*)  == ART2(U*)  / "NONPL." "PL-HORS-U".
  - ART2(U*)  == LES  / "DEF-DANS-U".
  -  == DES  / "IND-DANS-U".
  -  ==  / "NIL-DANS-U".
  - ART3(U*)  == ART3(U*)  / "POSS-DANS-U".
  - ART3(U*)  == LEURS  / "PL-DANS-U".
  -  == SON  / "PL-HORS-U" "ET-" "M-DANS-U".

Corrections:
- DE + UN  == D + ' + UN.
- NE + NOM(Y)  == N + ' + Y.
- DE + LESQUELLES  == DESQUELLES.
- A* + NOM(B*,U*)  == L + ' + NOM(B*,U*)
  / "A-DANS-LE", LA
  "ET-" "B-DANS-" A, E, O, U, E, I, Y.

© Ch. Boitet — MT lecture, part 4

1/12/2011
ROBRA

- Not enough time and too complex to give details…
  Principle: rewriting on decorated trees

Object tree traverses a CONTROL GRAPH using back-tracking
  • until it reaches an EXIT NODE
It is transformed by the TRANSFORMATIONAL GRAMMARS
  • associated with the traversed nodes
Left hand side (lhs) of rules are powerful parametrized PATTERNS
  • an OCCURRENCE of a lhs is a subtree
    ✓ not necessarily complete vertically or horizontally
  • possibility to express unordered of siblings under any node
  • generalized nodes: GN = tree + distinguished leave
  • top hat of pattern can be contextual
Other control structures
  • guarded iteration of grammars
  • guarded recursion: call of subgrammars or of transformational subsystems

Convergence/integration of MT & TA (+ TM) → CAT

- MT: Machine Translation
  the machine performs the translation step
  humans can help, it is still MT (HAMT)

<table>
<thead>
<tr>
<th>before</th>
<th>during</th>
<th>after</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre-edition</td>
<td>interactive disambiguation</td>
<td>post-editation</td>
</tr>
</tbody>
</table>

- annotating
- rewriting
  • during a phase
  • on a multiple IR
    (intermediate representation)
  • on 1 selected output
  • on a multiple output
    - lattice
    - confusion graph
  • co-edition of an IR
    - like UNL [Thai & Boitet 02]

- TA: Translation Aids
  the human performs the translation step
  the machine can help, it is still HT (MAHT)

Convergence/integration of MT & TA → CAT

- Addition of TA (translation aids) to MT
  Revision mode in Ariane-G5 (1982—)
  Virtuous circle in PAHO (1980—)
  Pensée, Yakushite.net (OKI electric)
  Feedback from post-editors (1985?, Keh-Yih Su, Taipei)

- Addition of MT modules to TA
  Integration of automatic dictionary look-up (Melby, 1982—)
  Integration of TM (IBM TM/1 1985, TM/2 1992—[2002 end of commercialisation])
    • TM = Translation Manager (©), then translation memory
    ✓ "TM/2 is in my estimation the first computer-aided translation package to really get it right." Expensive: US$6,300 for a single user [LM, October 1992]
  TM in all current TA tools (Transit, Déjà Vu, Eurolang, Trados, Similis…)

- Addition or integration of full MT to TA
  Often tried
    • LMT on TM, METAL, Ariane, Logos on EuroLang
  Often failed
    • because of absence of synergy between the TA dictionaries and the MT dictionaries

From homogeneous MT systems to heterogeneous CAT systems

This relates to the operational architecture of MT systems

- Homogeneous MT
  1 platform
  same tools used for all applications (language pairs or graphs)

- Heterogeneous MT
  parts can be developed independently
    • composition of phases/steps through the intermediate representations (IR)
    • linguistic & computational architectures can be different
    • computer tools ca be different (prog. languages, environments)
Convergence of evolutions since 1980

- Multi-engine MT systems
  - Pangloss (CMU)
  - VerbMobil (Germany, 1992-2000)
  - ALTFLASH (NTT, 2001—)
- The problem: how to combine the results of N MT systems? same problem as for combinatorial systems
  - scoring & voting mechanism
  - factorization of all results in a confusion graph
    - that allows to build a translation from parts of different outputs

Environments for Developing Lingware (EDL)

- EDL specific to 1 MT system
  - First idea and implementation of a development environment came from NLP, not CS!
    - that was in 1962 at Grenoble, for MT development (M. Quézel-Ambrunaz)
    - computer scientists were happy with the JCL, linguists had to be helped
- An EDL like Ariane-G5 manages
  - lingware components
    - creation, deletion, edition, compilation, listing of COMPONENTS of phases
    - declaration of attributes/types, templates, grammars, dictionaries
    - coherency/completeness checks
    - a kind of MAKE for an MT system
      - importance of separate compilability
- corpora for
  - development
  - test
- exploitation modules
  - compiled language pairs

Convergence of evolutions since 1980 (2)

- MT systems using non-linguistic knowledge
  - Interactive disambiguation
    - with questions directed to somebody understanding the text, not a linguist
    - The MT system behaves as if it would be intelligent
      - "implicit understanding"
  - Recourse to an external KB: database, knowledge base, ontology
    - Gerber 1994: a KB about chemistry (database)
    - KBMT-89 (CMU): ontology of the PC and the PC-5550 (Japanese)
    - Any KB can be used in principle to help.
  - The MT system has a real *interpretation function over a formalized domain*
    - "explicit understanding"
- Heterogeneity is necessary in that case
  - Different platforms, tools
    - dictionaries, rules, grammars / concept lattice, frames, methods, inference
  - Different developers
    - linguists, experts, cogniticians

Ariane-G5
Meta-EDL

- CASH, a meta-EDL for Ariane-G5 [E. Blanc] runs on any PC/Mac manages
  - copies of the source files (grammars, dictionaries, corpora)
  - specific tools (help to dictionary indexing, graphic editors on trees...)
remotely controls the native EDL
- sends commands and data (by socket, http, or SMTP)
  1. list compiled language pairs with FRE as source;
  2. update FRE-AM-DICT2 (morphological analysis) with 4 following items
  3. compile all phases of FRE-ENG
  4. execute on all texts of corpus Test2
- receives answers and data in the same format, stores them locally

- WICALE, a meta-EDL open to other MT/NLP systems [V. Carpena, Nguyen H.T.]

Towards an integrating EDL Reengineering the SLLPs of the distant EDLs Possibility to compile/execute locally or with the proper EDL
- less efficiently locally is OK
- delegation principle in another guise

WICALE

Navigation under WICALE

![Navigation under WICALE](image)
Good practices in implementing SLLPs

- **SLLP** = specialized language for linguistic programming
- **Compile into DATA**
  - not into large C/C++/java programs
  - data =
    - simple tables (compact binary representation with indexes…)
    - or byte code of a virtual machine implementing the computation model
  - this way, one builds a unique, small, table-driven program (the *engine*)
  - debugging far easier
  - more efficient, & amenable to SLLP-level debugging
- **Equip the SLLP with a 2-level debugger**
  - linguistic debugging
  - software debugging
- **Distinguish compilation and loading**
  - necessary to enable *incremental modifications*
  - avoid loading a 5 Mo module each time one translates 1 page (1,5—3 K!)
  - load permanently an *internal structure* (abstract + access functions + tactical variables)
  - perform *semantic checks* while loading, not while compiling

Design of Lexical Data Bases for MT systems

- **It is very difficult to design a lexDB for heterogeneous MT**
  - in general!
  - in particular, transfer dictionaries are too different
    - not same lexical units
    - not same linguistic theories, and associated codes
    - procedural choices using conditions on tree/graph structures
- **It is possible to do it for heterogeneous MT systems**
  - if they share a common lexical pivot
    - ex: PIVAX
      - developed over the Jibiki platform (Sérasset, Mangeot, Papillon project)
      - by Nguyen H.T. (2007-09)
      - The Papillon project started at NII in Aug. 2000
      - targeted MT system: UNL/U++

Lexical information handled by PIVAX

Recap

- **Taxonomy of SLLPs**
  - creation, addition, replacement
  - examples
  - principles of implementation
- **Why and how to unify MT and TA (translation aids) → CAT**
  - from homogeneous MT systems to heterogeneous CAT systems EDL and meta-EDL
- **Lexical databases to unify MT & TA lexicons**
  - PIVAX
- **Next**
  - Evaluation of MT
In short!

- **ABSTRACT.** External methods for evaluating MT systems define various measures based on MT results and their usage. While operational systems are mostly evaluated since long by task-based methods, evaluation campaigns of the last years use (parsimoniously) quite expensive subjective methods based on unreliable human judgments, and (for the most part) methods based on reference translations, that are impossible to use during the real usage of a system, less correlated with human judgments when quality increases, and totally unrealistic in that they force to measure progress on fixed corpora, endlessly retranslated, and not on new texts to be translated for real needs. There are also numerous biases introduced by the desire to diminish costs, in particular the usage of parallel corpora in the direction opposed to that of their production, and of monolingual rather than bilingual judges. We prove the above by an analysis of the history of MT evaluation, of the « mainstream » evaluation methods, and of certain recent evaluation campaigns. We propose to abandon the reference-based methods in external evaluations, and to replace them by strictly task-based methods, while reserving them for internal evaluations.

- **KEYWORDS:** Evaluation, Machine Translation, Usability Evaluation.
Outline of part 5

- Evolution of ideas in MT evaluation of text, then speech
- Meta-evaluation, formalisation & extension to other applications
  - EAGLES (1993-96), ISLE, FEMTI
- Explanation & critic of methods used currently in evaluation campaigns
  - Subjective measures (adequacy, fluidity)
  - Objective n-gram measures (BLEU & others)
- Taxonomy of external measures
  - task-related: diffusion (text) & communication (speech), comprehension (both) not task-related (± references)
  - Proposal: as for external measures, use only objective & cheap task-related measures

Evolution of ideas in MT evaluation (1)

- Distinction made since the beginning between
  - external measures
    - linguistic criteria: grammaticality, fidelity...
    - usage criteria: productivity, cost, delay...
    - always conflict between linguistic & usage criteria
  - internal measures
    - system design: linguistic & computational architecture
    - perspectives of improvements: quality coverage
    - ease of extension to: new languages new document types new tasks (assimilation → dissemination)
- Some reference points
  - ALPAC (66-67) (in)famous report... first broad evaluation
  - CMC/TAUM-Météo (77-) first task-related evaluation
  - EEC (76-90) first experts in evaluation (van Slype, Omnium)
  - JEIDA (89-92) multiple criteria (14) [Nomura]
  - ARPA (92-94) same line, less criteria
  - CSTAR-II (99) comparison with human performance [Sugaya al. 2001]
  - Nespole! (00-03), IWSLT (04-), NIST... mixed measures

Evolution of ideas in MT evaluation (2)

- 1992, end of Eurotra project:
  - many MT researchers convert to MT evaluation research
- research separates in 2 directions
  - deployed systems (MT & TA): productivity, time gained
    - see http://www.oocities.org/mtpostediting/ [Jeff Allen]
    - not very popular among researchers
      1. too simple (!)
      2. usable only on used systems
  - extreme refinement of criteria for linguistic quality
    - ARPA campaigns (1992-1994) MT proficiency Scale project
    - looking for automatic evaluation methods
    - trying to eliminate the need for human judges
      2002: BLEU [Papineni, IBM], then NIST
    - leading to OBJECTIVE METHODS (based on n-grams)
      - correlate less with human judgments when quality increases!
      - 1 exception: WNM [Babych & al. 2004], rarely used

BLEU

- Papineni et al. 2002 (IBM)
  - BLEU precision, $p_n$, is associated to 1 n-gram.
  - BLEU brevity penalty,
  - $|c| = \text{total length of candidate translations}$
  - $|r| = \text{total length of reference translations}$
  - $\text{BLEU} = \exp(\frac{1}{|c|} \log |c|)$
  - BLEU is its probability of occurrence in the whole corpus
- Finally
  - $\text{BLEU} = BP_{\text{BLEU}} \cdot \exp \left( \sum_{n=1}^{N} \frac{1}{N} \log p_n \right)$
Meta-evaluation, formalisation & extension to other applications

- **Meta-evaluation**
  research on MT evaluation itself
- **Formalisation**
  EAGLES-EWG-1 (1993-94)
  ✓ Expert Advisory Group on Language Engineering Standards
  - it would be more profitable to measure if a translation is good enough for a specific need
  - than to try to define a necessarily too abstract notion of quality in general
- **Extension to other applications**
  proposal of a quality model for NLP systems in general
  - influenced by ISO/IEC 9126
  - hierarchy of properties, leaves are measurable attributes
  EAGLES-EWG-2 (1995-96)
  - consolidation of methodology
    ✓ spelling checkers, grammar checkers, Translation Memories

ISLE (1999-2002) and FEMTI proposal

- International Standards for Language Engineering, 1999-2002
  - FEMTI proposes a classification of main features defining a usage context (=translational situation)
    - user type, task type, input data
    - quality features for systems
      - hierarchy of categories
      - leaves have internal or external attributes (=metrics)
      - upper levels coincide with ISO/IEC 9126 standard
    correspondence between these 2 classifications
    - allowing to find features & quality sub-features adapted to each usage context
    - associated attributes & metrics
  - FEMTI methodology geared towards various user categories
    - potential MT users can
      - define the features most important for them
      - choose the system best adapted to their needs
    - potential users & developers wanting to compare MT systems can
      - choose the features nearest to their situation
      - determine the pertinent evaluation measures & tests
    - developers can
      - identify users' needs and find niches for their applications.

Explanation & critic of methods used currently in evaluation campaigns

**Subjective measures (adequacy, fluidity)**

- adequacy always a positive number
- fluidity often correlates with adequacy

**Objective n-gram measures (BLEU & others)**

HWER (GALE project)

**Subjective measures (NIST++)**

<table>
<thead>
<tr>
<th>Score</th>
<th>Fluidity (this utterance is ~)</th>
<th>Adequacy (~of the meaning)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>flawless English</td>
<td>All</td>
</tr>
<tr>
<td>4</td>
<td>good English</td>
<td>Almost all</td>
</tr>
<tr>
<td>3</td>
<td>non-native English</td>
<td>Much</td>
</tr>
<tr>
<td>2</td>
<td>non-standard English</td>
<td>Little</td>
</tr>
<tr>
<td>1</td>
<td>ununderstandable</td>
<td>Nothing</td>
</tr>
</tbody>
</table>
Taxonomy of external measures

- task-related: diffusion (text) & communication (speech), comprehension (both)
- not task-related (± references)

Proposal

- As for external measures
  
  use only objective & cheap task-related measures
Conclusion

- Current objective evaluation measures are OK as internal measures of progress towards a goal (references) are not OK to evaluate systems in operation
- Objective task-related measures should be introduced they necessarily involve some human participation but they may have a null or only marginal cost
- N-gram based should be abandoned as external measures
- For next time

Corpora of translations
- Why it is important to unify the large variety of these corpora
- How it can be done
Integration of post-edition in the reading context
- The IMAG concept

References on MT evaluation (1)


References on MT evaluation (2)


References on MT evaluation (3)


References on MT evaluation (4)


References on MT evaluation (5)


Motivations for unifying translation corpora

• **Practical**
  - convergence / synergy between MT-TA-HT
  - reuse HT corpora as TM (translation memories)
  - reuse HT corpora to build better empirical MT systems
  - annotate relatively small HT corpora to build EBMT systems
  - check / improve / evaluate translation corpora
    - hence, handle them as documents to be post-edited
    - evaluate parallel documents, not only sets of parallel sentences

• **Theoretical**
  - clarify many apparently simple questions
    - what is a corpus? how to define it in a generic way?
    - what is a TM? a particular type of corpus? a particular type of dictionary?
    - what is a TU (translation unit)?
    - what is a segment, a textlet, a chunk, a phrase, a term, a word?
    - what is the context(s) of a segment?
  - use answers as a foundation to build an OS for translation corpora
Outline of part 6

- Motivations for unifying translation corpora
- Basic notions — usual definitions
- Variety of translation corpora & associated notions
- Toward a solution: principles, revised notions
- The case of phrasebooks

Basic notions — usual definitions (1)

- Word
  - from HT: typographical word
  - the unit to pay translators
  - does not apply to Japanese, Chinese, Thai, Lao, Khmer
  - applies to Vietnamese even if there is no (linguistic) word separator
- Page
  - from HT
    - 250 (typographical) words or 1400 characters — English, French
    - ≈ 1 A4 page, double-spaced, Times 12
    - 400 characters — Japanese, Chinese
    - using page as a unit is less prone to misunderstandings
- Segment
  - from TA: unit put in a translation memory
  - corpus = {segments}?
    - sentence or smaller but full unit (title, menu item)
    - usually a record (fields for source language, target languages)
    - monolingual or multilingual segments
      - multilingual in XLIFF
      - but can be different
        - 1 sentence → 2 sentences

Basic notions — usual definitions (2)

- Translation Unit
  - HT: 1 sentence or 1 paragraph, or 1 turn (speech)
  - TA: the smallest possible (to retrieve more)
  - MT: many possibilities
    - segment or infra-segment (Google) if segmentation problems
    - from supra-segment to infra-segment (Systran)
    - from segment to page to section to full document (Ariane, Sygmart)
- Document
  - From HT: input or output file in translation
    - corpus = {documents}?
- Text
  - from MT: input or output file in translation
    - corpus = {texts}?
      - textual part of a document
      - nothing usually said about the non-textual parts (formulas, etc.)

Basic notions — usual definitions (3)

- Corpus
  - from literary studies: closed set of homogeneous documents
    - homogeneous at some level, fuzzy notion (cf. Bible)
  - in expert MT: set of texts
    - ex: 1986 bulletins of the Russian Referativnyij Zhurnal
  - in SMT and evaluation campaigns: aligned translation units
    - ex: BTEC. TU is usually 1 segment
      - sometimes also: segment | segment | segment…
- Context
  - HT: document type, company
    - appareil → aircraft, airplane, device…
  - TA: ± occurrences in document(s)
    - TM2 has it, not Trados
  - MT: domain, genre, client
    - + words before/after
Some examples of translation corpora (1)

- EuroParl, JR-acquis, BTEC... collected pairs <SSeg_L1, TSeg_Li>
  - textual context lost (or corpus fabricated like BTEC)
  - more languages added by HT, but
    1. distinction SL, TL often lost
    2. history of translation lost (often, L1 → Lp → Lj)
  - relatively large (20Mw, 1—3.5 Mw)
- IBM TMs
  20Mw/year English → 25 languages
  - organized by language pairs
  - textual context preserved
    ✓ <English, French 1>: list of occurrences for that translation
    ✓ <English, French 2>: list of occurrences for that translation
  - possibility to see where a translation has been used

Other reasons for the variety

- Evaluation campaigns
  input is preprocessed
  - words separated (even in Japanese)
  - punctuation separated
    ✓ c’est-à-dire → c’est-à-dire
  - decapitalisation
    ✓ Where goes he go? He wants to Bill Gates. → he wants to bill gates
      prepare a bill for some gates...
    → idea of NORMALISATION
  input can be more than text
  - N outputs of ASR, with scores
  - a lattice output by ASR, with scores
  - a path to a sound file (.waw)
- Interpreted dialogues (ERIM) for studies towards better MT
document = dialogue
  - each turn is a .waw file
  - context includes the speaker (Agent, Client, Interpreter)

Towards a solution

- Consider the largest units as DOCUMENTS
  Corpus = (document)
  Define the structure of the corpus at 2 levels
    - as is done for lexical databases in Jibiki
    - macro-structure: logical parts of a document (often files) ex: EOLSS
      1. main file .html file
      2. satellite files icons, etc.
      3. companion files .uni (segmentation + normalization + UNL graphs)
    - micro-structure: detailed definition of each part
      1. definition of units
      2. format

Some examples of translation corpora (2)

- Ariane-G5 corpora of texts
  textual hierarchy for each corpus
  - defined by a list of separators: <p/> <h1> <h2> <h3> <h4>
  - each is an occurrence and will terminate a hierarchical unit
  - segmentation
    - into translation units, larger than segments: [min..max] length
    - within a TU, the linguistic processor segments in sentences, titles...

\begin{figure}
\centering
\includegraphics[width=\textwidth]{tu.png}
\caption{Translation Unit (TU) diagram}
\end{figure}

\begin{itemize}
\item Good weather.
\item Japan is great.
\item Cherry trees are beautiful.
\item Nihonkai also.
\end{itemize}
Towards a solution

- Allow multiple segmentation
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
  - \[a > b/2 \] (1)
- Allow recursive segmentation
  - footnotes
  - text in TITLE attribute of a link
  - text within javascript code
    - identify language of each piece of text
- Allow multiple normalization
  - to handle on-textual elements in segments
    - special occurrences

because different MT systems use different segmentations (1 each)

Duality of translation memories

- like wave + corpuscle...
- TM = dictionary (of synonymous segments)
  - implementable as lexical database
- TM = document
  - then a segment must
    - have a unique source language
    - contain its translation
  - if recycling a bilingual parallel document
    - SL not known, maybe 2 versions may be considered valid originals
    - then build 2 different segments

The case of phrasebooks

- New phenomenon: segments with VARIABLES over classes
  - I would like \#number [cup | glass] of [juice | coffee]
  - or better
  - I would like \#number $drink_recipient of $drink
    - with classes defined recursively
    - $drink: name of the class (subdocument)
    - values are bisegments
      - juice
      - coffee
  - Some interesting things
    - influence of \#number on other words: cup/cups
    - variants
      - a sentence (phrase)
        - is usually 1 segment
        - sometimes has also an explanation and a precision
      - ex: lassi
        - explained (only in non Indian languages)
        - precision possible on sweet / sour lassi

Possible structuration

```html
<entree id="xxx" type="titre"...>
<lgsource lang="eng">
<segment id="xxx" type="contenu" lang="eng"...>
<org>In restaurant</org>
<rev pers="Xan" date="xxx" level="***" score="11">In a restaurant</rev>
<mt target="fra" tool="Google" date="xxx">Dans le restaurant</mt>
<mt target="fra" tool="Systran" date="xxx">Dans restaurant</mt>
<pd target="fra" pers="Xan" date="xxx" level="***" score="13">Au restaurant</pd>
<mt target="ger" tool="Systran" date="xxx">In dem Restaurant</mt>
<pd target="ger" pers="Xan" date="xxx" level="***" score="13">Im Restaurant</pd>
</segment>
</lgsource>
```
Dans un restaurant

Au restaurant

In restaurant

In the restaurant

In a restaurant

In einem Restaurant

Im Restaurant

It is not possible to enter a restaurant after 8 pm even in large cities, last order will be at 8:30 pm

Il n’est pas possible d’entrer dans un restaurant après 8 heures même dans les grandes villes, la dernière commande sera à 8h30 pm

Il n’est pas possible d’entrer dans un restaurant après 8h même dans les grandes villes, la dernière commande sera à 8h30

Es ist nicht möglich, geben Sie ein Restaurant nach 8 Uhr auch in den großen Städten, letzte Bestellung wird um 8,30 Uhr

Es ist nicht möglich, in ein Restaurant nach 8 Uhr Abends einzutreten, selbst in den großen Städten, die letzte Bestellung wird um 8.30 Uhr stattfinden
Conclusion: important points

- The notion of segment varies with systems
  - documents should be multi-segmented
- Translation units (TU) may be segments, infra-segments, super-segments
- Segments may recursively contain subdocuments
  - (e.g., text in balloons)
- Context is important
  - textual context (basic): where does a segment appear?
    - crucial in technical translation
      - (airplane ≠ aircraft ≠ plane…)
    - linguistic, situational, dialogic contexts
      - crucial to solve some important problems (anaphora, ellipsis, tense agreement, gender of addressee, politeness expression, etc.).
- Varied and complex annotations are needed
  - attached to segments (± paragraphs, sections…)
  - concerning 1 language at a time (e.g., POS or linguistic trees)
  - concerning 2 languages (various alignments), or all languages (semantic or pragmatic representations)
- Multimedia corpora must be considered
  - spoken dialog translation: sound files, NL transcriptions

Conclusion: perspectives

- Towards SECTra++
  - to implement and test all these ideas
  - while adding programmability, and possibility to create and not only exploit corpora
- Most important
  - modularity embodied in the delegation principle
    - minimal version of a functionality offered
    - but API + synchronization method with any (better!) external one
    - ex: segmentation, transcription, call to MT…
  - architecture based on large-grained agents
    - not only servers
    - enable autonomy, in particular to handle infinite background loops
      - call to MT, preparation of local dictionaries
  - proactive helps specialization to
    - tasks
    - sublanguages