NLP Project Full Circle

Vsevolod Dyomkin
prj-nlp-2, 2019-03-21
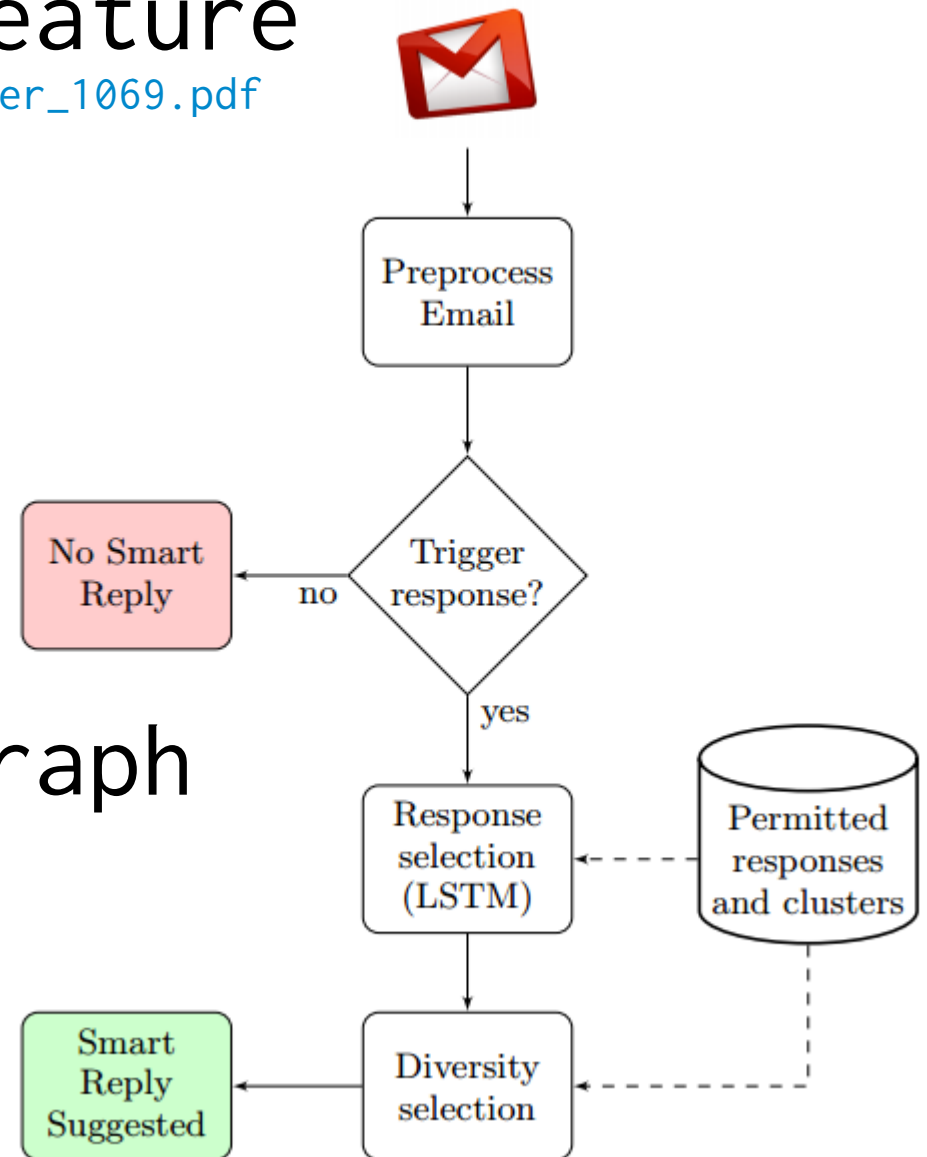
Глокая куздра штеко будланула бокра
И кудрячит бокрёнка

Example

Gmail Smart Reply Feature
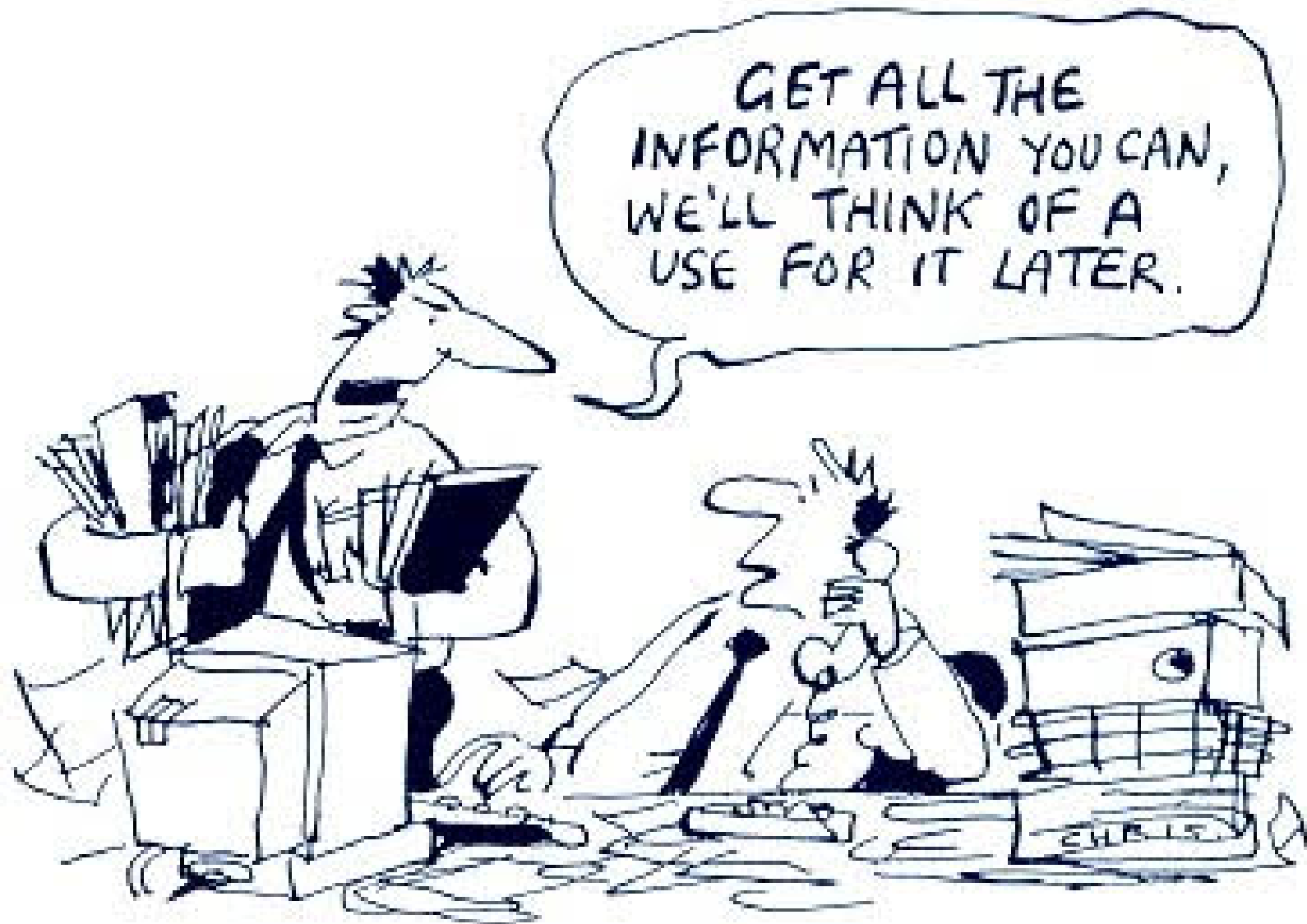http://www.kdd.org/kdd2016/papers/files/Paper_1069.pdf

* Data analysis
* Traditional NLP processing
* FNN
* LSTM
* Semi-supervised graph learning
* Rule-based post-processing
* Engineering

NLP Project Stages

1) Domain analysis
2) Data preparation
3) Iterating the solution
4) Productizing the result
5) Gathering feedback
   and reiterating

Domain Analysis

Domain Analysis

1) Problem formulation
2) Existing solutions
3) Possible datasets
4) Identifying challenges
5) Metrics, baseline & SOTA
6) Selecting possible
   approaches and devising
   a plan of attack

Глокая куздра штеко будланула бокра
И кудрячит бокрёнка

Problem Formulation

* Not an unsolved problem
* Short texts
* Support "all" languages

* The issue of UND/MULT texts
* Dialects?
* Encodings?

Interesting Examples

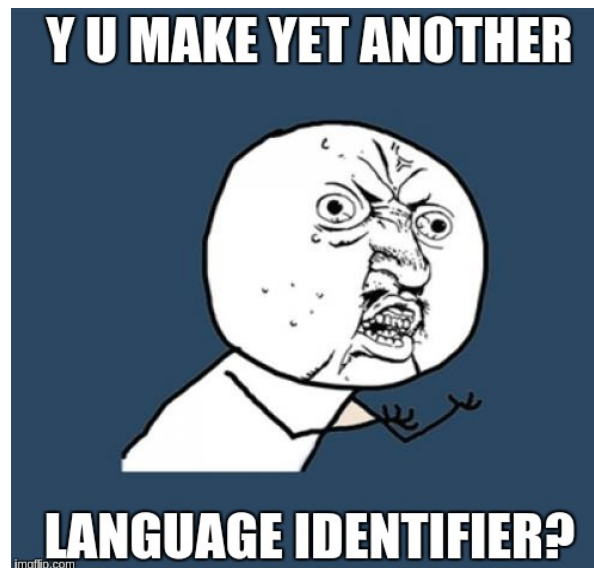| Text	Language	Explanation
Justin Bieber <3	und (Undefined)	NOT English; contains only a name.
Schalke XI v Chelsea: Fahrmann, Neustadter, Santana, Howedes, Uchida, Fuchs, Kirchhoff, Boateng, Hoger, Choupo-Moting, Huntelaar.	und (Undefined)	Contains only place/team/player names.
Ate spaghetti at La tratoria napolitana	en (English)	The name of the restaurant is in Italian, but the "main" language is English. An English-only speaker would understand this Tweet.
#NowListening Universo - Lodovica Comello @XYZ @XYZ	und (Undefined)	Italian song title and artist are just names. #NowListening is English but could be used by non-English speaker too.
#My #hot #naughty #neighbour #in #dallas: http://t.co/0dLJ 北京	en (English)	There is a Chinese word at the end, but the strongly prevailing language is English
Hahaha (•_•) (•_•)>⌐■-■ (⌐■_■) YEAHHH!	und (Undefined)	Emoticons and interjections only.
Que bonito!	und (Undefined)	Could be both Spanish and Portuguese
Pozor pozor	und (Undefined)	Could be Czech, Serbian, Croatian, Slovenian, ...
So warm in Berlin!	und (Undefined)	A valid sentence in both German and English
"Last Christmas" - Der Jose Carreras unter den Weihnachtsliedern.	de (German)	Contains an English song title and Spanish name, but is understandable to a German-only speaker.
Bécs <3	hu (Hungarian)	This is the Hungarian name for "Vienna", which is a proper name, but exists only in Hungarian
Estoy muy cansado voy a acostarme .... sooo tired goin to bedd	und (Undefined)	Strong mixture of Spanish and English, no clear "main" language

https://blog.twitter.com/2015/evaluating-language-identification-performance

Existing Solutions

* https://github.com/shuyo/language-detection/ (Java)
* https://github.com/saffsd/langid.py (Python)
* https://github.com/mzsanford/cld (C++)

Existing Solutions

* https://github.com/shuyo/language-detection/ (Java)
* https://github.com/saffsd/langid.py (Python)
* https://github.com/mzsanford/cld (C++)

* https://github.com/CLD2Owners/cld2
* https://github.com/google/cld3

Existing Solutions

* https://github.com/shuyo/language-detection/ (Java)
* https://github.com/saffsd/langid.py (Python)
* https://github.com/mzsanford/cld (C++)

* https://github.com/CLD2Owners/cld2
* https://github.com/google/cld3

Possible Datasets

* Debian i18n (~90 langs)

Possible Datasets

* Debian i18n (~90 langs)
* TED Multilingual (109 langs)

Possible Datasets

* Debian i18n (~90 langs)
* TED Multilingual (109 langs)
* Wiktionary (~150 langs)

Possible Datasets

* Debian i18n (~90 langs)
* TED Multilingual (109 langs)
* Wiktionary (~150 langs)
* Wikipedia (175 langs)

Test Data

* Twitter evaluation dataset
* Datasets with fewer langs
* Extract from Wikipedia
+ smoke test

Challenges

* Linguistic challenges
  - know next to nothing about
    90% of the languages
  - languages and scripts
    http://www.omniglot.com/writing/langalph.htm
  - language distributions
https://en.wikipedia.org/wiki/Languages_used_on_the_Internet#Content_languages_for_websites
  - word segmentation?

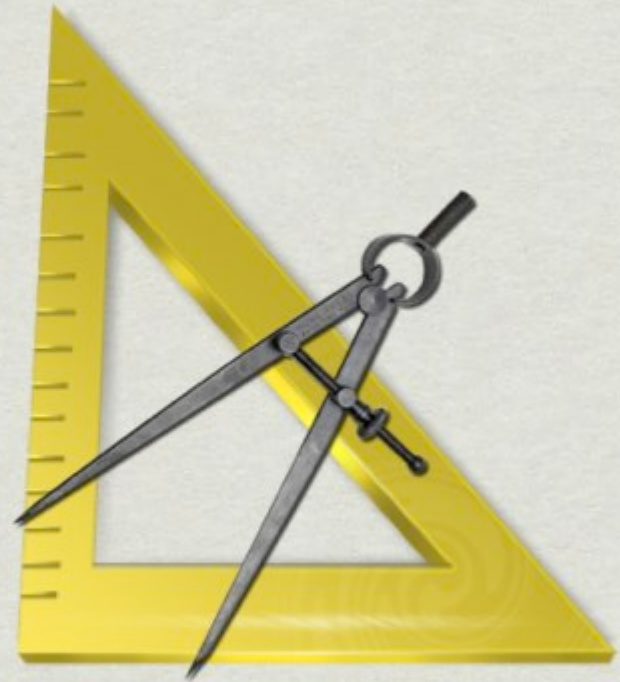* Engineering challenges

NLP Evaluation

Intrinsic: use a gold-standard dataset and some metric to measure the system's performance directly. (in-domain & out-of-domain)

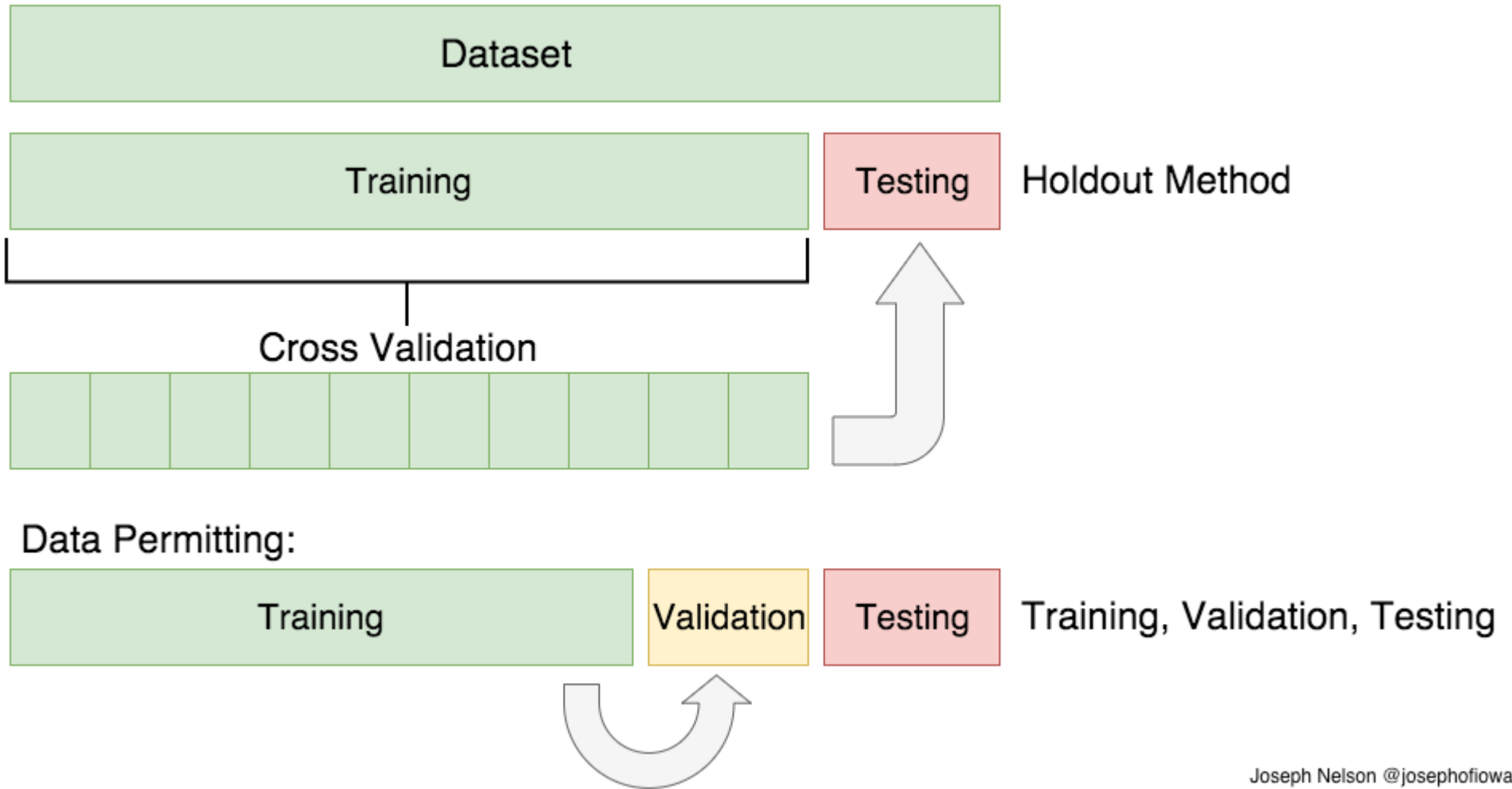**Extrinsic**: use a system as part of an upstream task(s) and measure performance change there.

Metrics

"IF YOU CAN'T MEASURE IT, YOU CAN'T MANAGE IT"

PETER DRUCKER

Dev-Test Split

f1 et al.

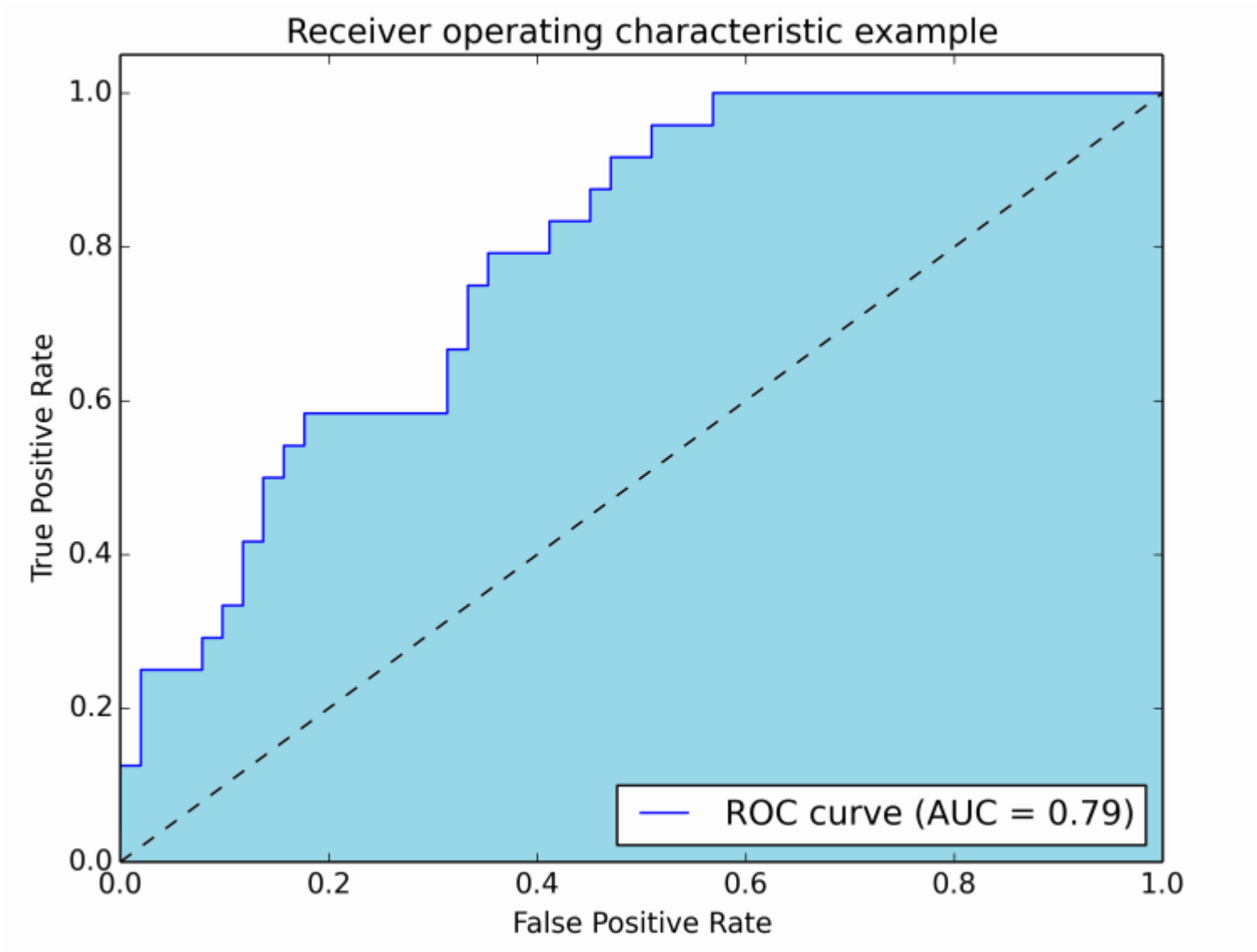|	Condition (as determined by "Gold standard")			
Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
Test outcome positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV, Precision) = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$
Test outcome negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$
Positive likelihood ratio (**LR+**) = TPR/FPR	True positive rate (TPR, Sensitivity, Recall) = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR, Fall-out) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	
Negative likelihood ratio (**LR−**) = FNR/TNR	False negative rate (FNR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR, Specificity, SPC) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$		
Diagnostic odds ratio (**DOR**) = LR+/LR−				

(left margin label: **Test outcome**)

f1 et al.

$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

$$G = \sqrt{\text{precision} \cdot \text{recall}}$$

ROC AUC

Receiver operating characteristic example

ROC curve (AUC = 0.79)

Confusion Matrix

Languages	English	German	French	Italian	Dutch	Spanish
English	**9244**	38	199	145	222	139
German	28	**9514**	67	29	325	27
French	20	52	**9525**	165	83	160
Italian	6	7	18	**9822**	16	134
Dutch	60	66	35	20	**9800**	19
Spanish	6	8	41	242	24	**9679**

```
AF: 1.00 |
DE: 1.00 |
EN: 1.00 |
ES: 0.94 | IT:0.06
NL: 0.85 | IT:0.03 CA:0.03 AF:0.03 DE:0.03 FR:0.03
RU: 0.82 | UK:0.12 EN:0.03 DA:0.02 FR:0.02
UK: 0.93 | TG:0.07
Total quality: 0.91
```

Evaluating "NLG"

1) Word Error Rate (WER)
2) BLEU, ROUGE, METEOR

BLEU Evaluation Metric

(Papineni et al, ACL-2002)

Reference (human) translation:
The U.S. island of Guam is maintaining a high state of alert after the Guam airport and its offices both received an e-mail from someone calling himself the Saudi Arabian Osama bin Laden and threatening a biological/ chemical attack against public places such as the airport .

Machine translation:
The American [?] international airport and its the office all receives one calls self the sand Arab rich business [?] and so on electronic mail , which sends out ; The threat will be able after public place and so on the airport to start the biochemistry attack , [?] highly alerts after the maintenance.

- BLEU is a weighted geometric mean, with a brevity penalty factor added.
  - Note that it's precision-oriented
- BLEU4 formula

  (counts n-grams up to length 4)

$$\exp(1.0 * \log p1 +$$
$$0.5 * \log p2 +$$
$$0.25 * \log p3 +$$
$$0.125 * \log p4 -$$
$$\max(\text{words-in-reference} / \text{words-in-machine} - 1, 0)$$

p1 = 1-gram precision
P2 = 2-gram precision
P3 = 3-gram precision
P4 = 4-gram precision

Note: only works at corpus level (zeroes kill it); there's a smoothed variant for sentence-level

Other Metrics

1) Cross-entropy
2) Perplexity
3) Custom: MaxMatch, Tribunal,…
4) Your own?

Baseline

Quality that can be achieved
using some reasonable
primitive approach
(on the same data).

2 ways to measure improvement:
* quality improvement
* error reduction

Recent SNLI Example

https://arxiv.org/pdf/1803.02324.pdf
- majority class baseline: 0.34
- SOTA models: 0.87

Recent SNLI Example

https://arxiv.org/pdf/1803.02324.pdf
- majority class baseline: 0.34
- SOTA models: 0.87
- basic fasttext classifier: 0.67

		Entailment		Neutral		Contradiction	
SNLI	outdoors	2.8%	tall	0.7%	nobody	0.1%
	least	0.2%	first	0.6%	sleeping	3.2%
	instrument	0.5%	competition	0.7%	no	1.2%
	outside	8.0%	sad	0.5%	tv	0.4%
	animal	0.7%	favorite	0.4%	cat	1.3%
MNLI	some	1.6%	also	1.4%	never	5.0%
	yes	0.1%	because	4.1%	no	7.6%
	something	0.9%	popular	0.7%	nothing	1.4%
	sometimes	0.2%	many	2.2%	any	4.1%
	various	0.1%	most	1.8%	none	0.1%

Table 4: Top 5 words by PMI(*word*, *class*), along with the proportion of *class* training samples containing *word*. MultiNLI is abbreviated to MNLI.
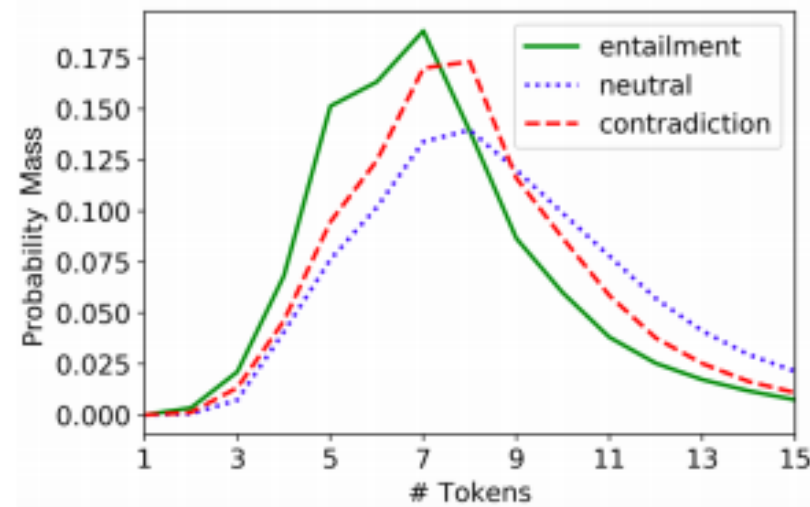
Figure 1: The probability mass function of the hypothesis length in SNLI, by class.

State-of-the-Art (SOTA)

The highest publicly known result on a well-established dataset.

https://aclweb.org/aclwiki/State_of_the_art

GLUE Benchmark

The General Language Understanding Evaluation (GLUE) benchmark is a collection of resources for training, evaluating, and analyzing natural language understanding systems. GLUE consists of:
* A benchmark of nine sentence- or sentence-pair language understanding tasks built on established existing datasets and selected to cover a diverse range of dataset sizes, text genres, and degrees of difficulty
* A diagnostic dataset designed to evaluate and analyze model performance with respect to a wide range of linguistic phenomena found in natural language, and
* A public leaderboard for tracking performance on the benchmark and a dashboard for visualizing the performance of models on the diagnostic set.

Evolution of NLProc Paradigms

1. Symbolical/rule-based
   "The (Weak) Empire of Reason"
2. Counting-based
   "The Empiricist Invasion"
3. Deep Learning-based
   "The Revenge of the Spherical Cows"

http://www.earningmyturns.org/2017/06/a-computational-linguistic-farce-in.html

Rule-based Approach
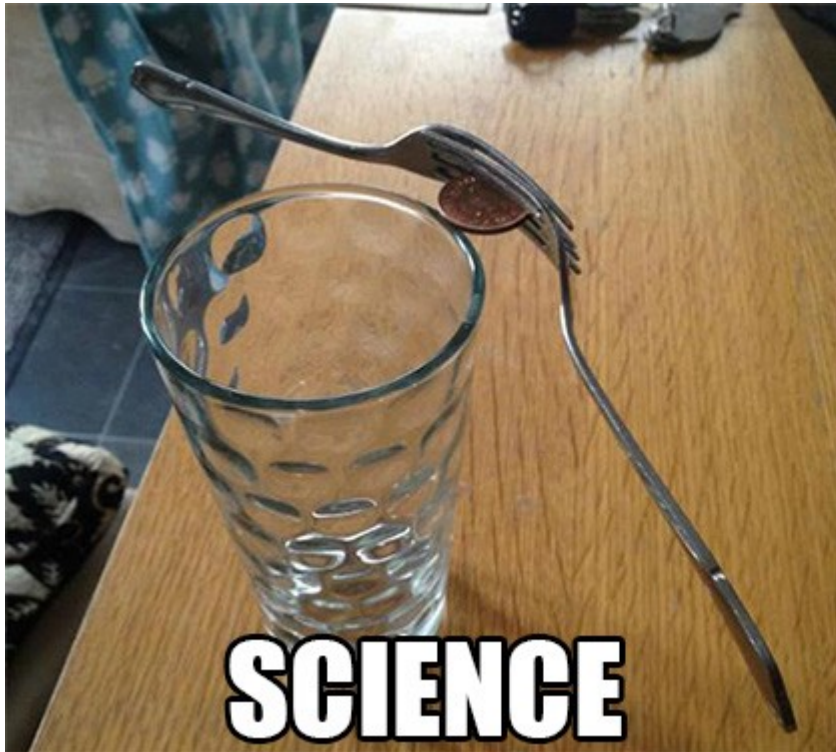
English Stemming
(rule-based definition)

Porter stemmer:

```
(defun stem (word)
  (if (<= (length word) 2)
      word
      (-> word step1ab step1c step2 step3 step4 step5)))


step1ab: (-> word stem-s stem-ed/ing)
stem-s:  (when (and (ends-with "s" word)
                    (not (ends-with "ss" word)))
           (substr word (if (or (ends-with "sses" word)
                                (ends-with "ies" word))
                            -2 -1)))
step1c:  (when (and (ends-with "y" word)
                    (vowel-in-stem? word 1))
           (:= (last-char word) #\i))
...
```
https://github.com/vseloved/cl-nlp/blob/master/src/lexics/porter.lisp

English Tokenization
(small problem)

* Simplest regex:
  [^\s]+
* More advanced regex:
  \w+|[!"#$%&'*+,\./:;<=>?@^`~…\(\)〔〕{}\[\|\]——————«»""''–]
* Even more advanced regex:
  [+-]?[0-9](?:[0-9,\.]*[0-9])?
  |[\w@](?:[\w''`@-][\w']|[\w'][\w@''`-])*[\w']?
  |["#$%&*+,/:;<=>@^`~…\(\)〔〕{}\[\|\]——————«»""''']
  |[\.!?]+
  |-+

Add post-processing:
* concatenate abbreviations and decimals
* split contractions with regexes
  - 2-character abbreviations regex:
    i['''`]m|(?:s?he|it)['''`]s|(?:i|you|s?he|we|they)['''`]d$
  - 3-character abbreviations regex:
    (?:i|you|s?he|we|they)['''`](?:ll|[vr]e)|n['''`]t$

https://github.com/lang-uk/tokenize-uk/blob/master/tokenize_uk/tokenize_uk.py

Error Correction
(inherently rule-based)

LanguageTool:

```xml
<category name="Стиль" id="STYLE" type="style">
    <rulegroup id="BILSH_WITH_ADJ" name="Більш з прикметниками">
      <rule>
        <pattern>
          <token>більш</token>
          <token postag_regexp="yes" postag="ad[jv]:.*compc.*">
            <exception>менш</exception>
          </token>
        </pattern>
        <message>Після «більш» не може стояти вища форма прикметника</message>
        <!-- TODO: when we can bind comparative forms togher again
            <suggestion><match no="2"/></suggestion>
            <suggestion><match no="1"/> <match no="2" postag_regexp="yes"
postag="(.*):compc(.*)" postag_replace="$1:compb$2"/></suggestion>
            <example correction="Світліший|Більш світлий"><marker>Більш
світліший</marker>.</example>
        -->
        <example correction=""><marker>Більш світліший</marker>.</example>
        <example>все закінчилось більш менш</example>
      </rule>
```

...
https://github.com/languagetool-org/languagetool/blob/master/languagetool-language-modules/uk/src/main/resources/org/languagetool/rules/uk/

Fighting Spam
(just a bad idea :)

SpamAssasin:

```
# bodyn
# example: postacie greet! Chcesz porozmawiac  Co prawda tu rzadko bywam,
zwykle pisze tu -  http://hanna.3xa.info
uri        __LOCAL_LINK_INFO   /http:\/\/\w{3,8}\.\w\w\w\.info/
header     __LOCAL_FROM_O2     From =~ /\@o2\.pl/
meta       LOCAL_LINK_INFO     __LOCAL_LINK_INFO && __LOCAL_FROM_O2
describe   LOCAL_LINK_INFO     Link postaci http://cos.cos.info i z o2.pl
score      LOCAL_LINK_INFO     5
```

https://wiki.apache.org/spamassassin/CustomRulesets

(…which continues being reimplemented)

Facebook antispam (using HAXL):

```
fpSpammer :: Haxl Bool
fpSpammer =
  talkingAboutFP .&&
  numFriends .> 100 .&&
  friendsLikeCPlusPlus
 where
  talkingAboutFP =
    strContains "Functional Programming" <$> postContent

friendsLikeCPlusPlus = do
  friends <- getFriends
  cppFriends <- filterM likesCPlusPlus friends
  return (length cppFriends >= length friends `div` 2)
```

http://multicore.doc.ic.ac.uk/iPr0gram/slides/2015-2016/Marlow-fighting-spam.pdf

https://petrimazepa.com/m_li_ili_kak_algoritm_facebook_blokiruet_polzovatelei_na_osnovanii_stop_slov

Dialog Systems
(inherently scripted)

ELIZA:

```
(defparameter *eliza-rules*
 '((((?* ?x) hello (?* ?y))
    (How do you do.  Please state your problem.))
   (((?* ?x) I want (?* ?y))
    (What would it mean if you got ?y)
    (Why do you want ?y) (Suppose you got ?y soon))
   (((?* ?x) if (?* ?y))
    (Do you really think its likely that ?y) (Do you wish that ?y)
    (What do you think about ?y) (Really-- if ?y))
   (((?* ?x) no (?* ?y))
    (Why not?) (You are being a bit negative)
    (Are you saying "NO" just to be negative?))
   (((?* ?x) I was (?* ?y))
    (Were you really?) (Perhaps I already knew you were ?y)
    (Why do you tell me you were ?y now?))
   (((?* ?x) I feel (?* ?y))
    (Do you often feel ?y ?))
   (((?* ?x) I felt (?* ?y))
    (What other feelings do you have?))))
```
https://norvig.com/paip/eliza1.lisp

…and if you thought rules are dead

Wit.ai:

To make a bot, there are two schools: rules or machine learning. (Everybody claims rules are bad and their bot is powered by AI, but when you really look under the hood, the core is often imperative.)

Machine learning is of course more desirable, but the problem is the training dataset. Training a Wit intent with a dozen examples works well, and it's easy to leverage the community to get more examples. But in order to entirely learn the business logic of a bot of medium complexity, we would need many, many thousands of example conversations.

Rules (or any kind of imperative approach, including plain script/program) are kind of the opposite. The good thing with rules is, you can have a demo working after you write two rules. As long as you follow the script carefully, your bot will work and your audience will be impressed. But as you discover new "paths" in the dialog, you'll add more and more rules, until one day everything collapses. You're doomed by the curse of combinatorics. Any new rule conflicts with old rules that you totally forgot the reason for. Your bot cannot improve anymore.

When you create your bot, you just start with a few stories that describe the most probable conversations paths. At this stage, Bot Engine will build a machine learning model that deliberately overfits the stories dataset. Practically, it means that stories will behave almost like rules.

https://medium.com/wit-ai/bot-engine-26af22d37fd6

wit.ai

🔒 l5t / MovieTicket1 / ●   💬 Inbox   🖼 Stories   ⚡ Actions   ✳ Understanding   ▭ Logs   ⚙ Settings

⊕ Create story

✕   All changes saved ● 🗑

👤 I want to go see Straight Outta Compton

     movie_name   [Straight Outta Compton ▼]

     ⊕ Add a new entity

say (Where would you like to go watch {movie_name}?) 💬🤖

⊕ Add variable

👤 ACME theatre in NY

     wit/location   ACME theatre in NY

     ⊕ Add a new entity

say (What time and how many tickets?) 🤖

⊕ Add variable

👤 4 tickets today at 7pm please

     wit/datetime   4/8/2016, 7:00:00 PM

     wit/number   4

     ⊕ Add a new entity

book_tickets (context, entities) ⚡🤖

Doesn't produce context

⊕ Add context field

say (You're all set!) 💬🤖

⊕ Add variable

👤 User says...   💬🤖 Bot says...   ⚡🤖 Bot executes...

More Rule-based Examples

* Rule-based MT (Systran)
* Various NLG systems
* Custom parsing (Sparser)

Event Detection at Ravenpack

A Template Trie

Rete Algorithm

Efficient matching of many templates
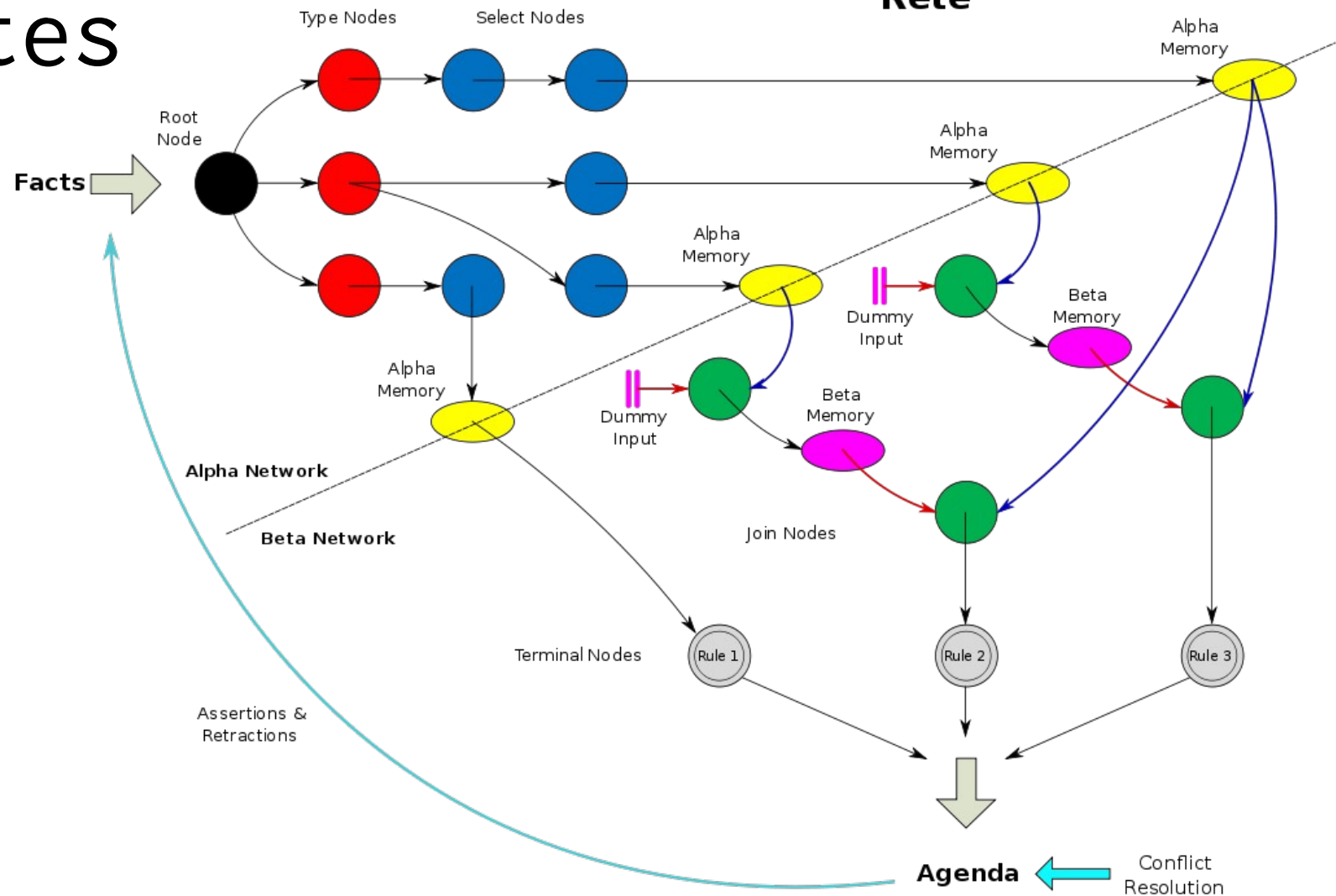
Rete

Type Nodes   Select Nodes

Alpha Memory

Root Node

Facts

Alpha Memory

Alpha Memory

Dummy Input

Beta Memory

Alpha Memory

Dummy Input

Beta Memory

Alpha Network

Beta Network

Join Nodes

Terminal Nodes   Rule 1   Rule 2   Rule 3

Assertions & Retractions

Agenda   Conflict Resolution
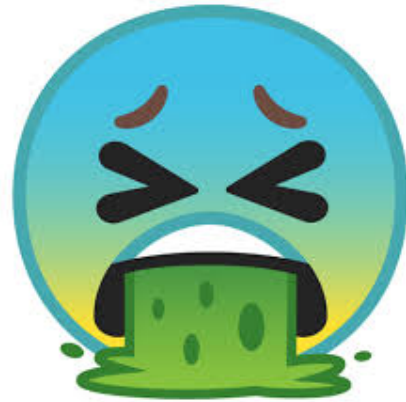
Languages for Rules

* regexes

Languages for Rules

* regexes

Languages for Rules

* regexes
* XML, JSON

Languages for Rules

* regexes
* XML, JSON
* external DSLs (example: JBOSS drules)

Listing 5.7   A simple set of rules for email spam filtering

```
package demo;
import iweb2.ch5.classification.data.Email;
import iweb2.ch5.classification.rules.ClassificationResult;

global ClassificationResult classificationResult;

rule "Tests for viagra in subject"
when
    Email( $s : subject )
    eval( classificationResult.isSimilar($s, "viagra" ) )
then
    classificationResult.setSpamEmail(true);
end

rule "Tests for 'drugs' in subject"
when
  Email( $s : subject )
    eval( classificationResult.isSimilar($s, "drugs" ) )
then
    classificationResult.setSpamEmail(true);
end
```

Rule for identifying "Viagra" in email subject

Rule for identifying "drugs" in email subject

Languages for Rules

* regexes
* XML, JSON
* external DSLs
* internal DSLs

```
(match-html source
           '(>> article
             (aside (>> a ($ user))
                   (>> li (strong "Native Tongue:") ($ lang)))
             (div |...| (>> (div :data-role "commentContent"
                             ($ text) (span) |...|))
             !!!))
```

https://github.com/vseloved/crawlik

Rules Pros&Cons

+ compact & fast
+ full control
+ arbitrary recall
+ iterative
+ best interpetability
+ perfectly accommodate domain experts
- precision ceiling
- non-optimal weights
- require a lot of human labor
- hard to interpret/calibrate score

Hybrid Approach

"Rule-based" framework that incorporates Machine Learning models.

+ control
+ best of both worlds
- more complex
- not sexy

Counteracting Bias

* Rule-based post-processing
* Mix multiple datasets
  + special feature-selection
* Train multiple models on different datasets and create an ensemble
* Use domain adaptation techniques

Productization

Delivering your NLP application to the users
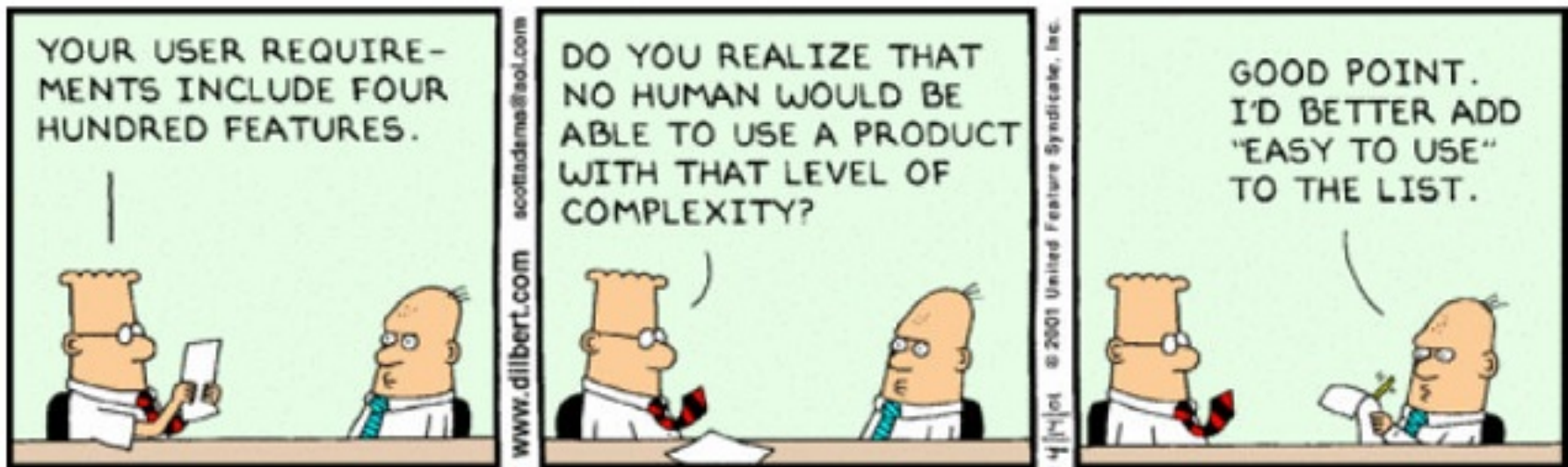
Product variants

* end-user product
  (web, mobile, desktop)
* internal feature
* API
* library
* scientific paper

Requirements

* speed (processing, startup)
* memory usage
* storage
* interoperability
* ease of use
* ease of update

Storage Optimization

WILD Initial model size ~ 1G
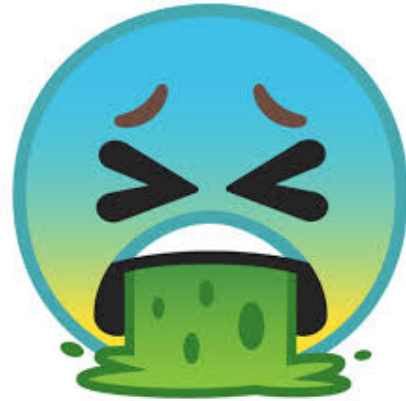(compared to megabytes for CLD,
 target: 10-20 MB)

Ways to reduce:
  - partly rule-based
  - model pruning
  - clever algorithms: perfect hash
    tables, Huffman coding
  - model compression
    https://arxiv.org/abs/1612.03651

Model Formats

* pickle & co.

Model Formats

* pickle & co.
* JSON, ProtoBuf, ...

Model Formats

* pickle & co.
* JSON, ProtoBuf, ...
* HDF5

Model Formats

* pickle & co.
* JSON, ProtoBuf, ...
* HDF5
* Custom gzipped text-based

Adaptation
(manual vs automatic)

* for RB systems: add more rules :D
* for ML systems: online learning algorithms
* for DL systems: transfer Learning

* Lambda architecture

The Pipeline

"Getting a data pipeline into production for the first time entails dealing with many different moving parts — in these cases, spend more time thinking about the pipeline itself, and how it will enable experimentation, and less about its first algorithm."

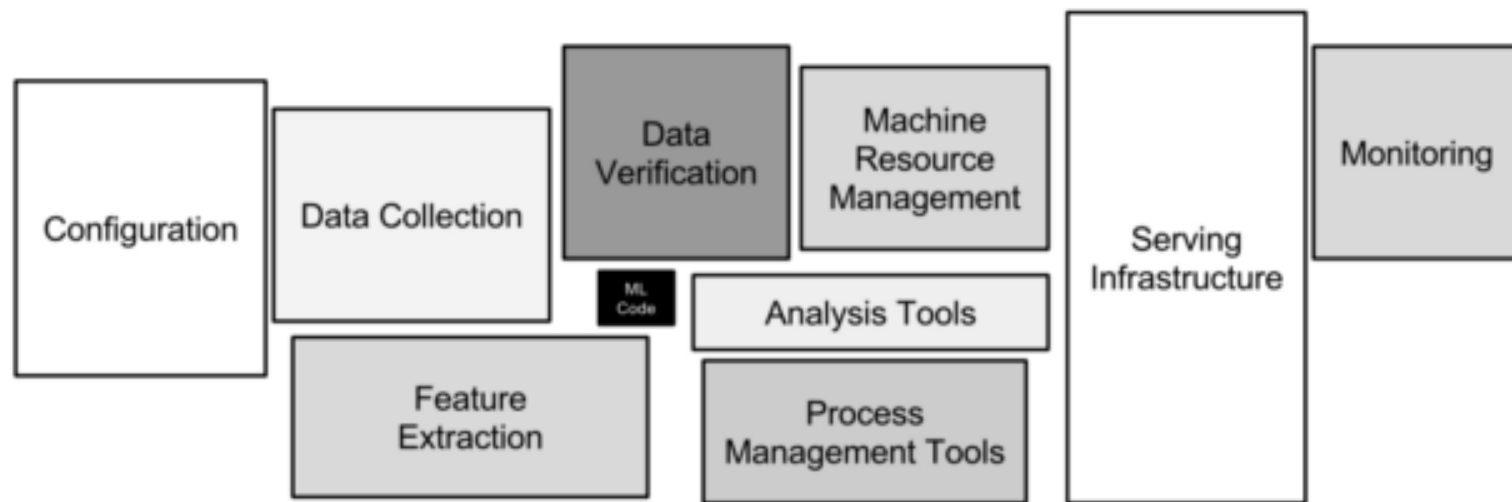https://medium.com/@neal_lathia/five-lessons-from -building-machine-learning-systems-d703162846ad

Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.
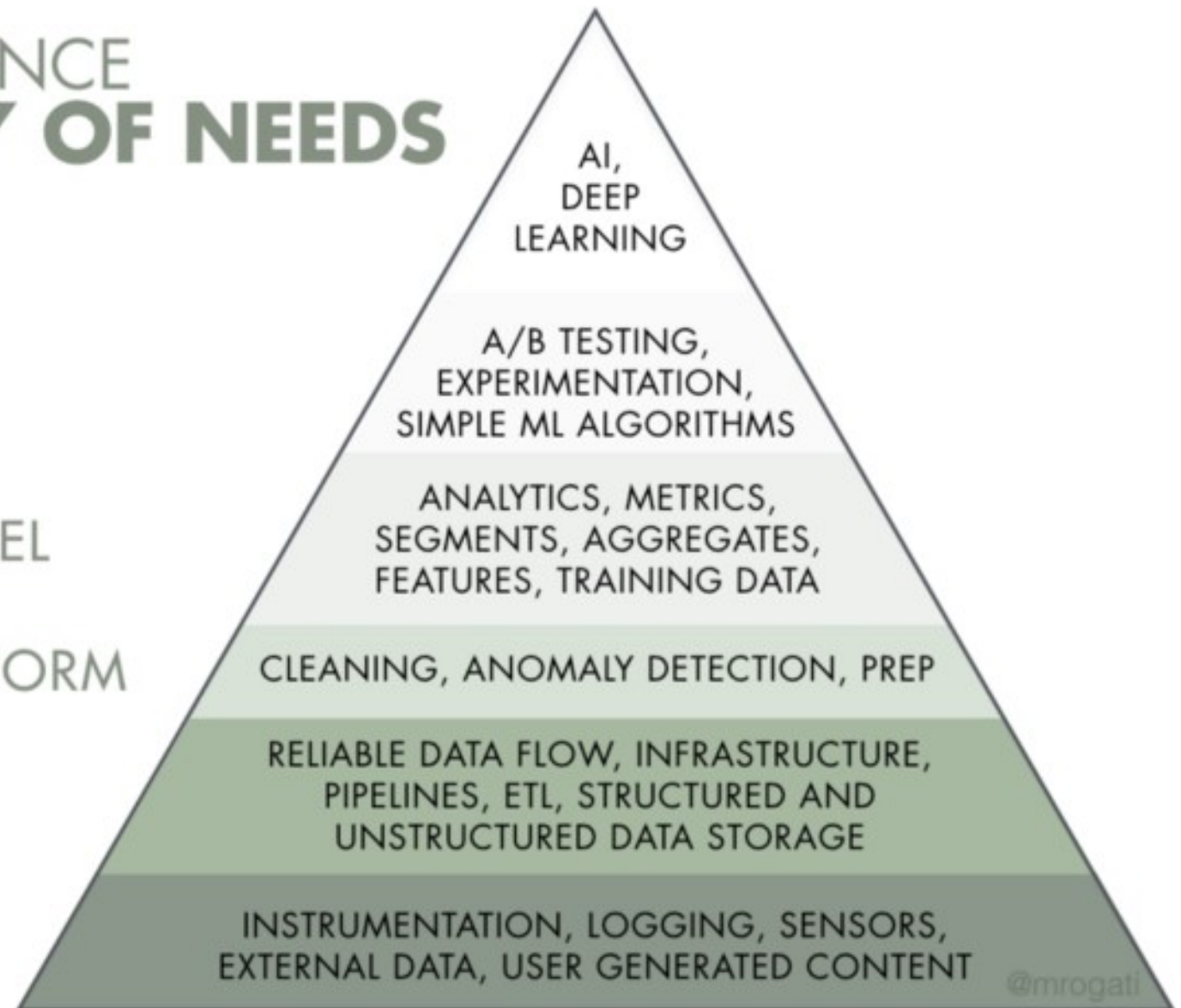
THE DATA SCIENCE
HIERARCHY OF NEEDS

LEARN/OPTIMIZE

AGGREGATE/LABEL

EXPLORE/TRANSFORM

MOVE/STORE

COLLECT

AI, DEEP LEARNING

A/B TESTING, EXPERIMENTATION, SIMPLE ML ALGORITHMS

ANALYTICS, METRICS, SEGMENTS, AGGREGATES, FEATURES, TRAINING DATA

CLEANING, ANOMALY DETECTION, PREP

RELIABLE DATA FLOW, INFRASTRUCTURE, PIPELINES, ETL, STRUCTURED AND UNSTRUCTURED DATA STORAGE

INSTRUMENTATION, LOGGING, SENSORS, EXTERNAL DATA, USER GENERATED CONTENT

@mrogati

https://hackernoon.com/the-ai-hierarchy-of-needs-18f111fcc007

Read More

- http://people.cs.umass.edu/~brenocon/inlp2015/15-eval.pdf
- https://ehudreiter.com/2017/05/03/metrics-nlg-evaluation/
- https://nlpers.blogspot.com/2014/11/the-myth-of-strong-baseline.html
- http://kavita-ganesan.com/what-is-rouge-and-how-it-works-for-evaluation-of-summaries
- https://www.slideshare.net/frandzi/native-language-identification-brief-review-to-the-state-of-the-art
- https://www.slideshare.net/grammarly/grammarly-ainlp-club-1-domain-and-social-bias-in-nlp-case-study-in-language-identification-tim-baldwin-80252288