University of Zurich^UZH

Institute of Computational Linguistics

A method for in-depth comparative evaluation

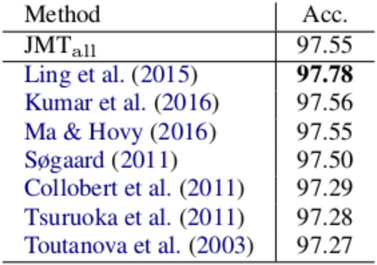
How (dis)similar are outputs of POS taggers, dependency parsers and coreference resolvers really?

Don Tuggener `tuggener@cl.uzh.ch`

April 21, 2017

Motivation

Top systems often result in very **similar scores** despite (potentially vastly) **different architectures**



Method	Acc.
JMT_{all}	97.55
Ling et al. (2015)	**97.78**
Kumar et al. (2016)	97.56
Ma & Hovy (2016)	97.55
Søgaard (2011)	97.50
Collobert et al. (2011)	97.29
Tsuruoka et al. (2011)	97.28
Toutanova et al. (2003)	97.27

Hashimoto et al. (2017): POS tagging WSJ

Question & Goal

Are these systems roughly producing the **same output**?

- The top system is just **generally (a bit) better**? Or . . .
- **Some systems** have an **area of specialty** where they outperform the others (despite overall lower score)?
- Overall Accuracy / F1 doesn’t tell us
- Generally, **little is known/done** about this

Devise evaluation method that

- **compares two outputs** (or more) to a gold standard
- **highlights** and **quantifies** their **specific differences**

Question & Goal

Are these systems roughly producing the **same output**?

- The top system is just **generally (a bit) better**? Or ...
- **Some systems** have an **area of specialty** where they outperform the others (despite overall lower score)?
- Overall Accuracy / F1 doesn't tell us
- Generally, **little is known/done** about this

Devise evaluation method that

- **compares two outputs** (or more) to a gold standard
- **highlights** and **quantifies** their **specific differences**

Simple Metric

T: tokens in the test set, S_1, S_2: system outputs

$$diff(S_1, S_2 \mid T) = \frac{|\forall t_i \in T : label(t_i, S_1) \neq label(t_i, S_2)|}{|T|}$$

- Isn't this just $1 - Accuracy$? Yes!
- Does this tell us whether S_1 or S_2 is correct where they're different? No!
- Include the gold standard

Difference classes

Introduce a **inventory of classes** to label **differences** in S_1 and S_2 given the gold standard on the **token level**.
(Let's assume e.g. S_1: baseline, S_2: new SOTA)

Gold	$\mathbf{S_1}$	$\mathbf{S_2}$	Class
A	B	A	*Correction*
A	A	B	*New error*
A	B	C	*Changed error*

Analyse **distribution** of these classes

Dependency parsing WSJ

Stanford PCFG/NN, Parsey McParseface

	UAS	**LS**	**LAS**
Stanford PCFG	87.96	92.26	85.36
Stanford NN	88.68	92.45	86.43
Parsey	**92.70**	**92.86**	**88.94**

	Δ LAS	$\mathit{diff}\,(S_1 \neq S_2)$
Stanford PCFG ↔ Stanford NN	**1.07**	**14.01**
Stanford NN ↔ Parsey	2.51	13.62
Parsey ↔ Stanford PCFG	3.58	15.49

⇒ diff does not seem to correlate with ΔLAS

Dependency parsing WSJ: Class distribution

- "Only" **half** the differences are **corrections**
- Most frequent corrections wrt. **attachment**
- New errors often involve labeling
- Changed errors are mixed

Stanford NN → Parsey	
Corrections:	**50.22**
nn → nn	10.93
prep → prep	9.49
cc → cc	5.32
New errors:	**31.79**
vmod → partmod	9.38
amod → nn	8.08
prep → prep	7.38
Changed errors:	**17.99**
prep → prep → prep	5.00
vmod → vmod → partmod	2.95

Another view on difference: Oracle ensemble

- Take gold standard and n system outputs
- Whenever **at least 1** of n systems has the **correct** label, count as correct (**oracle** ensemble)
- Measure **oracle** score vs. **best** performing **single system**
- **Upper bound** for **ensemble**
- **Indicator** for how **complementary** (or **different**) the n sytem outputs are

Oracle ensemble POS tagging TüebaD/Z

Stanford POS, TreeTagger, CleverTagger

	Stan.	**Tree.**	**Clever.**	**Upper bound**	
Overall	90.41	94.38	96.16	**98.52**	**+2.36**
NE	**87.35**	*77.46*	85.31	95.17	**+9.86**
ADV	89.25	91.71	90.93	95.48	**+4.55**
VVFIN	*79.73*	**95.15**	91.52	97.48	**+5.96**
ADJD	*72.37*	89.29	88.80	93.53	**+4.73**

- **Best tagger** overall is **outperformed** by a large margin for **particular tags** (e.g. VVFIN)
- Vast differences in performances wrt. different tags (Stan.)
- Oracle performance near optimal

Oracle ensemble dependency parsing WSJ: LAS

	S-PCFG	**S-NN**	**P-MP**	**Upper bound**	
Overall	85.36	86.43	88.94	**94.93**	**+5.99**
nsubj	92.08	*89.78*	94.41	97.85	**+3.44**
amod	87.59	88.45	86.95	95.26	**+8.31**
root	93.79	*89.61*	95.74	98.63	+2.89
dobj	90.19	90.88	92.91	97.47	**+4.56**
advmod	74.48	78.56	82.97	91.40	**+8.43**

- Parsey consistently best (ex. amod; adjectival modifier)
- Large distance to upper bound ($\sim$ 6 LAS)

Conclusion

- **Method** and **class inventory** for **comparative evaluation**
- **SOTA outputs** are more **heterogeneous** than (small) differences in Acc. suggest
- Most **advances** come at the **cost of new errors**
  - Quantifiable with the proposed method
- Why does my **feature X** not improve the baseline?
  - Maybe it does in the intended subproblem, but also harms performance in other areas
  - Now you can find out
- A means to help you point out **in what regard your system output differs from others** - even if it's not the new SOTA, maybe it solves a (sub-)problem the SOTA can't!