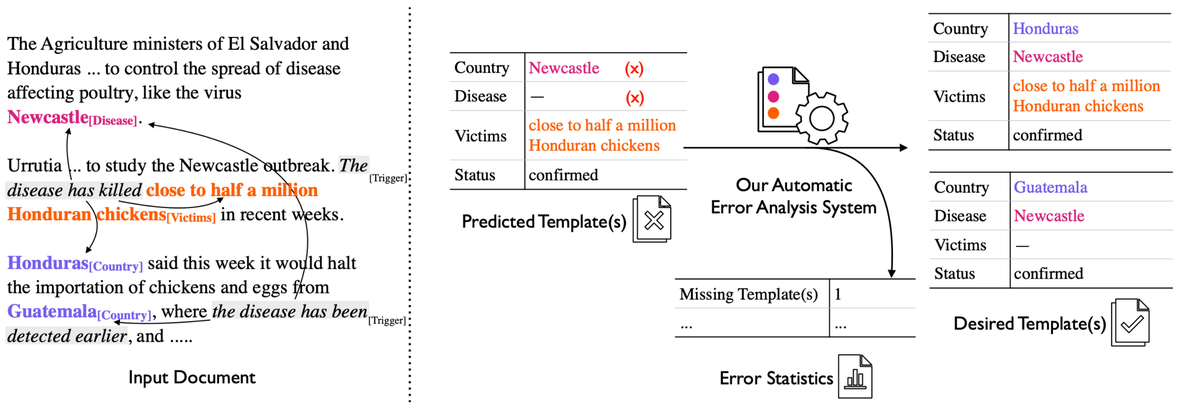
Automated Error Analysis For Document-Level Information Extraction

Cornell Bowers CIS College of Computing and Information Science

Aliva Das*, Xinya Du*, Barry Wang*, Kejian Shi, Jiayuan Gu, Thomas Porter, Claire Cardie

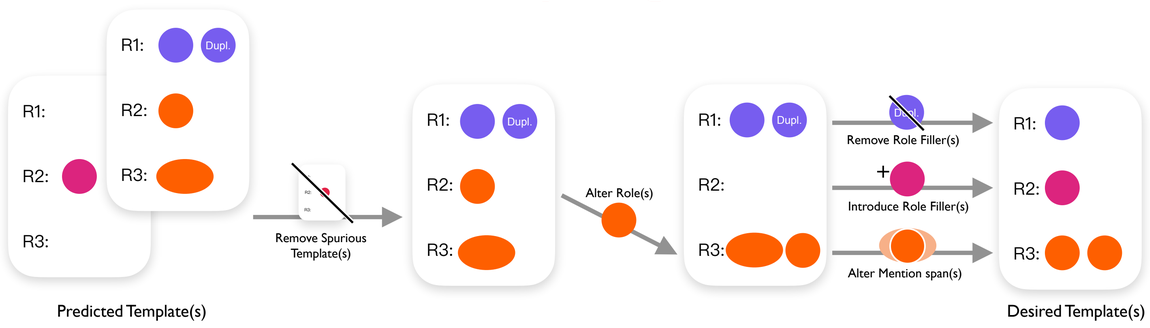
*Co-first Author

Document-Level Information Extraction



- Document-level information extraction models is a task to extract "templates" from given texts and are compared with gold results.
- Standard evaluation scores (e.g. F1) do not reflect error type information.
- Our system compares outputs and gold results to categorize all their errors, which draws more insights into model performance.

Error Analysis Algorithm



- With gold results and model outputs, our system uses transformations to convert model outputs to desired templates.
- The system generates error information based on the transformations used.
- In this output, a model outputs two template for a document while the gold/desired template only has one.
  - The first step is to remove the extra template, and apparently the one of the template is closer to the desired result than the other. Hence a transformation removes the leftmost template.
  - Later steps try to give partial credits as much as possible (for example by changing the assigned role), before directly remove/span alter/add transformations to the role fillers.

Errors Types

	Error Type	Error Component: Mis-placement	Error Component: Span Error	Error Name	Transformations(s)	Predicted	Gold
i)	Within Template		■	Span Error	Alter Span	PerInd: [members]	PerpInd: [members of the maoist terrorist organization shining path]
ii)				Duplicate Role Filler	Remove Role Filler	Target: [electrical appliance store], [store]	Target: [electrical appliance store, store]
iii)			■	Duplicate Partially Matched Role Filler	Alter Span + Remove Role Filler	Target: [store], [electrical]	Target: [store, electrical appliance store]
iv)		■ Spurious		Spurious Role Filler	Remove Role Filler	PerpOrg: [fmln] Victim: [rosa imelda gonzalez medrano]	PerpOrg: — Victim: [rosa imelda gonzalez medrano]
v)		□ (Missing)		Missing Role Filler	Introduce Role Filler	Target: —	Target: [local garrison, garrison]
vi)		■ Role Error		Incorrect Role	Alter Role	PerpInd: — Victim: [gonzalo rodriguez gacha]	PerpInd: [gonzalo rodriguez gacha] Victim: —
vii)			■	Incorrect Role + Partially Matched Filler	Alter Span + Alter Role	PerpInd: — Victim: [gonzalo rodriguez]	PerpInd: [gonzalo rodriguez gacha] Victim: —
viii)		■ Template Error		Wrong Template Role Filler	Remove Cross Template Spurious Role Filler	T1:Target: [public bus] T2:Target: —	T1:Target: — T2:Target: [public bus, bus]
ix)			■	Wrong Template For Partially Matched Role Filler	Alter Span + Remove Cross Template Spurious Role Filler	T1:Target: [public] T2:Target: —	T1:Target: — T2:Target: [public bus, bus]
x)	Within and Cross Template	■■ Role + Template Error		Wrong Template + Wrong Role	Alter Role + Remove Cross Template Spurious Role Filler	T1: Victim: — Weapon: — T2: Victim: [adolfo spezua] Weapon: [thomas pellisier]	T1: Victim: [thomas pellisier] Weapon: — T2: Victim: [adolfo spezua] Weapon: —
xi)			■	Wrong Template + Wrong Role + Partially Matched Filler	Alter Span + Alter Role + Remove Cross Template Spurious Role Filler	T1: Victim: — Weapon: — T2: Victim: [adolfo spezua] Weapon: [thomas]	T1: Victim: [thomas pellisier] Weapon: — T2: Victim: [adolfo spezua] Weapon: —
xii)	Template Detection	■ Spurious		Spurious Template	Remove Template	T1: PerpOrg: [fmln]	—
xiii)		□ (Missing)		Missing Template	Introduce Template	—	T1: PerpOrg: [fmln]

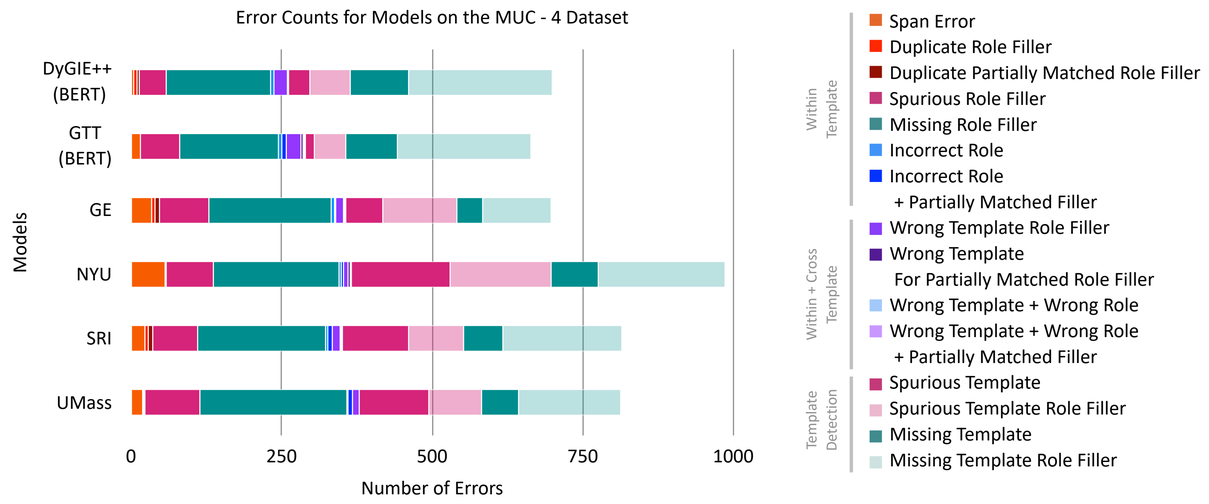
- We present all error error type we generate based on the transformations, with examples from MUC4.
- For each template, in every role, the role fillers in brackets refer to the same entity (i.e., these role fillers are coreferent mentions), while role fillers in different brackets refer to different entities.
- The underlined text indicates the error in the predicted output.

Datasets

	# docs (train/val/test/unannot.)	# tokens per doc (min/max/avg.)	# templates per relevant doc (max/avg.)	% docs with 0 templates
# MUC-4 (MUC-4, 1992)	1300 / 200 / 200 / 0	31 / 1695 / 362	14 / 1.61	44.59
ProMed[1]	125 / 12 / 108 / 4979	57 / 4417 / 621	9 / 1.55	19.83
# SciREX (Jain et al., 2020)	304 / 66 / 66 / 0	1153 / 13155 / 5401	16 / 2.28	0.00

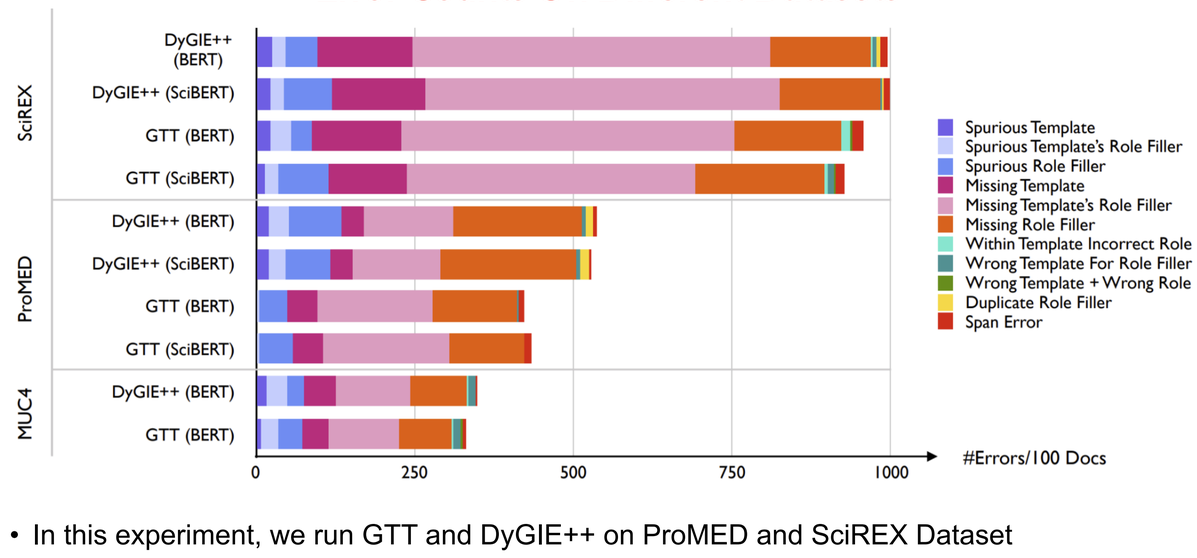
- Roughly 2000 document from MUC4, SciREX, and ProMed.
  - MUC4: Newswire covering Latin American terrorist incidents
  - SciREX: Annotated ML articles from Papers with Code
  - ProMED: News-style infectious disease report
- Across all datasets, every example has ~1.7 gold templates.

Result: Old v.s. New: Error Counts For Different Models on MUC



- NIST's MUC4 dataset in 1992 contains both the corpora and back-then SotA model outputs.
- We compare their results with model models like GTT (Template Filling with Generative Transformers) and DyGIE++.
- We discover that:
  - Recent systems have much fewer **Span Error** error, hence higher Precision.
  - 1990s systems have relatively fewer **Missing Role Filler** and **Missing Template** errors, which contribute to higher Recall.
  - 1990s systems actually can obtain higher F1 score due to a better balance of Recall and Precision, as both systems suffer mostly on **Missing Role Filler** and **Missing Template** still.

Result: Scientific v.s. News: Error Counts On Different Datasets



- In this experiment, we run GTT and DyGIE++ on ProMED and SciREX Dataset
- We see an increase in F1 scores for all SciBERT-based models when compared to their BERT counterparts for the SciREX dataset.
  - Except GTT. GTT (SciBERT) has more **Missing Template** errors than GTT (BERT)
  - Can be explained as GTT (BERT) is better at detecting events.
- DyGIE++ is worse at coreference resolution when compared to GTT.
  - As DyGIE++ makes more Duplicate Role Filler errors across all datasets.
- The major source of error for both GTT and DyGIE++ across all the datasets is missing recall in the form of Missing Role Filler and Missing Template errors.

Take Away

- State-of-the-art models perform better at span extraction but worse at template detection and role assignment.
- With a better balance between precision and recall, the best early models still outperform the relatively high precision, low-recall modern models.

- Missing role fillers remain the main source of errors (or generally, Recall).
- Scientific corpora are the most challenging datasets for all systems

Improvements in these areas should be a priority for future system development.