Welcome to the OPERA
AIDA in 2019 ... a challenge

No more training data, only examples that illustrate the evaluation

Increasingly data-intensive neural learners

What do we do???
A range of responses...

• Just make machine learning work! 1
• Learning, augmented with external data 2
• Half-half 3
• Include (some) learning but only if it’s easy 4
• Forget machine learning! 5
Overview

1. System overview
2. TA1 English entity and relation processing
3. TA1 Rus/Ukr entity and event processing
4. TA1/2 KB construction and validation
5. TA3 Hypotheses
SYSTEM OVERVIEW
OPERA architecture

- TA1 extraction engines
- TA2 coref engine
- TA3 hypothesis formation

Input:
- text
- speech
- images
- video

Outputs:
- Ontology
- CSR
- PowerLoom database
OPERA framework

Input

Speech

Text

Images

English entities

Rus/Ukr entities

English events

Rus/Ukr events

Mini-KB creation/AIF validation

TA1 Mini-KB

Coref

Mini-KB creation/AIF validation

Mini-KB creation/AIF validation

TA2 Mini-KB

Belief Graph construction

Hypothesis formation

Queries

English events

Rus/Ukr events
TA1 framework

Input

Domain filter & language detection

English pipeline

- English entity detection
  - English entity linking
  - Eng entity relations

- English event detection
  - English argument detection
  - English coref

- Event frame assembly I

Speech pipeline

- English event detection

Ru/Uk pipeline

- MT: Ru/Uk → Eng
  - Ru/Uk Entity and Event detection

- Event frame assembly II

Image pipeline

- Entity detection
  - Person and Geo ID

CSR Combination

Mini-KB creation/AIF validation
OPERA TA2 + TA3 framework

- TA1 Mini-KB
  - Coref
  - Mini-KB creation/AIF validation

- TA2 Mini-KB
  - Belief Graph construction
  - Hypothesis formation
  - Mini-KB creation/AIF validation

- Query input
KBs and notations

- All results written in OPERA-internal frame notation (json) and stored in CSR (BlazeGraph)
- Input / output converters from/to AIDA AIF

- Two separate KB creation and validation procedures, for two parallel KBs (gives insurance, coverage, and backup):
  - Chalupsky: uses PowerLoom and Chameleon reasoner
  - Chaudhary: uses specialized rules
Internal dryruns

- Internal dry run mini-evals using the practice annotations released by LDC
- Evaluated results manually

- Results look promising, BUT ... hard to calculate P/R/F1 for various parts of the TA1 pipeline because LDC does not label all mentions of events, relations and entities, just the “salient” or “informative” ones (so we have to judge them ourselves ... laborious and not guaranteed)
TA1 TEXT:
ENGLISH ENTITIES AND RELATIONS
OPERA TA1 framework

Input

Domain filter & language detection

English pipeline

Speech pipeline

Ru/Uk pipeline

Image pipeline

MT: Ru/Uk → Eng

Ru/Uk Entity and Event detection

Event frame assembly II

CSR Combination

Mini-KB creation/AIF validation

English entity detection

English event detection

English argument detection

English coref

Eng entity relations

Event frame assembly I

English entity linking

English event detection

English argument detection

Eng entity relations

Event frame assembly I
1. Entity detection: Type-based NER data

• Multi-level learning:
  – Train separate detectors for type, subtype, and subsubtype-level type classification
  – Addresses data imbalance
  – May introduce layer-inconsistent types!

• Type-level from LDC ontology:
  – Training data: KBP NER data and a small amount of self-annotated data

• Sub(sub)type-level:
  – Training data: YAGO knowledge base (350k+ entity types) obtained from Heng Ji — thanks!
2. Entity linking

• Task: Given NER output mentions, link them to the reference KB

• Challenges: Over-large KB, noisy Geonames
  – Preprocess KB: Remove duplicated and unimportant entries (i.e., not located in Russia or Ukraine, or no Wikipedia page)

• Approach, given an entity:
  – Use Lucene to find all candidates in KB
  – Filter spurious matches
  – Build connectedness graph, with PageRank link strength scores
  – Prune (densify) graph to disambiguate entity
3. Entity relation extraction

• Task: Extract entity properties and event participants

• Four-step approach:
  1. BERT word embeddings for features
  2. Convolution: extract and merge all local features for a sentence
  3. Piecewise max pooling: split input into three segments (by position) and return max value in each segment, for 2 entities + 1 relation
  4. Softmax classifier to compute confidence of each relation
English entity/relation discussion

• Challenges and problems
  – Subsubtype is super fine-grained; our NER engine is still not robust enough
  – We return both type and subsubtype labels, but in the eval NIST will judge only one of them

• Mostly learned, but some manual assistance
Goals and challenges

• Goal: Extract entity and event mentions from Russian and Ukrainian text, and build frames

• Challenges:
  – Lack of pretrained off-the-shelf extractors
  – Lack of annotated data to train systems
  – Highly specific ontology

• Two pipelines:
  1. Rus and Ukr source text
  2. MT into English
Example input and output

**Input:** Про-российские сепаратисты атаковали Краматорский аэропорт.

*Translation:* Pro-Russian separatists attacked Kramatorsk airport.

**Output:**

- `mn0`: event `Conflict.Attack`, 
  Attacker: `mn1`, Target: `mn3`
- `mn5`: relation `GeneralAffiliation.MemberOriginReligionEthnicity`
  - Person: `mn1`, EntityOrFiller: `mn2`, 
  - text: Про-российские сепаратисты
- `mn6`: relation `Physical.LocatedNear`, 
  - EntityOrFiller: `mn3`, Place: `mn4`
  - text: Краматорский аэропорт
- `mn1`: entity `ORG`, 
  - text: Про-российские сепаратисты
- `mn2`: entity `GPE.Country.Country`, 
  - text: Про-российские
- `mn3`: entity `FAC.Installation.Airport`, 
  - text: Краматорский аэропорт
- `mn4`: entity `GPE.UrbanArea.City`, 
  - text: Краматорский
Approach 1: Processing in Rus/Ukr

Universal Dependency Parsing
StanfordNLP  UDPipe

Conceptual Mention Extraction (COMEX)
Ontology  Lexicon

- Our ontology is a superset of the NIST/LDC ontology
- Lexicons are (semi-)manually created from the training data
- Conceptual extraction using (manual) rule-based inference
- Focus is on high precision
Parsing/tagging/chunking pipeline

• Syntax pipeline:
  – UDPipe 1.2 (Straka & Strakova 2017)
  – Extract head nouns and dependents
  – Not all entities and events needed

• Event frame construction: COMEX
  – Our ontology is a superset of the AIDA ontology
  – Trigger terms manually mapped to ontology:
    • Direct matching — manually curated list of trigger words
    • English triggers — translation or WordNet/dictionary lookup
  – Analysis guided by annotation:
    • LDC annotations from seedling corpus
    • Own manual annotation as well
COMEX ontology

- Multiple inheritance
- Greater coverage

*entity

*physical-entity

*vehicle
LDC_ent_140

*airplane
LDC_ent_142

*mil-vehicle
LDC_ent_145

*fighter-plane
LDC_ent_146

*weapon
LDC_ent_160

*MiG-29
COMEX lexicons

- Connect words to ontology concepts via word senses
- Provide rules for connecting concepts into a mention graph
- Semantic requirements for slot fillers are specified in the ontology

While the lexicons contain hundreds of words, the number of rules is small
Lexicon construction

- Initial vocabulary and the corresponding concepts from the available LDC annotations
- Vocabulary enrichment by extracting all named and nominal entities from the seedling corpus files that contain at least one LDC annotation
- Event trigger enrichment using WordNet
- Cross-language vocabulary enrichment using MT and alignment
- Manual curation of the resulting vocabulary
- Manual addition of attribute rules
- Iterative improvement process:
  1. Extract mentions from a new file
  2. Score results
  3. Add vocabulary, fix rules and do cross-language transfer
### Sample COMEX performance

<table>
<thead>
<tr>
<th></th>
<th>English</th>
<th>Russian</th>
<th>Ukrainian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>0.91–1.0</td>
<td>0.93–1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Recall</td>
<td>0.22–0.56</td>
<td>0.11–0.70</td>
<td>0.07–0.42</td>
</tr>
<tr>
<td>F1</td>
<td>0.35–0.70</td>
<td>0.20–0.62</td>
<td>0.13–0.59</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>178</td>
<td>1483</td>
<td>1430*</td>
</tr>
<tr>
<td>Rules</td>
<td>33</td>
<td>30</td>
<td>13</td>
</tr>
</tbody>
</table>

(This work continues; the numbers change every day)

COMEX is the most ‘manual’ of OPERA’s TA1 extraction modules
Approach 2: Rus/Ukr $\text{MT} \rightarrow$ English

• Pipeline:
  – MT Rus/Ukr $\rightarrow$ English using MS Azure
  – Run OPERA TA1 extractors
  – Align source text to extracted mentions in Eng
    • Back-translate from Eng, including XML-like entity/event tags

• Output is generally good (esp when no XML tags)

• Problems in back-translation:
  – Sometimes messes up the XML tags
  – May switch event arguments
  – May mess up proper names (e.g. Slavyansk $\rightarrow$ Slavska, Slavovsk, Slavic
  – Things like typos or uncommon words get translated incorrectly into Eng, but may be easy to fix in the source using fuzzy matching
Approaches complementary

• Rus/Ukr: more precise
  – Less noise, better entity typing

• MT: more general
  – Better at names, time/numbers, event typing

• Overlaps and differences:
  – Entity overlap: 84% of Rus/Ukr = 44% of MT output
  – Event overlap: 58% of Rus/Ukr = 49% of MT output
  – Type agreement: 87% of overlap
  – Remaining mentions: 65–70% correct on each side
  – Differences in spans, event vs. entity choices
Rus/Ukr entity/relation discussion

• Challenges and problems
  – Slow manual rule building, limited coverage (but high precision)
  – COMEX<—>AIDA ontology alignment
  – Noise in translation

• Mostly manual
Hans Chalupsky

TA1/2 KB CONSTRUCTION AND VALIDATION
OPERA TA1 framework

Input

Domain filter & language detection

English pipeline

Speech pipeline

Ru/Uk pipeline

Image pipeline

MT: Ru/Uk → Eng

Ru/Uk Entity and Event detection

Event frame assembly II

Entity detection

Person and Geo ID

CSR Combination

Mini-KB creation/AIF validation

English entity detection

English event detection

English argument detection

English coref

Eng entity linking

Eng entity relations

Event frame assembly I

Event frame assembly II
CSR: PowerLoom-based Common semantic repository

- Contains all KEs
  - Contains discrete term propositions, [structured] distributional vectors/tensors, continuous embeddings
  - Each with vector of scores (e.g., TA1 extraction confidence, source trustworthiness, reasoning implication confidence, cross-KE compatibility, hypothesis-based likelihoods, etc.)

- Represented in PowerLoom (Chalupsky et al. 2010)
  - Predicate-logic-based representation based on KIF that is a supported syntax of Common Logic
  - Dynamic, scalable, multi-contextual system to store, manage and reason with information
  - Blazegraph database tech for scalability and integration
  - Represent hypotheses and probabilities via reification

- In the CSR everything is a hypothesis
M9 Approach: 3-step decoupling for KB construction and validation

Extractors ×N
- NER
- Entity Coref
- EDL
- DBPedia
- Events
- Event Coref
- Relations
- Audio
- Image, Video

Lots of type heterogeneity
- Multi-media Annotations
- Annotation Ontology

Reuse TACEVIC domain model
- Knowledge Aggregation
- Domain Ontology

Export to different targets
- Export Ontologies
- AIDA Seedling
- AIDA Full
- .....
M18 Approach: Single augmented ontology for KB construction and validation

Support some type heterogeneity

Domain additions, API to code

Knowledge Aggregation

Hypothesis Integration, Evaluation, Inference

AIDA Full

Blazegraph Triple store

Background KB

Extractors ×N

NER
Entity Coref
EDL
DBPedia
Events
Event Coref
Relations
Audio
Image, Video

Multi-media Annotations

Domain Ontology

Annotation Augmentations

Domain Augmentations

2

5

35
Incremental cycle of hypothesis representation, evaluation, refinement

- **Cycle:**
  - Use corefs and other identity to connect annotations (mention overlap, name links, EDL, within-doc coref, event coref)
  - Apply inferences, evaluate constraints, detect conflicts, do attribution
TA1/2 KB integration challenges

• Challenges: Ontological
  – Multiple type systems: NER types, relation types, event types, KB schemas, target schemas...
  – Missing types, conflicting types once things are linked
  – Types, even if fine-grained, primarily useful as constraints, not as equality signal – “Humvee17 generally-not-equal-to Humvee42”
  – Inference requirements: “Donechyna” and “Ukraine” are compatible locations of an event but not with respect to having “Donetsk” as their capital
  – Ontological “fluidity” — things change until late in the game

• Challenges: Data sparsity and noise
  – Multi-lingual names and cross-lingual matching
  – Language-specific naming schemes (e.g., patronyms)
  – Cross-lingual use of context vectors
  – No fine-grained document, text or media context allowed across documents
  – Linking decisions aggregate support and ontological conflict which propagates
### TA1 scores

#### TA1 Class queries

<table>
<thead>
<tr>
<th>Best MAP</th>
<th>Worst MAP</th>
<th>TREC MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4843</td>
<td>0.4737</td>
<td>0.4773</td>
</tr>
<tr>
<td>0.4527</td>
<td>0.3697</td>
<td>0.4020</td>
</tr>
<tr>
<td>0.4379</td>
<td>0.2816</td>
<td>0.3278</td>
</tr>
<tr>
<td>0.4243</td>
<td>0.1470</td>
<td>0.1957</td>
</tr>
<tr>
<td>0.2290</td>
<td>0.0892</td>
<td>0.1244</td>
</tr>
</tbody>
</table>

#### TA1 Graph queries

<table>
<thead>
<tr>
<th>Prec</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4715</td>
<td>0.2163</td>
<td>0.2966</td>
</tr>
<tr>
<td>0.4944</td>
<td>0.1328</td>
<td>0.2094</td>
</tr>
<tr>
<td>0.3605</td>
<td>0.0533</td>
<td>0.0929</td>
</tr>
<tr>
<td>0.0398</td>
<td>0.0312</td>
<td>0.0350</td>
</tr>
<tr>
<td>0.0138</td>
<td>0.0040</td>
<td>0.0062</td>
</tr>
</tbody>
</table>

Run: TA1a_OPERA_TA1a_aditi_V2
OPERA TA2 + TA3 framework
Candidate hypothesis generation

1. (Have completed Belief Graph and belief score propagation throughout)

2. Retrieve KEs corresponding to entrypoints

3. Retrieve events $E$ and relations $R$ that match constraints. If:
   - Zero-hop: the retrieved event is one hypothesis
   - One-hop: obtain an event for every role. Prioritize events/relations with maximum overlap with the roles — this may give many permutations

4. Generate hypothesis candidate set $H = h_1, h_2 \ldots h_n$ from the retrieved $E$ and $R$
Entry points are matched to Evidence Nodes

Information Need statement → Mention matcher → BG Evidence KEs

Hypothesis Graph selector → Hypothesis Graph candidates → Role inference

Augmented Hypothesis Graphs → Hypothesis ranking and filtering → Hypotheses
Candidate hypothesis generation

Knowledge Graph (KG)

- Retrieve candidate events and relations for each frame:
  - Candidate Events
    - Frame 1: E1, E2
    - Frame 2: E3
  - Candidate Relations
    - Frame 4: R1
    - Frame 5: R2, R3

- Further permutations can get generated if argument roles contain XORs. In that case, top 2 highest scoring alternatives are considered.

- Candidate hypotheses:
  1. E1 E3 R1 R2
  2. E2 E3 R1 R2
  3. E1 E3 R1 R3
  4. E2 E3 R1, R3
  ...

- Relation Frame F4, F5,
- Event Frame F1, F2, F3...
- Entity Entry Points
Hypothesis ranking

• Given information need I and candidate set \( H = h_1, h_2 \ldots h_n \), we need to rank \( H \) based on relevance and diversity

• Maximal marginal relevance:

\[
\text{MMR} = \lambda \arg\max_{h_i,E,H} \text{Sim}(h_i, I) - (1-\lambda) \arg\max_{h_j,E,H} \text{Sim}(h_i, h_j)
\]

\(- \text{Sim}(h_i, I) = \) similarity score between hypothesis \( h_i \) and the information need \( I \) — gives relevance

\(- \text{Sim}(h_i, h_j) = \) similarity score between hypotheses \( h_i \) and \( h_j \) — gives diversity
Relevance and diversity

• Sim(h_i, I) = Measuring **relevance** ... sum of:
  – Percentage of frames covered in I
  – Percentage of events satisfying the event frames
  – Percentage of relations satisfying the relation frames
  – Number of role-entity exact match constraints

• Sim(h_i, h_j) = Measuring **diversity** (= inverse similarity) between two hypotheses ... sum of:
  – Number of overlapping events
  – Number of overlapping relations
  – Number of overlapping entities
  – Number of overlapping arguments for the asked frames
TA3 M18 evaluation

• This was an *extremely* complex task
• We received a lot of numbers
• We’re still analyzing them

• Most of the numbers are not helpful for us
• Many things confuse us
• We wish for more detail about certain aspects
Task 3a (using own(any) TA2 KB)

<table>
<thead>
<tr>
<th>Hypos submitted</th>
<th>Theories matched</th>
<th>Correct-ness</th>
<th>Edge coherence</th>
<th>KE coherence</th>
<th>Rel strict</th>
<th>Rel lenient</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>6</td>
<td>0.4393</td>
<td>0.4834</td>
<td>0.6655</td>
<td>0.2832</td>
<td>0.6554</td>
<td>0.0320</td>
</tr>
<tr>
<td>42</td>
<td>4</td>
<td>0.2607</td>
<td>0.2894</td>
<td>0.423</td>
<td>0.1343</td>
<td>0.4192</td>
<td>0.0127</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0035</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.4167</td>
<td>0.4167</td>
<td>1.0000</td>
<td>0</td>
<td>1.0000</td>
<td>0.0032</td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td>0.3864</td>
<td>0.4475</td>
<td>0.5851</td>
<td>0.3295</td>
<td>0.4836</td>
<td>0.0032</td>
</tr>
</tbody>
</table>
Task 3a (using other TA2 teams’ KBs)

<table>
<thead>
<tr>
<th>Hypos submitted</th>
<th>Theories matched</th>
<th>Correctness</th>
<th>Edge coherence</th>
<th>KE coherence</th>
<th>Rel strict</th>
<th>Rel lenient</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>34</td>
<td>2</td>
<td>0.1042</td>
<td>0.2178</td>
<td>0.3711</td>
<td>0.1002</td>
<td>0.3512</td>
<td>0.0079</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>0.5107</td>
<td>0.6072</td>
<td>0.8145</td>
<td>0.3823</td>
<td>0.7973</td>
<td>0.0061</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0035</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.4167</td>
<td>0.4167</td>
<td>1.0000</td>
<td>0</td>
<td>1.0000</td>
<td>0.0032</td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td>0.3864</td>
<td>0.4475</td>
<td>0.5851</td>
<td>0.3295</td>
<td>0.4836</td>
<td>0.0032</td>
</tr>
</tbody>
</table>
## Task 3b (using LDC KB)

<table>
<thead>
<tr>
<th>Hypos submitted</th>
<th>Theories matched</th>
<th>Correctness</th>
<th>Edge coherence</th>
<th>KE coherence</th>
<th>Rel strict</th>
<th>Rel lenient</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>2</td>
<td>0.5961</td>
<td>0.6249</td>
<td>0.839</td>
<td>0.3829</td>
<td>0.839</td>
<td>0.0238</td>
</tr>
<tr>
<td>45</td>
<td>6</td>
<td>0.8065</td>
<td>0.8069</td>
<td>0.9589</td>
<td>0.6263</td>
<td>0.9589</td>
<td>0.0153</td>
</tr>
<tr>
<td>42</td>
<td>4</td>
<td>0.8266</td>
<td>0.8612</td>
<td>0.8979</td>
<td>0.8312</td>
<td>0.9034</td>
<td>0.0100</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>0.8207</td>
<td>0.8406</td>
<td>1.0000</td>
<td>0.5905</td>
<td>1.0000</td>
<td>0.0060</td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td>0.8925</td>
<td>0.9063</td>
<td>1.0000</td>
<td>0.7421</td>
<td>1.0000</td>
<td>0.0051</td>
</tr>
</tbody>
</table>
Hypothesis assessment procedure

• Assessment procedure:
  – First assess each edge as correct/incorrect
  – For only correct ones, match against the gold prevailing theory

• How assess/match? Decisions:
  – Type and informative mention of Edge
  – Type and informative mention of Left side
  – Type and informative mention of Right side

Defined in ontology.  Undefined.  Many Small differences.  differences of opinion
Confusion #1: Initial edge filtering

• Difference in assessed hypothesis scores on same gold-standard LDC KB input:

<table>
<thead>
<tr>
<th></th>
<th>Edges correct</th>
<th>Edges submitted</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERA</td>
<td>59.6%</td>
<td>2545</td>
</tr>
<tr>
<td>BBN</td>
<td>80.6%</td>
<td>908</td>
</tr>
<tr>
<td>GAIA</td>
<td>82.1%</td>
<td>571</td>
</tr>
<tr>
<td>UTexas</td>
<td>82.6%</td>
<td>1079</td>
</tr>
<tr>
<td>PNNL</td>
<td>89.3%</td>
<td>394</td>
</tr>
<tr>
<td>LDC on TA2</td>
<td>0.59 Precision</td>
<td></td>
</tr>
</tbody>
</table>

• Why the discrepancies? Our SIN-driven hypothesis creation was different. But why does LDC’s own Precision not get up to .80?
Confusion #2:

- Why did GAIA do a lot better on GAIA’s own KBs than on LDC’s KBs?

<table>
<thead>
<tr>
<th>Coverage</th>
<th>Query Type</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0320</td>
<td>_version2_QueryTypeBthroughE_GAIA_1.GAIA_2.GAIA_2_v2</td>
<td>GAIA2_v2 running on GAIA KBs</td>
</tr>
<tr>
<td>0.0248</td>
<td>_version4_QueryTypeBthroughE_GAIA_1.GAIA_2p.GAIA_2</td>
<td>GAIA2 running on GAIA KBs</td>
</tr>
<tr>
<td>0.0238</td>
<td>_version2_QueryTypeBthroughE_LDC_2.LDC_2.OPERATA3b_2</td>
<td>OPERA running on LDC KBs</td>
</tr>
<tr>
<td>0.0153</td>
<td>_version4_QueryTypeBthroughE_LDC_2.LDC_2.BBN_TA3_v2a</td>
<td>BBN running on LDC KBs</td>
</tr>
<tr>
<td>0.0127</td>
<td>_version2_QueryTypeBthroughE_OPERATA1a_hans_V3.OPERATA2_hans_V5.OPERATA3a_2</td>
<td>OPERA running on OPERA KBs</td>
</tr>
<tr>
<td>0.0100</td>
<td>_version3_QueryTypeBthroughE_LDC_2.LDC_2.UTexas_3</td>
<td>UTexas running on LDC KBs</td>
</tr>
<tr>
<td>0.0079</td>
<td>_version3_QueryTypeBthroughE_GAIA_1.GAIA_2.OPERATA3a_1</td>
<td>OPERA running on GAIA KBs</td>
</tr>
<tr>
<td>0.0061</td>
<td>_version2_QueryTypeBthroughE_BBN_1.BBN_TA2_v2.GAIA_2</td>
<td>GAIA running on BBN KBs</td>
</tr>
<tr>
<td>0.0060</td>
<td>_version2_QueryTypeBthroughE_LDC_2.LDC_2.GAIA_2</td>
<td>GAIA running on LDC KBs</td>
</tr>
<tr>
<td>0.0051</td>
<td>_version3_QueryTypeBthroughE_LDC_2.LDC_2.PNNL_sheafbox_10</td>
<td>PNNL running on LDC KBs</td>
</tr>
</tbody>
</table>
Confusion #3: Informative mentions

evt/rel:  data:relation-instance-HYP-E102-3-r201907150216-23
type:  ldcOnt:Physical.LocatedNear
handle:  woman of Odessa
edge prov:  HC000Q7MI:(5700-0)-(5714-0)
  woman of Odessa
-------------------------
edge:  ldcOnt:Physical.LocatedNear_EntityOrFiller
arg:  data:entity-instance-HYP-E102-3-r201907150216-0
handle:  the strangled woman of Odessa, who for pro-Russians has become a symbol of the Wests partiality in the Ukrainian crisis
assessed:  CORRECT
-------------------------
edge:  ldcOnt:Physical.LocatedNear_Place
arg:  data:entity-instance-HYP-E102-3-r201907150216-24
handle:  Odessa
assessed:  WRONG
Arg 1: the woman

edge:  ldcOnt:Physical.LocatedNear_EntityOrFiller
arg:  data:entity-instance-HYP-E102-3-r201907150216-0
handle:  the strangled woman of Odessa, who for pro-Russians has become a symbol of the Wests partiality in the Ukrainian crisis
conf:  1.000000
assessed:  CORRECT
best PT:  E102Theory4
arg prov:  HC000Q7MI:(5686-0)-(5804-0)
  the strangled woman of Odessa, who for pro-Russians has become a symbol of the Wests partiality in the Ukrainian crisis
context:
xactly what happened. This scarcity of information explains why so many rumours have emerged around >>the strangled woman of Odessa, who for pro-Russians has become a symbol of the Wests partiality in the Ukrainian crisis<<. This photo was originally posted here.
Arg 2: Odessa

- Why is it wrong? Some theories:
  - Odessa is not LocatedNear Odessa, it IS Odessa
  - The woman was born in Odessa but now lives in Kiev
  - The woman was only rumored to have been strangled
  - ...and more...

edge:  ldcOnt:Physical.LocatedNear_Place
arg:   data:entity-instance-HYP-E102-3-r201907150216-24
handle: Odessa
conf:  1.000000
assessed: WRONG
arg prov: HC000Q7MI:(5709-0)-(5714-0)
  Odessa
context:
  his scarcity of information explains why so many rumours have emerged around the strangled woman of >>Odessa<<, who for pro-Russians has become a symbol of the Wests partiality in the Ukrainian crisis. This p
Challenges

• Hypotheses graphs are too extensive, as events are connected by at least one common argument — need to add restrictions

• Background knowledge sometimes required to link entities (e.g., SU25 == military jet) — perhaps pre-populate KB with background knowledge?
FINALE
Finale

• OPERA is an end-to-end system
  – Successful combination of machine learning and manual components and approaches
  – Task 3b (using LDC’s KB) submission had the highest coverage
  – Managed with limited data and changing ontology
• Absorbed & processed GAIA TA1 & TA2 outputs
  – Got top TA2 graph query F1 (1a) score using GAIA KBs
• Current focus:
  – (Of course) improvements everywhere
  – New domain and ontology
  – Serious integration of component-level scores
Some discussion points

1. Multiple inheritance in the ontology

2. Main role of annotated data is as examples: continuous team–LDC interaction to gt system feedback?

3. It is hard to reconstruct what exactly assessors saw. If the specific textual context that an assessor looked at for a decision is recorded, then we can
   – see the text they based their judgment on
   – maybe also get some finer classification of the error type

4. TA1b bias: Component-based re-processing of same results when given new hypotheses