Error Analysis in an Automated Narrative Information Extraction Pipeline

by Josep Valls-Vargas, Jichen Zhu, and Santiago Ontanon

Sheyanova Mariya

NRU HSE

Table of contents

Introduction

Intro

- **Motivation**: gamedev with automatic plot generation
- **Data**: unannotated Russian folk tales translated into English
- **Information Schema**: Vladimir Propp's narrative theory (*Hero, Villain, Dispatcher, Donor, Helper, Soughtfor-person, and False Hero*)
- **Extraction tool**: *Voz* (to be described)
- **Purpose**: quantify the contribution of each module to the final error, identify the bottleneck with the largest impact

Voz

What it is?

A fully-automated narrative extraction system.

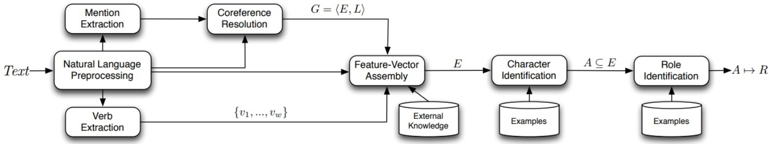


Fig. 2. Architecture of the *Voz* system illustrating the modules and workflow of the narrative information extraction pipeline.

NL Preprocessing

Stanford CoreNLP suite:

- segmentation
- POS-tagging
- syntactic parsing
- lemmatization

Coreference resolution: Stanford Coreference Resolution

Mention Extraction

A mention is a word or phrase used to refer to a specific entity or character, e.g. «the sister». Algorithm:

1. Voz traverses each of the sentence parse trees looking for an NP.
2. For each NP node, if the subtree rooted at the current NP node contains nested clauses, Voz traverses its subtrees. Otherwise, the node is marked as a mention.
3. For each mention, a feature vector is initialized.

The output of this module is a set of mentions (vectorized).

Other modules

- *Coreference Resolution*
- *Verb Extraction*: Voz considers each verb as a relationship between an executor and receiver of an action, and thus extracts verb triplets
- *Feature-Vector Assembly*: allows later modules in the pipeline to predict which mentions correspond to characters
- *Character Identification*
- *Role Identification*

Error Analysis

Questions

1. What is the error introduced by each module?
2. How much does the error introduced by one module affect a later module in the pipeline?

Methodology

1. Formalize the information extraction pipeline as a directed acyclic graph. Then, compute one topological ordering1 of the nodes in the DAG.
2. Annotate an incremental Ground Truth dataset. (in fact, GS for each module)
3. *Individual Module Evaluation.* Evaluate each module m_i using as input $GT_i - 1$ and comparing the output against GT_i.
4. *Error propagation.* Given a pair of modules m_a and m_b where $m_a m_b$, we compare the performance of m_b in two settings:
    - feed $GT_a - 1$ to m_a then running the pipeline from m_a to m_b
    - feed GT_a to $m_a + 1$ then running the pipeline from $m_a + 1$ to m_b

Evaluation

Measures: precision, recall, f-measure.
Coreference resolution module: a measure, characterizing the spread of a single character:

- the average number of coreference groups with a reference to a single character
- the misgrouping of different characters

Baselines:

- *Random*: generates predictions randomly
- *Informed*: always predicts the most common solution in the training set

Individual Module Evaluation

Each module m_i was evaluated using as input $GT_i - 1$ and comparing the output against GT_i.

Mention extraction, Coreference Resolution:

	P	R	f
Rand.	.446	.488	.467
Inf. B.	893	1.00	.944
Auto	.893	1.00	.944

	C/Gr	Gr/C
Rand.	10.7	1.00
Inf. B.	1.00	11.9
w/GT	1.07	6.00
Auto	1.07	6.00

Individual Module Evaluation

Verb Extraction, Character Identification:

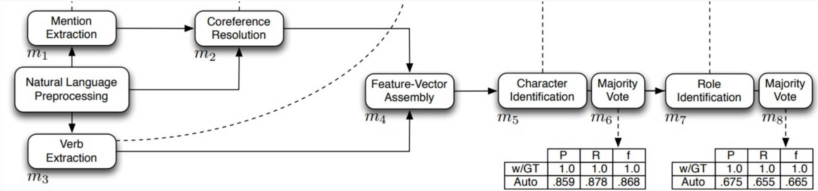
	P	R	f
Rand.	.021	.040	.027
Inf. B.	.089	.324	.139
Auto	.260	.204	.228

	P	R	f
Rand.	.502	.446	.448
Inf. B.	.423	.650	.512
w/GT	.850	.852	.851
Auto	.844	.876	.860

Role Identification:

	P	R	f
Rand.	.021	.143	.036
Inf. B.	.100	.316	.152
w/GT	.689	.672	.689
Auto	.685	.671	.675

Error propagation



	P	R	f
w/GT	1.0	1.0	1.0
Auto	.859	.878	.868

	P	R	f
w/GT	1.0	1.0	1.0
Auto	.675	.655	.665

Error propagation

TABLE III
EFFECT OF THE VERB FEATURES (FROM m_3) ON THE CHARACTER AND ROLE IDENTIFICATION PROCESSES (m_5 AND m_7). ROWS REPORT RESULTS USING THE AUTOMATICALLY EXTRACTED VERB FEATURES (m_3), USING THE VERB FEATURES FROM THE GROUND TRUTH GT_3, AND COMPLETELY REMOVING THE VERB FEATURES (GT_4 W/O VERBS).

Input	Module	P	R	f
m_3	m_5	0.844	0.876	0.860
GT_3	m_5	0.850	0.852	0.851
GT_4 w/o Verbs	m_5	0.832	0.851	0.841
m_3	m_7	0.685	0.671	0.675
GT_3	m_7	0.689	0.672	0.689
GT_4 w/o Verbs	m_7	0.618	0.595	0.602

Error propagation

TABLE IV

EFFECT OF COREFERENCE INFORMATION (FROM m_2) ON THE MAJORITY VOTING PROCESSES ($m6$ AND m_8). ROWS REPORT RESULTS WITHOUT COREFERENCE INFORMATION, USING THE AUTOMATIC COREFERENCE (m_2) AND USING THE COREFERENCE FROM THE GROUND TRUTH GT_2.

Voting	Module	P	R	f
Without Voting	m_6	0.844	0.876	0.860
m_2	m_6	0.859	0.878	0.868
GT_2	m_6	0.896	0.839	0.868
Without Voting	m_8	0.685	0.671	0.675
m_2	m_8	0.644	0.624	0.629
GT_2	m_8	0.728	0.713	0.714

Error propagation

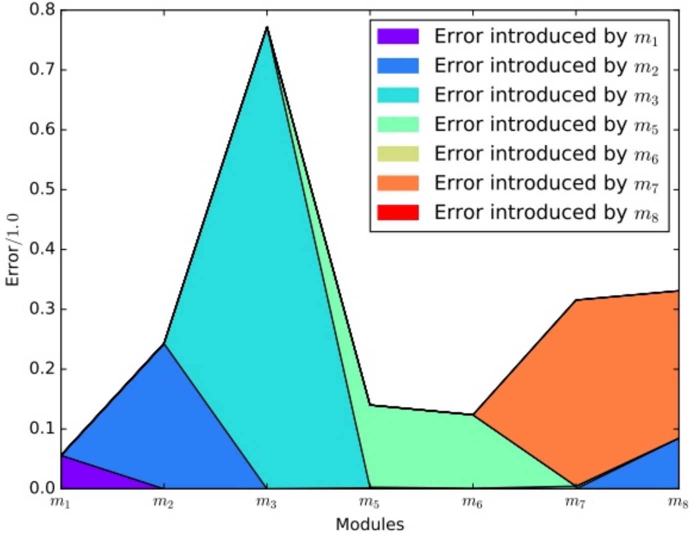


Fig. 5. Summary of error distribution through the pipeline when running *Voz*

Discussion

Three phenomena:

- the error between some modules is mitigated (i.e., Verb Extraction to Character Identification)
- the error introduced by certain modules has a considerable impact in later modules (i.e., Coreference Resolution to Role Identification)
- the error between some modules is independent (Character Identification and Role Identification)

Discussion

The main implication: working toward improving modules such as verb extraction will have no immediate impact on the performance of the character and role identification tasks.

Conclusion

Summary

The authors:

- want to improve narrative applications such as plot generators and interactive storytelling systems
- present *Voz*, a system for automatic narrative information extraction
- propose a novel methodology to study error propagation in information extraction pipelines

Conclusions

- part of the error of certain NLP tasks may be mitigated by later modules
- verb extraction sufficient to reduce the overall error of the system
- the impact of the coreference and verb extraction modules to the final result of the pipeline is smaller than anticipated

Questions?