Imitation learning, zero-shot learning and automated fact checking

Andreas Vlachos
http://andreasvlachos.github.io/
Introduction

- Lecturer at the Department of Computer Science at the University of Sheffield
  - Member of the natural language processing (NLP) and machine learning (ML) research groups

- Research in developing ML methods for:
  - natural language understanding: convert text into (machine-readable) meaning representations
  - natural language generation: convert meaning representations into (human-readable) text
  - applications encompassing both directions
Research Context
Natural Language Understanding (NLU)

- Named entity recognition (Vlachos et al., PSB 2006)
- Relation extraction (Vlachos and Craven, CoNLL 2011)
- Semantic parsing (Goodman et al., ACL 2016)
Natural Language Generation (NLG)

**INPUT:**

```plaintext
predicate= INFORM
name = "The Saffron Brasserie"
type = placetoeat
eattype = restaurant
area = riverside, "addenbrookes"
near = "The Cambridge Squash", "The Mill"
```

**OUTPUT:**

The Saffron Brasserie is a restaurant at the side of the river near the Cambridge Squash and the Mill in the area of Addenbrookes

- SOTA on 3 datasets (Lampouras and Vlachos, Coling 2016)
- NN-based system most fluent among 20 systems in End2End NLG (Chen et al., 2018)
Applications encompassing both directions

- Translation Quality Estimation (Beck et al., WMT 2016)
- Digital Personal Assistants (Vlachos and Clark, TACL 2014)
- Automated Fact Checking (Vlachos and Riedel, Computational Social Science and NLP 2014)
Machine Learning for Natural Language

Learning from data allows us to adapt rapidly to:
- language evolution
- different applications

Compared to rule-based approaches:
- wider coverage
- weighted feature combinations
- feature learning with neural networks/deep learning
  - reuse models across tasks (trade-off between feature engineering vs architecture engineering)
  - facilitate focus on novel tasks
This talk

- Improved structure prediction with *imitation learning*

- Ability to predict labels unseen during training using *zero-shot* learning with neural networks

- A challenge to advance ML, NLP and artificial intelligence: automated fact checking
Imitation learning for structured prediction
Structured prediction in NLP is everywhere

Sequences of labels, words and graphs combining them
Imitation learning for structured prediction

- Assume human-annotated input-output \((x, y)\) for supervised training

- Train a classifier to predict the actions \((\alpha)\) constructing the output \(y\)

- Actions not annotated; imitation learning is semi-supervised
Imitation learning in robotics

**Meta-learning:** better model (≈policy) by generating better training data from expert demonstrations
Relation to reinforcement learning

- Both reinforcement and imitation learning learn a classifier/policy to maximize reward
- Learning in imitation learning is facilitated by an expert
Breaking output into actions constructing it

structured output $y$

actions: $\alpha_1$ $\alpha_2$ $\cdots$ $\alpha_T$

states: $S_1$ $S_2$ $\cdots$ $S_T$

input sentence $x$
Incremental structured prediction

A classifier $f$ predicting actions to construct the output:

$$
\hat{\alpha}_1 = \arg \max_{\alpha \in A} f(\alpha, x),
$$

$$
\hat{y} = \text{output}
\left(
\hat{\alpha}_2 = \arg \max_{\alpha \in A} f(\alpha, x, \hat{\alpha}_1), \ldots
\right)
$$

$$
\hat{\alpha}_N = \arg \max_{\alpha \in A} f(\alpha, x, \hat{\alpha}_1 \ldots \hat{\alpha}_{N-1})
$$

✓ Use our favourite classifier
✓ No need to enumerate all possible outputs
✓ No modelling restrictions on features

$x$ Prone to error propagation
$x$ Classifier not trained w.r.t. task-level loss
Imitation learning

Improve incremental structured prediction by:
- addressing error-propagation
- training wrt the task-level loss function

Meta-learning: use our favourite classifier and features, but generate better training data

Can handle more complex problems than joint inference approaches:
- no output enumeration ⇒ no need for dynamic programming
- no dynamic programming ⇒ no modelling restrictions such as Markov assumptions used in conditional random fields, etc.
Imitation learning for part of speech tagging

Human annotated tags:

<table>
<thead>
<tr>
<th>Noun</th>
<th>Verb</th>
<th>Modal</th>
<th>Pronoun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*I*  
*can*  
*fly*

**expert policy:** at each word return the correct tag

**loss:** number of incorrect tags
Imitation learning for part of speech tagging

Standard incremental structured prediction:

<table>
<thead>
<tr>
<th>word</th>
<th>label</th>
<th>features</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Pronoun</td>
<td>token=I, prev=NULL ...</td>
</tr>
<tr>
<td>can</td>
<td>Modal</td>
<td>token=can, prev=Pronoun</td>
</tr>
<tr>
<td>fly</td>
<td>Verb</td>
<td>token=fly, prev=Modal</td>
</tr>
</tbody>
</table>
Imitation learning for part of speech tagging

Labels as costs:

<table>
<thead>
<tr>
<th>word</th>
<th>Pronoun</th>
<th>Modal</th>
<th>Verb</th>
<th>Noun</th>
<th>features</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>token=I, prev=NULL...</td>
</tr>
<tr>
<td>can</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>token=can, prev=Pronoun...</td>
</tr>
<tr>
<td>fly</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>token=fly, prev=Modal...</td>
</tr>
</tbody>
</table>
Imitation learning for part of speech tagging

Breaking down action costing:

- **rollin** to obtain a trajectory through the sentence
- **rollout** to complete the output prediction
- **cost** the complete output with the task loss

If rollin and rollout with the expert policy, correct labels have 0 cost, incorrect labels have 1.

<table>
<thead>
<tr>
<th>word</th>
<th>Pronoun</th>
<th>Modal</th>
<th>Verb</th>
<th>Noun</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>can</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>token=can, prev=Pronoun...</td>
</tr>
</tbody>
</table>
Imitation learning for part of speech tagging

Mixed rollins/rollouts with the expert policy and the classifier

<table>
<thead>
<tr>
<th>word</th>
<th>Pronoun</th>
<th>Modal</th>
<th>Verb</th>
<th>Noun</th>
<th>features</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>token=I, prev=NULL...</td>
</tr>
<tr>
<td>can</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>token=can, prev=Pronoun...</td>
</tr>
<tr>
<td>fly</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>token=fly, prev=Verb...</td>
</tr>
</tbody>
</table>
Back to learning how to drive

- Instead of observing the expert drive, let the classifier drive
- The expert gives the correct actions given the classifier’s ones
- The classifier is allowed to explore the effect of its own actions
Imitation learning for NLP

- Explores only the parts of the search space likely to be encountered ⇒ applicable to complex outputs
- Training data generation mixing expert and classifier ⇒ addresses error propagation
- Task loss only used on complete outputs ⇒ can train against non-decomposable loss functions such as BLEU, ROUGE, etc.
- Addresses a fundamental limitation of incremental predictors, including recurrent neural networks

More in our EACL 2017 tutorial, but now some real applications
Imitation learning for semantic parsing

- Convert a syntax tree to a meaning graph
- Long complex action sequences (>100 actions, 10K labels)
- Used in many applications: summarization, generation, etc.
Imitation learning benefits

- DAGGER uses rollins (Ross et al., AISTATS 2011)
- V-DAGGER uses roll-in/-outs (Vlachos and Clark, TACL 2014)
Semantic Parsing Evaluation

- Best reported results (Goodman et al., ACL 2016)
- No external resources used, just the training data
- Docker image of parser downloaded $>100$ times
## Imitation learning for Language Generation

**INPUT:**

```plaintext
predicate= INFORM
name = "The Saffron Brasserie"
type = place_to_eat
eat_type = restaurant
area = riverside, "addenbrookes"
near = "The Cambridge Squash", "The Mill"
```

**OUTPUT:**
The Saffron Brasserie is a restaurant at the side of the river near the Cambridge Squash and the Mill in the area of Addenbrookes

- Reversed semantic parsing, similar to machine translation (MT)
- Unlike MT, labeled data is rather limited
Language Generation - Human Evaluation

- SOTA on three datasets (Lampouras and Vlachos, 2016)
- No rules, re-ranking or templates, just two classifiers
More imitation learning applications

Own work:
- Biomedical Event Extraction (Vlachos and Craven, CoNLL2011)
- Language Understanding for Digital Personal Assistants (Vlachos and Clark, TACL 2014)
- Knowledge Base Population (Augenstein et al., EMNLP 2015)
- Machine Translation Quality Estimation (Beck et al., WMT 2016)

Others:
- Syntactic dependency parsing
  - Dynamic oracles (Goldberg and Nivre, Coling 2012)
  - LSTM-based (Ballesteros et al., EMNLP 2016)
  - Popular spacy.io NLP toolkit
- Coreference resolution (Clark and Manning, ACL 2015)
Zero-shot learning with neural networks
Zero-shot learning

ML models typically can predict only labels they saw in the training data, e.g. a model trained on cats and dogs can’t recognize birds.

Zero shot learning explores how to predict labels unseen in training.
Stance classification

Given a target concept, e.g. abortion or Hillary Clinton, decide whether a text is **positive/negative/neutral** towards the target:

No more Hillary!

Can we learn a model for targets unseen in training?

No more Hillary!
Zero-Shot Stance Classification

Standard supervised learning:
\[ \hat{y} = \arg \max_{y \in \mathcal{Y}} (w_t^y \cdot \phi(x)) \]
- learn weights \( w \) for each label \( y \) and target \( t \) assuming a feature construction \( \phi \) for tweet \( x \) (e.g. bag-of-words)
- fails for new targets (Trump vs Hillary)

Idea: use the target \( t \) in feature construction \( \phi \)
\[ \hat{y} = \arg \max_{y \in \mathcal{Y}} (w^y \cdot \phi(x, t; \theta)) \]
Learn the parameters \( \theta \) constructing the feature representation jointly with \( w \) using Long Short Term Memory Networks (LSTMs)
Stance Classification with Conditional LSTMs

- One LSTM encodes the target, another LSTM the tweet
- The representation of the tweet is **conditioned** on the target
- Same tweet-different target ⇒ **different stance**
Results

- Train on stance-annotated tweets for 5 targets, test on Trump
- State-of-the-art results without training data for target and with weak supervision (Augenstein et al., EMNLP 2016)
Zero-shot Relation Classification

<table>
<thead>
<tr>
<th>Relation</th>
<th>Subject (X)</th>
<th>Object (Y)</th>
<th>Text (Premise)</th>
<th>Description (Hypothesis)</th>
</tr>
</thead>
<tbody>
<tr>
<td>religious order</td>
<td>Lorenzo Ricci</td>
<td>Society of Jesus</td>
<td>X (August 1, 1703 – November 24, 1775) was an Italian Jesuit, elected the 18th Superior General of the Y.</td>
<td>X was a member of the group Y</td>
</tr>
<tr>
<td>director</td>
<td>Kispus</td>
<td>Erik Balling</td>
<td>X is a 1956 Danish romantic comedy written and directed by Y.</td>
<td>The director of X is Y</td>
</tr>
<tr>
<td>designer</td>
<td>Red Baron II</td>
<td>Dynamix</td>
<td>X is a computer game for the PC, developed by Y and published by Sierra Entertainment.</td>
<td>Y is the designer of X</td>
</tr>
</tbody>
</table>

Extended relation classification using descriptions instead of labeled data (Obamuyide and Vlachos, under review):

- Given training for director relation, we can predict designer
- Formulated the task textual entailment (sentence-pair classification)
Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>F1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMU-RC</td>
<td>ESIM</td>
<td>20.16</td>
</tr>
<tr>
<td></td>
<td>CIM</td>
<td>22.20</td>
</tr>
<tr>
<td>UW-RE</td>
<td>ESIM</td>
<td>61.32</td>
</tr>
<tr>
<td></td>
<td>CIM</td>
<td>63.58</td>
</tr>
</tbody>
</table>

- Good results on two datasets, improved using conditional encoding
- Can use labeled training data if available
Automated fact checking
A new challenge for AI: Automated fact-checking

The United Kingdom has ten times Italy’s number of immigrants.

FALSE: We find no data to support this claim. The UK does not have "ten times Italy’s number of immigrants".

<table>
<thead>
<tr>
<th>Country/Immigration</th>
<th>Italy</th>
<th>UK</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>4.92M</td>
<td>5.05M</td>
</tr>
<tr>
<td>2015</td>
<td>5.01M</td>
<td>5.42M</td>
</tr>
<tr>
<td>2016</td>
<td>5.03M</td>
<td>5.64M</td>
</tr>
</tbody>
</table>

(Vlachos and Riedel, 2014)
What do we want from automated fact-checking?

- Verdict justification, a.k.a. algorithmic transparency
  - Can’t convince otherwise
  - Need to check their correctness

- Generalization to different domains (economy, health, etc.)

- Learn with (relatively) little data

(Vlachos and Riedel, 2014)
What claims should we fact-check?

- Syrian refugees are not properly vetted or tracked by the FBI once in the US
- Leaving the EU would put 3M jobs at risk

- Does the source of the claim matter?
- Does the linguistic style matter?
Evidence for or against a claim

Claim: Doctors confirmed the first case of death by genetically modified food


Resolved  Added Mar 9

It originated on a fake news website and is therefore false. Emergent is as of now the only site to offer a full debunking.

Sources

Sources Tracked: 3  Total Shares: 62,188

For

2  Shares
60,596

Against

1  Shares
1,592
Results

- 300 claims from debunking website www.emergent.info
- Automated stance classification with 73% accuracy (Ferreira and Vlachos, 2016)
- Advisor to the Fake News Challenge with 50 participants
New datasets needed

AI successes follow dataset availability (Wissner-Gross, 2016)

<table>
<thead>
<tr>
<th>Year</th>
<th>Breakthroughs in AI</th>
<th>Datasets (First Available)</th>
<th>Algorithms (First Proposed)</th>
</tr>
</thead>
</table>

Average No. of Years to Breakthrough: 3 years 18 years

300 claims are not enough to learn fact checking
Fact Extraction and VERification (FEVER)

Claim:
The Rodney King riots took place in the most populous county in the USA.

Evidence:
[wiki/Los Angeles Riots]: The 1992 Los Angeles riots, also known as the Rodney King riots, were a series of riots, lootings, arsons, and civil disturbances that occurred in Los Angeles County, California in April and May 1992.
[wikipedia/Los Angeles County]: Los Angeles County, officially the County of Los Angeles, is the most populous county in the United States.

● 200K claims verified on Wikipedia (Thorne et al., NAACL 2018)
● 3-way classification:
  ○ The claim is SUPPORTED by the evidence
  ○ The claim is REFUTED by the evidence
  ○ NOT ENOUGH INFORMATION in Wikipedia to verify it
Annotation process details

- 50 annotators, all native speakers, trained by the authors or more experienced annotators
- Fixed Wikipedia dump to avoid changes in labels
- One annotator constructs the claim, different annotator verifies it
- Dedicated user interfaces were developed for the task
- Guidelines were refined through pilot studies
- Advised to spend 2-3 minutes per claim
- Instructed to avoid using their own world knowledge: “Shakira is Canadian” is NOT ENOUGH INFORMATION
Annotation findings

- 0.68 in Fleiss Kappa inter-annotator agreement on 3.4K claims
- 96.12% precision and 74.84% recall in evidence retrieval: measured against annotators who were not time-constrained
- Claims were 7.9 tokens long
- Multi-sentence evidence was chosen for 28.04% of the claims
- Evidence from different pages was chosen for 11.47%
- 7.6% of the mutated claims were excluded due to being too vague/ambiguous
- Final verification by the authors: 91.2% correct on 227 claims.
Results

Unlike previous tasks and datasets, evidence matters:
- a correct label with incorrect supporting evidence is wrong
- a simple approach using TF-IDF-based similarity for evidence selection and LSTMs for labeling the claim given the evidence achieved 31.87% acc. (50.91% ignoring evidence)

Room for improvement:

Fact Extraction and Verification (FEVER) shared task
- EMNLP 2018 workshop with Amazon Research Cambridge and Imperial College
- Interest from academics, industry and journalists and you?
Research summary

- Imitation learning for structured prediction in NLP
- Zero-shot learning with neural networks
- Automated fact-checking (see our Coling 2018 survey)

Other work:

- active learning (CSL 2008)
- Bayesian non-parametric approaches for NLP (PhD)
- syntax-based neural language models (ACL 2015, with Piotr Mirowski from Google DeepMind)
- authorship attribution with neural networks (EACL 2017, Coling 2018)
Thanks to my collaborators and sponsors

Looking forward to Cambridge from October!
Questions?