Unlocking Resources for Under-resourced Languages

Graham Neubig

Shruti Rijhwani, Xinyi Wang

Antonios Anastasopoulos, Daisy Rosenblum, Sebastian Ruder
Language technologies in a multilingual world
Language technologies in a multilingual world

Considerable recent progress in expanding NLP to many languages!
Language technologies in a multilingual world

Considerable recent progress in expanding NLP to many languages!

Multilingual benchmark datasets for NLP tasks

- WMT
- MasakhaNER
- TyDiQA
- XNLI
- FLORES-101
- UD Treebank
- XTREME
Language technologies in a multilingual world

Considerable recent progress in expanding NLP to many languages!

Multilingual benchmark datasets for NLP tasks

- WMT
- MasakhaNER
- TyDiQA
- XNLI
- FLORES-101
- UD Treebank
- XTREME

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR
Language technologies in a multilingual world

Considerable recent progress in expanding NLP to many languages!

Multilingual benchmark datasets for NLP tasks
- WMT
- MasakhaNER
- TyDiQA
- XNLI
- FLORES-101
- UD Treebank
- XTREME

Pretrained multilingual language models
- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Commercial models that support many languages
- Voice assistants
- Predictive keyboards
- Translation
- Web search
- Text analytics
Language technologies in a multilingual world

Pretrained multilingual language models

XLM-R  mBERT  mT5  mBART  ERNIE-M  Turing ULR
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Trained on unlabeled text corpora (e.g., Wikipedia and Common Crawl)
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages
Language technologies in a multilingual world

Pretrained multilingual language models
- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages

Enables NLP applications through cross-lingual transfer
- Named entity recognition
- Entity linking
- Web search
- Machine translation
- Question answering
  ...

...
Support 100 – 200 languages

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 - 200 languages

Multiple societal benefits of NLP that includes many languages!
Language technologies in a multilingual world

Pretrained multilingual language models
- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages

Wikipedia
From Wikipedia, the free encyclopedia

This article is about the online encyclopedia written and maintained by a community of contributors. For other languages, see List of Wikipedias.

Access to information and education from other languages

Automatic translation of information

Multiple societal benefits of NLP that includes many languages!
Language technologies in a multilingual world

Pretrained multilingual language models
- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages

Multiple societal benefits of NLP that includes many languages!

Access to information and education from other languages

Language technologies that serve many more people!
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

There are over 7000 living languages!

Support 100 – 200 languages

- 2%

98%
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages

6500+ languages are under-represented in NLP
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

Support 100 – 200 languages

6500+ languages are under-represented in NLP

Over 2.2 billion people are under-served by modern language technologies
Language technologies in a multilingual world

Pretrained multilingual language models

- XLM-R
- mBERT
- mT5
- mBART
- ERNIE-M
- Turing ULR

What’s stopping us from expanding NLP systems to more languages?

Support 100 – 200 languages

98%

2%
The unlabeled text bottleneck
The unlabeled text bottleneck

Number of Wikipedia Articles

<table>
<thead>
<tr>
<th></th>
<th>1,750,000</th>
<th>3,500,000</th>
<th>5,250,000</th>
<th>7,000,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cebuano</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swedish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>German</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>French</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dutch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spanish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Egyptian Arabic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Polish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vietnamese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Waray</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chinese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arabic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ukrainian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portuguese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Persian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Catalan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Serbian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Indonesian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norwegian (Bokmål)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Korean</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finnish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hungarian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turkish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Serbo-Croatian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chechen</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Southern Min</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Romanian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bashkort</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tatar</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Malay</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Esperanto</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hebrew</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Armenian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bulgarian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Danish</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>South Azerbaijani</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slovak</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kazakh</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minangkabau</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estonian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belarusian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Croatian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Greek</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lithuanian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple English</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Azerbaijani</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Galician</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slovene</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Urdu</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norwegian (Nynorsk)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Georgian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hindi</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thai</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tamil</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uzbek</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latin</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Welsh</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asturian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Volapük</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Macedonian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cantonese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bengali</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Labrador</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tajik</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Burmese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Afrikaans</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Malay</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bosnian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ocitano</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Albanian</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low German</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Marathi</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kyrgyz</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Māori</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belarusian (Tarasievica)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Telugu</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Newar</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Breton</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swahili</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Venetan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Piedmontese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The unlabeled text bottleneck
The unlabeled text bottleneck
The unlabeled text bottleneck

- Annotated datasets: need existing text data or have to recruit speakers to create it.
The unlabeled text bottleneck

- Annotated datasets: need existing text data or have to recruit speakers to create it.
- Multilingual LMs: performance is limited by the amount of text available!
The unlabeled text bottleneck
The unlabeled text bottleneck

Some text is available in few hundred languages
The unlabeled text bottleneck

Some text is available in few hundred languages

Thousands of languages languages without sufficient text for developing NLP systems
The unlabeled text bottleneck

Some text is available in few hundred languages

How do we move towards including these languages in modern NLP systems?
Text resources do exist in many more languages!
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

Printed books
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

- Printed books
- Handwritten notes
- Typewritten documents
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

- Printed books
- Handwritten notes
- Typewritten documents

or in other formats such as bilingual lexicons
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

Printed books

Handwritten notes

Typewritten documents

or in other formats such as bilingual lexicons

PanLex
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

- Printed books
- Handwritten notes
- Typewritten documents

or in other formats such as bilingual lexicons
Text resources do exist in many more languages!

But locked away in formats that are not machine-readable

- Printed books
- Handwritten notes
- Typewritten documents

or in other formats such as bilingual lexicons

what can we do?!
Text resources do exist in many more languages!
Text resources do exist in many more languages!
Unlocking non-traditional resources

PanLex
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

XLM-R  mBERT
mT5  mBART  ERNIE-M
Turing ULR
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

Annotate datasets for downstream NLP tasks

XLM-R  mBERT  mT5  mBART  ERNIE-M  Turing ULR

PanLex
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

Annotate datasets for downstream NLP tasks

Support communities that speak these languages

PanLex
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

Annotate datasets for downstream NLP tasks

Support communities that speak these languages

Make native texts digitally accessible and searchable

PanLex

Text resources do exist in many more languages!

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

Annotate datasets for downstream NLP tasks

Support communities that speak these languages

Make native texts digitally accessible and searchable

PanLex
Text resources do exist in many more languages!

Unlocking non-traditional resources

Enable NLP for under-resourced languages

Expand multilingual LMs to more languages

Annotate datasets for downstream NLP tasks

Support communities that speak these languages

Make native texts digitally accessible and searchable

Aid language researchers, educators, libraries…

PanLex
Unlocking Un-digitized Text


Extracting text from scanned documents

"Ma ti eXi’ pu klei’?"
"Iklèo ka ‘tela n’armastò."
I vèkkia àggale tria dattilitia:

Scanned document
Extracting text from scanned documents

"Ma ti eì? pu klei'?"
"Iklèo ka ètela n'armastò."
I vèkka âggale tria datilitia:
Extracting text from scanned documents

"Ma ti exi' pu klei'?"
"Iklèo ka itela n'armastò."
I vèkkia àggale tria datilitia:

Scanned document
Extracting text from scanned documents

”Ma ti ezi’ pu klei’?”
”Iklèo ka ìtela n’armastò.”
I vèkkia àggale tria dattilitia:

Optical Character Recognition (OCR)

Scanned document
Extracting text from scanned documents

Optical Character Recognition (OCR)

Scanned document

"Ma ti eşi pu klei?"
"Ikléo ka itela n’armastò."
I vèkkia àggale tria dattilitia:

Machine readable text

"Ma ti eşi pu klei?"
"Ikléo ka itela n’armastò."
I vèkkia àggale tria dattilitia:
Extracting text from scanned documents

"Ma ti eži' pu klei'?”
"Iklèo ka îtela n’armastò.”
I vèkkia àggale tria dattilitàia:

Optical Character Recognition (OCR)

"Ma ti eži' pu klei'?”
"Iklèo ka îtela n’armastò.”
I vèkkia àggale tria dattilitàia:

Scanned document

Machine readable text
Extracting text from scanned documents

- High accuracy on languages that have easily available resources!
Extracting text from scanned documents

- High accuracy on languages that have easily available resources!
- Off-the-shelf tools support many scripts and languages
Extracting text from scanned documents

- High accuracy on languages that have easily available resources!
- Off-the-shelf tools support many scripts and languages

Scanned document

Optical Character Recognition (OCR)

Machine readable text

"Ma ti eşi' pu klei?"
"Iklèo ka ìtela n'armastò."
I vèkkia àggale tria dattilitia:

"Ma ti eşi' pu klei?"
"Iklèo ka ìtela n'armastò."
I vèkkia àggale tria dattilitia:

Google Vision
Tesseract
EasyOCR
...

"Ma ti eşi' pu klei?"
"Iklèo ka ìtela n'armastò."
I vèkkia àggale tria dattilitia:
Extracting text from scanned documents

- High accuracy on languages that have easily available resources!
- Off-the-shelf tools support many scripts and languages

Support 80-100 languages

Google Vision
Tesseract
EasyOCR

Scanned document

Optical Character Recognition (OCR)

Machine readable text
High accuracy on languages that have easily available resources!

Off-the-shelf tools support many scripts and languages

Little to no prior work on very low-resourced settings
Extracting text from scanned documents

• Little to no prior work on very low-resourced settings
Extracting text from scanned documents

• Little to no prior work on very low-resourced settings

Evaluation dataset
Promises and pitfalls of existing methods
Neural models for improving OCR performance in low-resource settings

Extracting text from scanned documents

• Little to no prior work on very low-resourced settings

Evaluation dataset
Promises and pitfalls of existing methods
Neural models for improving OCR performance in low-resource settings

Extracting text from scanned documents

- Little to no prior work on very low-resourced settings

Evaluation dataset
Promises and pitfalls of existing methods
Neural models for improving OCR performance in low-resource settings

Extracting text from scanned documents

• Little to no prior work on very low-resourced settings

Evaluation dataset
Promises and pitfalls of existing methods
Neural models for improving OCR performance in low-resource settings
Semi-supervised learning to improve performance with unlabeled images

"Ma ti eşi’ pu klei’?"
"Iklèo ka îtrela n’armastò."
I vèkkia àggale tria dattilitia:

Optical Character Recognition (OCR)

"Ma ti eşi’ pu klei’?"
"Iklèo ka îtrela n’armastò."
I vèkkia àggale tria dattilitia:

Machine readable text

Evaluation dataset for low-resource OCR
Evaluation dataset for low-resource OCR
Evaluation dataset for low-resource OCR

Ainu (Japan)

kira-an patek
aeyairamshitne

hushkotai wano

iki-an aine

Griko (Italy)

"Ma ti exi’ pu klei’?"
"Iklèo ka itela n’armastò."
I vèkkia àggale tria dattilitia:
Evaluation dataset for low-resource OCR

Ainu (Japan)

kira-an patek
aeyairamshitne(1)
sho-hushkotoi wano(2)
iki-an aine

Griko (Italy)

"Ma ti e'xi' pu klei'?"
"Iklèo ka itela n'armastò."
I vèkkia àggale tria dattilitia:

Yakkha (Nepal)

मा, ना यिगा निङ्ग्याङ्ग ओम,
हास्योक्सागी लेम्माङ्ग लाःला लुथा,
"पिङ्नाङ्ग लेम्माहोड़ प्याक छो छो
लाङ्ग्याङ्ग मेंजोंक्माहा।"
Evaluation dataset for low-resource OCR

Ainu (Japan)

```
kira-an patek
aeyairamshitne
hushkotoi wano
iki-an aine
```

Griko (Italy)

"Ma ti eXi' pu klei'?
'Iklèo ka itela n'armastò."

I vèkkia àggale tria dattilitia:

Yakkha (Nepal)

"Mā, na vinga niṣṭvāmaṅ ∆ōm,
haṅsokṇāṅi lomāṅ saṅsaṅa ṃaya,
"piṅgaṅāṅu letaṅmaṅaṅ pvaṅ kō ṃo
laṃlaṅ maṅ maṅjoṅkmaṅa."

Kwak’wala (Canada)

```
qu’eləx gwěg’ılasasa lexələx lexat’yə
lexələs naṅkwəxə nek’ulə. Wə, hə’n
wə, la hełəda ’nemsgənə; wə, hə’mi
lexələs. Wə həem lęgəmsa ’wələga.
```
Evaluation dataset for low-resource OCR

- Orthographically, typologically, geographically diverse
## Evaluation dataset for low-resource OCR

<table>
<thead>
<tr>
<th>Language</th>
<th>Text</th>
</tr>
</thead>
</table>
| Ainu (Japan) | kira-an patek aeyairamshite

| Griko (Italy) | "Ma ti ẹxị pu klei'?"  
"Iklèo ka itela n'armastò."  
I vèkkia ąggale tria dattilitia:  

| Yakkha (Nepal) | मा, ना चिया निहृवामाह ओम,  
हाकोङ्डोगी लेम्साङ्ग लाङ्गा लुया,  
"पिछानागा लेज्जमासो प्याकः छो छो  
लाप्लाप मेंजीतमाहा।"  

| Kwak’wala (Canada) | q'alelax gwégi'lasasa lexèli'xa lexu'ye  
lexelasa nekwäxa nek'üle. Wà, hē'n wà, la he'dà 'nemsgenë; wà, hē'mi lexelas. Wà hēem łegemso 'wäléga' |

- Orthographically, typologically, geographically diverse

Evaluation dataset for low-resource OCR

- **Ainu** (Japan)
  - Latin
  - orthographically, typologically, geographically diverse

- **Griko** (Italy)
  - Latin+Greek
  - "Ma ti eXi' pu klei'?"
  - "Iklèo ka itela n'armastò."
  - I vëkkia àggale tria dattilitia:

- **Yakkha** (Nepal)
  - Latin+Greek
  - मा, ना चिपा निह्न्यामाहार्णोम, नाचूने डागौ लेम्नुस खालु लुम्नावः
  - "पिण्यासाय लेम्नाहार्णोपाल्य छो छो लाप्लाप मेन्जोक्माहाय।"

- **Kwak’wala** (Canada)
  - Latin+Greek
  - q'alelaq gwēg'ílasasa lexēlaxa lexat'ye lelexílasa nekwāxā nekwālē. Wā, hēn wā, la hēlēda 'nemsgenē; wā, hēmí lelexílas. Wā hēem tēgema 'wālēga'
Evaluation dataset for low-resource OCR

- Ainu (Japan)
  - kira-an patek
  - aeyairamshitne
  - shushkotoi wano
  - iki-an aine

- Griko (Italy)
  - "Ma ti e Xi pu klei’?"
  - "Iklëo ka itela n’armastò."
  - I vëkkia àggale tria dattilitia:

- Yakkha (Nepal)
  - मा, ना चिमा निश्चितामाह ओम,
  - हाशोङ्कङ्गो लेम्साहङ्ग लायला सुणा,
  - "पिङ्ङङ्गामा लेड्महोडङ्ग प्याङ्ग छो छो
  - लाप्लाप मेन्जौक्माहा।"

- Kwak’wala (Canada)
  - q’alelax gwèg’ilasasa lexèlaxa lexu’yè
  - lexelas nekwaxa nek’ułe. Wà, hë’n wà, la hełèda ’nemsgënè; wà, hë’mi
  - lexelas. Wà hëem téngensa ’wàlèga’

- Orthographically, typologically, geographically diverse

Evaluation dataset for low-resource OCR

- Orthographically, typologically, geographically diverse

Ainu (Japan)  
```
kira-an patek  
ayeairamshitne\(^1\)  
shushkotoi wano\(^2\)  
iki-an aine
```

Griko (Italy)  
```
"Ma ti e\'i pu klei'?"  
"Ikl\'e\'o ka itela n'armast\'o."  
I v\'ekkia a\'ggale tria dattilitia:
```

Yakkha (Nepal)  
```
मा, ना चिणा निघ्नामाइ ओम,  
हाखोड़ागी तेम्माइ लाघ्ना लुया,  
"पिछनांगा लेप्माहोड़ प्याक छो छो  
लाप्प्लाप मेन्जीव्माहाइ।"
```

Kwak’wala (Canada)  
```
q\'alelax gw\'eg\'ilasasa lexel\'aaxa lexel\'y\'e  
lexel\'aaxa nek\'waxa nek\'ul\'e.  
Wä, hë\'n wä, la hel\'ëd\'a \'nemsgenë; wä, hë\'m\'i  
lexelas.  Wä hë\'em \'eg\'emsa \'wâlëga'
```

Latin

Latin+Greek

Devanagari

Boas

Evaluation dataset for low-resource OCR

- Ainu (Japan)
  - kira-an patek
  - aeyairamshitne(1)
  - shushkotoi wano(2)
  - iki-an aine

- Griko (Italy)
  - "Ma ti exi' pu klei'?"
  - "Iklèo ka itela n'armastò."
  - I vèkkia âggale tria datitilia:

- Yakkha (Nepal)
  - मा, ना विपा नित्न्वामाङ ओम,
  - हासोकडागी लेम्साङ लाङ्ना लुया,
  - "पिछनाछा लेंड्माहों प्याक छो छो लाप्लाप में-जोकमहान।"

- Kwak’wala (Canada)
  - q'àlelax gwèg’ilasasa lexèlaxa lexà’yè
  - lexelása nekwàxà nek’ulé. Wà, hè’n wà, la helèda ‘nemsgènè; wà, hè’mi
  - lexelàs. Wà hèem tégemsa ‘wàléga

- Orthographically, typologically, geographically diverse

- The languages currently have:
  - No Wikipedia/Common Crawl text
  - Not supported by multilingual LMs
  - No easily accessible bilingual lexica

Evaluation dataset for low-resource OCR

Ainu
(Japan)
kira-an patek
ayairamshirne
$hushkotoi wano$ 
iki-an aine

Griko
(Italy)
"Ma ti eix' pu klei'?"
"Ikléo ka itela n'armastò."
I vèkkia àggale tria dattilitia:

Yakkha
(Nepal)
Ma, na miga nişvamah odum, hašvōdbaši lemvamahu laša dhuma, "pičanamah lemvamah o p'yaq choo lapalap manjoekmahah.

Kwik'wala
(Canada)
q'alelah gwèg-îlasa lexélaxa lexatyelexelasa nekwaxa nek'ulë. Wā, hē'n wā, la heleda 'nemsgen.: wā, hēmi lexelas. Wā hēem tégeansa 'wälēga'

• Orthographically, typologically, geographically diverse
• The languages currently have:
  • No Wikipedia/Common Crawl text
  • Not supported by multilingual LMs
  • No easily accessible bilingual lexica
• <1000 transcribed lines per language
Existing OCR methods
Existing OCR methods

Supervised

Large neural networks

Requires: 10000s of transcribed images
Existing OCR methods

Supervised

- Requires: 10000s of transcribed images
- Large neural networks

Unsupervised

- Unlabeled images
- Language model

Requires: text corpus or lexicon in the target language
Existing OCR methods

**Supervised**
- Large neural networks
- Requires: 10,000s of transcribed images

**Unsupervised**
- Unlabeled images
- Language model
- Requires: text corpus or lexicon in the target language

**Off-the-shelf**
- Support ~100 languages
- Not trained on our target languages
- Can act as a general character recognizer for many scripts
Existing OCR methods

**Supervised**

- Requires: 10000s of transcribed images
- Large neural networks

**Unsupervised**

- Unlabeled images
- Language model
- Requires: text corpus or lexicon in the target language

**Off-the-shelf**

- Support ~100 languages
- Not trained on our target languages
- Can act as a general character recognizer for many scripts
Existing OCR methods

**Supervised**

- Large neural networks
- Requires: 10,000s of transcribed images

**Unsupervised**

- Unlabeled images
- Language model
- Requires: text corpus or lexicon in the target language

**Off-the-shelf**

- Support ~100 languages
- Not trained on our target languages
- Can act as a general character recognizer for many scripts
Existing OCR methods: promises and pitfalls
Existing OCR methods: promises and pitfalls
Existing OCR methods: promises and pitfalls

Word edit distance between prediction and reference

Number of words in reference

% Word Error Rate

0 25 50 75

Ainu  Griko  Yakkha  Kwak'wala
Existing OCR methods: promises and pitfalls

Lower is better!
Existing OCR methods: promises and pitfalls

Off-the-shelf models for languages in 29 scripts
Script-specific models

Lower is better!
Existing OCR methods: promises and pitfalls

Lower is better!

% Word Error Rate

Google Vision

Ainu
Griko
Yakkha
Kwak'wala
Existing OCR methods: promises and pitfalls

Google Vision performs well if the script is known, even without any language specific data.
Existing OCR methods: promises and pitfalls

Lower is better!

Google Vision

Script not known to the model
Existing OCR methods: promises and pitfalls

Example in Griko: \text{exi i k\dd{a}din\ara!}
Existing OCR methods: promises and pitfalls

Example in Griko:

exi i kaddinàra!

exi i kaddinàra!
Existing OCR methods: promises and pitfalls

Example in Griko:

exī i kāddinàra!

exī i kāddinàra!
Existing OCR methods: promises and pitfalls

Example in Griko:

Google Vision

Mixed scripts

Uncommon diacritics

Exi i kaddinàra!
Existing OCR methods: promises and pitfalls
Existing OCR methods: promises and pitfalls

Lower is better!
Existing OCR methods: promises and pitfalls

% Word Error Rate

Lower is better!

- Ainu
- Griko
- Yakkha
- Kwak'wala

Google Vision

Ocular

Open-source unsupervised OCR software
Requires an LM in the target language
Existing OCR methods: promises and pitfalls

% Word Error Rate

Lower is better!

- Google Vision
- Ocular

Languages:
- Ainu
- Griko
- Yakkha
- Kwak'wala
Existing OCR methods: promises and pitfalls

Lower is better!

Google Vision

Ocular

Ocular is highly reliant on the LM
Existing OCR methods: promises and pitfalls

% Word Error Rate

Google Vision

Ocular

Ocular is better for Kwak’wala

Lower is better!
Existing OCR methods: promises and pitfalls

% Word Error Rate

- Google Vision
- Ocular

<table>
<thead>
<tr>
<th>Language</th>
<th>Google Vision</th>
<th>Ocular</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ainu</td>
<td>6.3</td>
<td>47.5</td>
</tr>
<tr>
<td>Grikö</td>
<td>15.6</td>
<td>15.8</td>
</tr>
<tr>
<td>Yakkha</td>
<td>31.6</td>
<td>75.0</td>
</tr>
<tr>
<td>Kwak’wala</td>
<td>38.2</td>
<td>82.1</td>
</tr>
</tbody>
</table>
Existing OCR methods: promises and pitfalls

Google Vision vs. Ocular

- Ainu: 6.3% (Google Vision) vs. 47.5% (Ocular)
- Griko: 15.6% (Google Vision) vs. 15.8% (Ocular)
- Yakkha: 31.6% (Google Vision) vs. 75.0% (Ocular)
- Kwak'wala: 31.6% (Google Vision) vs. 82.1% (Ocular)

- Considerable room for improvement compared to high-resource languages
Existing OCR methods: promises and pitfalls

Google Vision

Ocular

% Word Error Rate

• Considerable room for improvement compared to high-resource languages

• Recognizes the majority of words correctly
Existing OCR methods: promises and pitfalls

- Considerable room for improvement compared to high-resource languages
- Recognizes the majority of words correctly
- Reliable starting point for further improvements
Improving the results of existing OCR systems
Improving the results of existing OCR systems

"Ma ti e xi’ pu klei’?"
"Iklèo ka ìtela n’armastò."
I vèkkià àggale tria dattilìtìa:
Improving the results of existing OCR systems

OCR output ("first pass")
Improving the results of existing OCR systems

OCR output ("first pass")

OCR output has some errors
Improving the results of existing OCR systems

OCR output ("first pass")

Automatic OCR Post-Correction
Improving the results of existing OCR systems

OCR output (“first pass”) → Automatic OCR Post-Correction → Corrected transcription

"Ma ti eXi’ pu klei’?"
"Iklèo ka itela n’armastò.”
I vèkkia àggale tria dattilitia:
Improving the results of existing OCR systems

OCR output (“first pass”)

Automatic OCR Post-Correction

Corrected transcription

"Ma ti eχi’ pu klei’?"
"Iklèo ka ītela n’armastò."
I vèkkia āggale tria dattilitia:

"Ma ti eχi’ pu klei’?"
"Iklèo ka ītela n’armastò."
I vèkkia āggale tria dattilitia:

Previous work: improve results for unseen fonts, layouts, domains.
This talk: low-resourced languages.
Improving the results of existing OCR systems

Automatic OCR Post-Correction

Text-based sequence-to-sequence task

OCR output ("first pass")

Corrected transcription
Improving the results of existing OCR systems

OCR output ("first pass")

Text input

Automatic OCR Post-Correction

Text-based sequence-to-sequence task

Corrected transcription
Improving the results of existing OCR systems

OCR output ("first pass")

Text input

Automatic OCR Post-Correction

Text-based sequence-to-sequence task

Corrected transcription

Text output
Adapting to low-resource settings

Prior work: character-level encoder-decoder with attention

- Add structural biases to the model
- Diagonal attention loss, copy mechanism, coverage mechanism
Adapting to low-resource settings

Prior work: character-level encoder-decoder with attention

- Add structural biases to the model
  - Diagonal attention loss, copy mechanism, coverage mechanism
- Leverage additional information from the source document
What additional information is available?

- Many documents containing text in low-resource languages also contain a translation of the text.

- Interlinear glosses, dictionaries, linguistic documentation, language learning material...
Multi-source model for post-correction
Multi-source model for post-correction

Low-resource language encoder

Griko first pass OCR
eXi i kaddinàra!

Attention
Multi-source model for post-correction

Italian first pass OCR

possiede la gallinaia

Low-resource language encoder

Griko first pass OCR
e\text{xi} i \text{ka}d\text{dinàra}!

Attention
Multi-source model for post-correction

Italian first pass OCR
possiede la gallinaia

High-resource translation encoder

Low-resource language encoder
exi i kaddiinàra!

Attention
Multi-source model for post-correction

Italian first pass OCR

possiede la gallinaia

High-resource translation encoder

Attention

Low-resource language encoder

Griko first pass OCR

exi i kaddinàra!
Multi-source model for post-correction

Italian first pass OCR
possiede la gallinaia

High-resource translation encoder

Attention

Concatenate context vectors

Decoder LSTM

Low-resource language encoder
exi la diinàra!

Attention

Griko first pass OCR
Multi-source model for post-correction

Italian first pass OCR

possiede la gallinaia

High-resource translation encoder

Attention

Concatenate context vectors

Decoder LSTM

Low-resource language encoder

exi i ka\text{ddinàra!}

Griko first pass OCR

Supervised training with a small amount of annotated data (<1000 lines)
Experiments: how do existing post-correction methods perform?
Experiments: how do existing post-correction methods perform?

% Word Error Rate

- First pass OCR
- Encoder-decoder

Ainu  Griko  Yakkha  Kwak’wala
Experiments: how do existing post-correction methods perform?

% Word Error Rate

- First pass OCR
- Encoder-decoder

Languages:
- Ainu
- Griko
- Yakkha
- Kwak’wala

Authors:
- Dong and Smith, 2018
- Hämäläinen and Hengchen, 2019
- Todorov and Colavizza, 2020
- Lyu et al., 2021
- Duong et al., 2021
- ...
Experiments: how do existing post-correction methods perform?

Cross validation

First pass OCR
Encoder-decoder

% Word Error Rate

Ainu  Gikko  Yakkha  Kwak'wala
Experiments: how do existing post-correction methods perform?

- First pass OCR
- Encoder-decoder

Lower is better!
Experiments: how do existing post-correction methods perform?

Lower is better!
Experiments: how do existing post-correction methods perform?

The baseline sometimes increases the error rate over the first pass.
Experiments: do the adaptations help low-resource learning?

Lower is better!
Experiments: do the adaptations help low-resource learning?

Lower is better!

% Word Error Rate

Ainu  Griko  Yakkha  Kwak'wala

- First pass OCR
- Encoder-decoder
- + Proposed Adaptations

17% reduction  52% reduction  34% reduction  38% reduction
Experiments: do the adaptations help low-resource learning?

Lower is better!

% Word Error Rate

Ainu  Grikö  Yakkha

First pass OCR
Encoder-decoder
+ Proposed Adaptations

17% reduction
52% reduction
34% reduction
38% reduction

Which adaptations were most helpful?

Ablations: all adaptations are useful, copy mechanism impacts performance the most
Improving performance without additional annotation
Improving performance without additional annotation

Very small number of manually transcribed pages

Supervised training

Current model
Improving performance without additional annotation

Very small number of manually transcribed pages
Improving performance without additional annotation

Very small number of manually transcribed pages

Relatively larger number of raw images that need to be digitized

Our dataset:
Documents contain 300 – 800 pages
Only ~30 are manually transcribed

Improving performance without additional annotation

Very small number of manually transcribed pages

Relatively larger number of raw images that need to be digitized

Improving performance without additional annotation

Very small number of manually transcribed pages

Relatively larger number of raw images that need to be digitized

Semi-supervised learning for efficient use of the unlabeled images

Self-training for OCR post-correction
Self-training for OCR post-correction

First pass OCR on unlabeled images
Self-training for OCR post-correction

First pass OCR on unlabeled images → Trained post-correction model
Self-training for OCR post-correction

First pass OCR on unlabeled images

Trained post-correction model

Previous best supervised model
Self-training for OCR post-correction

First pass OCR on unlabeled images → Trained post-correction model → Post-correction predictions
Self-training for OCR post-correction

1. First pass OCR on unlabeled images
2. Post-correction predictions
3. Retrain the model

Trained post-correction model
Self-training may introduce noise
Self-training may introduce noise

Can we bias post-correction towards generating correct words?
Self-training may introduce noise

Can we bias post-correction towards generating correct words?
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denema lāq. Wā, gīləmēsē

. . . Wā, gīləmēsē ŋnā

Wā, gīləmēsē lāg’alis lāx

...
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denema lāq. Wä, gîl mêsē
gîl mêsē gwālamasqēxs laē
... Wä, gîl mêsē ŋnā
Wä, gîl mêsē lāg'alis lāx
...
gîl mêsē lāg'aa lāqēxs laē
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denema lāq. Wā, g’il⁵mēsē

g’il⁵mēsē gwālamasqēxs laē

... Wā, g’il⁵mēsē zānā

Wā, g’il⁵mēsē lāg’alīs lāx

...

g’il⁵mēsē lāg’aa lāqēxs laē
Self-training may introduce noise

Can we bias post-correction towards generating correct words?
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denêma lāq. Wā, g’ilêmēsē

g’ilêmēsē gwālāmasqēxs laē

. . . Wā, g’ilêmēsē ḳnā

Wā, g’ilêmēsē lāq’alīs lāx

...
Self-training may introduce noise

Can we bias post-correction towards generating correct words?
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denēma lāq. Wä, g·îlᵉmēsē
g·îlᵉmēsē gwālamasqēxs lāē
... Wä, g·îlᵉmēsē ènā
Wä, g·îlᵉmēsē lāg·alēs lāx
...

✔️ g·îlᵉmēsē (7)
❌ g·îlᵉmēsē (5)
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

denema ːľaŋ. Wā, g·îľmēsē

g·îľmēsē ɣwālamasqēxs ːlāŋ

... Wā, g·îľmēsē ːnāŋ

Wā, g·îľmēsē lāg̱alīs ːlāx

... 

g·îľmēsē lāg̱aːa lāqēxs ːlāŋ

- g·îľmēsē (7)
- g·îľmēsē (5)
- g·îľmēsē (2)
Self-training may introduce noise

Can we bias post-correction towards generating correct words?
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

Different subsets of characters are incorrect

Empirical observations
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

Noise from self-training is typically inconsistent at the word-level

Empirical observations
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

Correct form of the word ends up being more frequent

Noise from self-training is typically inconsistent at the word-level

Empirical observations
Self-training may introduce noise

Can we bias post-correction towards generating correct words?

Can we use the word frequency information to bias the model towards correct forms?

Correct form of the word ends up being more frequent
Incorporating word frequency information
Incorporating word frequency information

\[ P(y) = p_{lstm}(y) \]
Incorporating word frequency information

\[ P(y) = p_{\text{lstm}}(y) \]
Incorporating word frequency information

\[ P(y) = p_{\text{lstm}}(y) \]

- Next character probability
- Decoder probability
Incorporating word frequency information

\[ P(y) = p_{\text{lstm}}(y) \]

\[ P_{\text{freq}} \]

Frequency-based probability to explicitly bias the model
Incorporating word frequency information

\[ P(y) = p_{\text{lstm}}(y) \]

How do we get probabilities based on word frequency?
Modeling word frequency
Modeling word frequency

Simple model for word frequency: count-based language model
Modeling word frequency

Simple model for word frequency: count-based language model

Predictions from self-training
Modeling word frequency

Simple model for word frequency: count-based language model

Train a smoothed unigram LM

Predictions from self-training
Simple model for word frequency: **count-based language model**

- **Modeling word frequency**
- Predictions from self-training
- Using counts of the word forms
- Train a smoothed unigram LM
Modeling word frequency

Simple model for word frequency: count-based language model

Train a smoothed unigram LM

Frequency-based word-level probabilities

Predictions from self-training
Modeling word frequency

Simple model for word frequency: **count-based language model**

- Predictions from self-training
- Train a smoothed unigram LM
- Frequency-based word-level probabilities
- Noisy weighted lexicon of the words in the predictions
Modeling word frequency

Simple model for word frequency: count-based language model

Predictions from self-training

Train a smoothed unigram LM

Frequency-based word-level probabilities
Lexically-aware decoding for post-correction
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \quad p_{\text{freq}} \]
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \]

\[ p_{\text{freq}} \]

We have frequency-based probabilities from the unigram LM!
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \]

But these are at the word-level: how we get character-level scores?
Scoring at the character-level

Example LM with two words:
- P(‘dog’) = 0.75
- P(‘door’) = 0.2
- P(<unk>) = 0.05
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- $P(\text{‘dog’}) = 0.75$
- $P(\text{‘door’}) = 0.2$
- $P(\text{<unk>}) = 0.05$
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- P('dog') = 0.75
- P('door') = 0.2
- P(<unk>) = 0.05

Set of states with weighted transitions
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- $P(‘\text{dog’}) = 0.75$
- $P(‘\text{door’}) = 0.2$
- $P(‘\text{unk’}) = 0.05$
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- $P(\text{‘dog’}) = 0.75$
- $P(\text{‘door’}) = 0.2$
- $P(\text{<unk>}) = 0.05$
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- $P(‘dog’) = 0.75$
- $P(‘door’) = 0.2$
- $P(<unk>) = 0.05$
Scoring at the character-level

Example LM with two words:
- $P('dog') = 0.75$
- $P('door') = 0.2$
- $P(<\text{unk}>) = 0.05$

**Weighted Finite State Automaton (WFSA)** representation of the LM

```
start

```

```
``
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- \( P(\text{‘dog’}) = 0.75 \)
- \( P(\text{‘door’}) = 0.2 \)
- \( P(\text{<unk>} ) = 0.05 \)
Scoring at the character-level

**Weighted Finite State Automaton (WFSA)** representation of the LM

Example LM with two words:
- \( P(\text{`dog'}) = 0.75 \)
- \( P(\text{`door'}) = 0.2 \)
- \( P(\text{`unk'}) = 0.05 \)

Probability from the count-based LM
Scoring at the character-level

Weighted Finite State Automaton (WFSA)
representation of the LM

Example LM with two words:
• $P(\text{`dog'}) = 0.75$
• $P(\text{`door'}) = 0.2$
• $P(\text{`unk'}) = 0.05$
Scoring at the character-level

Example LM with two words:
- $P(\text{‘dog’}) = 0.75$
- $P(\text{‘door’}) = 0.2$
- $P(\text{<unk>}) = 0.05$

Weighted Finite State Automaton (WFSA) representation of the LM

Probability from the count-based LM

Weight of path for “dog” = 0.75
Same as the word-level LM!
Scoring at the character-level

**Weighted Finite State Automaton (WFSA)** representation of the LM

Example LM with two words:
- $P(\text{`dog'} ) = 0.75$
- $P(\text{`door'} ) = 0.2$
- $P(\text{<unk>)} = 0.05$

Probability from the count-based LM
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- \( P(\text{‘dog’}) = 0.75 \)
- \( P(\text{‘door’}) = 0.2 \)
- \( P(\text{<unk>}) = 0.05 \)

Probability from the count-based LM
Scoring at the character-level

Example LM with two words:
- $P(\text{`dog'}) = 0.75$
- $P(\text{`door'}) = 0.2$
- $P(\text{<unk>}) = 0.05$

Weighted Finite State Automaton (WFSA) representation of the LM
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- $P('dog') = 0.75$
- $P('door') = 0.2$
- $P('<unk>') = 0.05$
Scoring at the character-level

Weighted Finite State Automaton (WFSA) representation of the LM

Example LM with two words:
- \( P('dog') = 0.75 \)
- \( P('door') = 0.2 \)
- \( P(<unk>) = 0.05 \)

![Diagram of WFSA](image-url)
Scoring at the character-level

Example LM with two words:
- $P('dog') = 0.75$
- $P('door') = 0.2$
- $P(<\text{unk}>) = 0.05$
Scoring at the character-level

Example LM with two words:
- \( P('dog') = 0.75 \)
- \( P('door') = 0.2 \)
- \( P(<unk>) = 0.05 \)

Output:

Output:

\[
\begin{align*}
\text{start} & \rightarrow \text{o} \rightarrow \text{g} \rightarrow \text{l} \\
\text{d} & \rightarrow \text{o} \rightarrow \text{o} \rightarrow \text{r} \rightarrow \text{2}
\end{align*}
\]
Scoring at the character-level

Output: door

Example LM with two words:
- $P('dog') = 0.75$
- $P('door') = 0.2$
- $P(<\text{unk}>) = 0.05$
Scoring at the character-level

Example LM with two words:

- $P('dog') = 0.75$
- $P('door') = 0.2$
- $P(<unk>) = 0.05$

Output: 

\[ \text{door} \]
Scoring at the character-level

Example LM with two words:
- P('dog') = 0.75
- P('door') = 0.2
- P(<unk>) = 0.05

Output: door
Scoring at the character-level

Example LM with two words:

• $P('dog') = 0.75$
• $P('door') = 0.2$
• $P(<unk>) = 0.05$

Output: 

```
d o o r
```
Scoring at the character-level

Example LM with two words:
- P('dog') = 0.75
- P('door') = 0.2
- P(<unk>) = 0.05

Output: door _ d
Scoring at the character-level

Example LM with two words:
- \( P('dog') = 0.75 \)
- \( P('door') = 0.2 \)
- \( P(<unk>) = 0.05 \)

Output: 
```
d o o r _ d
```
Scoring at the character-level

Example LM with two words:
- \( P('dog') = 0.75 \)
- \( P('door') = 0.2 \)
- \( P(<unk>) = 0.05 \)

Output: door _ d

Diagram:
- Start node with transitions to 'd' with probability 0.75
- 'd' transitions to 'o' with probability 0.2
- 'o' transitions to 'g'
- 'g' transitions to 'l'
- 'r' transitions to '2'
- Space and punctuation nodes

Diagram shows the probability transitions for the words 'dog' and 'door'.
Scoring at the character-level

Example LM with two words:
- $P(\text{\textquotesingle}\text{dog}\text{\textquotesingle}) = 0.75$
- $P(\text{\textquotesingle}\text{door}\text{\textquotesingle}) = 0.2$
- $P(\text{\textless unk\textgreater}) = 0.05$
Scoring at the character-level

Example LM with two words:

- \( P(\text{'dog'}) = 0.75 \)
- \( P(\text{'door'}) = 0.2 \)
- \( P(\text{<unk>}) = 0.05 \)
Scoring at the character-level

Example LM with two words:
- $P(\text{‘dog’}) = 0.75$
- $P(\text{‘door’}) = 0.2$
- $P(\text{<unk>}) = 0.05$
Scoring at the character-level

Example LM with two words:
- $P(‘dog’) = 0.75$
- $P(‘door’) = 0.2$
- $P(<\text{unk}>) = 0.05$

Character n-gram language model to score unknown sequences
Scoring at the character-level

Example LM with two words:
- $P(\text{"dog"}) = 0.75$
- $P(\text{"door"}) = 0.2$
- $P(\text{<unk>}) = 0.05$

Determinization and minimization for a compact and efficient representation

Character n-gram language model to score unknown sequences
Lexically-aware decoding for post-correction
Lexically-aware decoding for post-correction

\[ P(y) = p_{lstm}(y) \cdot p_{freq}(y) \]
Lexically-aware decoding for post-correction

\[ P(y) = \frac{p_{\text{lstm}}(y)}{p_{\text{freq}}(y)} \]
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \cdot p_{\text{freq}}(y) \]
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \cdot p_{\text{wfsa}}(y) \]
Lexically-aware decoding for post-correction

\[ P(y) = p_{\text{lstm}}(y) \]

WFSA representation gives character-level scores
Lexically-aware decoding for post-correction

Linear interpolation to combine the probabilities for joint inference

\[ P(y) = (1 - \lambda) \ast p_{\text{lstm}}(y) + \lambda \ast p_{\text{wfsa}}(y) \]
Lexically-aware decoding for post-correction

Linear interpolation to combine the probabilities for joint inference

\[ P(y) = (1 - \lambda) \cdot p_{\text{lstm}}(y) + \lambda \cdot p_{\text{wfsa}}(y) \]
Experiments: does self-training improve performance?
Experiments: does self-training improve performance?

- First pass
- Supervised model
- Self-training

% Word Error Rate

Ainu Grika Yakkha Kwak'wala
Experiments: does self-training improve performance?

- First pass
- Supervised model
- Self-training

% Word Error Rate

- Ainu
- Grikö
- Yakkha
- Kwak’wala
Experiments: does self-training improve performance?

- First pass
- Supervised model
- Self-training

% Word Error Rate

- Ainu
- Griko
- Yakkha
- Kwak'wala
Experiments: does self-training improve performance?

- First pass
- Supervised model
- Self-training

% Word Error Rate

- Ainu
- Griko
- Yakkha
- Kwak'wala
Experiments: does self-training improve performance?

- First pass
- Supervised model
- Self-training

- Ainu: 2.3% increase
- Griko: 14% reduction
- Yakkha: 15% reduction
- Kwak'wala: 13% reduction
Experiments: does self-training improve performance?

- Ainu: 2.3% increase
- Griko: 14% reduction
- Yakkha: 15% reduction
- Kwak'wala: 13% reduction
Experiments: does lexically-aware decoding counteract noise?

- First pass
- Supervised model
- Self-training
- + Lexically-aware decoding
Experiments: does lexically-aware decoding counteract noise?

- First pass
- Supervised model
- Self-training
- + Lexically-aware decoding

**Ainu**
- 15% reduction

**Griko**
- 15% reduction

**Yakkha**
- 22% reduction

**Kwak'wala**
- 18% reduction

Legend:
- Blue bar: First pass
- Green bar: Supervised model
- Blue bar with pattern: Self-training
- Red bar with pattern: + Lexically-aware decoding

**Graph:**
- Y-axis: % Word Error Rate
- X-axis: Languages (Ainu, Griko, Yakkha, Kwak'wala)
Experiments: does lexically-aware decoding counteract noise?

- First pass
- Supervised model
- Self-training
- + Lexically-aware decoding

<table>
<thead>
<tr>
<th>Language</th>
<th>First pass %</th>
<th>Supervised model %</th>
<th>Self-training %</th>
<th>+ Lexically-aware decoding %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ainu</td>
<td>15% reduction</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Griko</td>
<td>15% reduction</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yakkha</td>
<td></td>
<td>22% reduction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kwak'wala</td>
<td></td>
<td></td>
<td>18% reduction</td>
<td></td>
</tr>
</tbody>
</table>
Summary
Summary

- Thousands of languages do not have easily accessible text to build NLP models
Summary

- Thousands of languages do not have easily accessible text to build NLP models
  - Text data does exist in many of these languages!
Summary

• Thousands of languages do not have easily accessible text to build NLP models
  • Text data does exist in many of these languages!
  • Locked away in non-machine-readable formats like printed books
Summary

- Thousands of languages do not have easily accessible text to build NLP models
  - Text data does exist in many of these languages!
  - Locked away in non-machine-readable formats like printed books

- OCR post-correction improves text extraction in very low-resourced settings
Summary

- Thousands of languages do not have easily accessible text to build NLP models
  - Text data does exist in many of these languages!
  - Locked away in non-machine-readable formats like printed books

- OCR post-correction improves text extraction in very low-resourced settings

Multi-source model: ↓ WER 17% – 52%
Summary

- Thousands of languages do not have easily accessible text to build NLP models
  - Text data does exist in many of these languages!
  - Locked away in non-machine-readable formats like printed books

- OCR post-correction improves text extraction in very low-resourced settings

<table>
<thead>
<tr>
<th>Multi-source model:</th>
<th>↓ WER 17% – 52%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-supervised with lexically-aware decoding:</td>
<td>↓ WER 29% – 59%</td>
</tr>
</tbody>
</table>
Impact case study: Kwak’wala
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from
Impact case study: Kwak’wala

- Collaborating with documentary linguists and Kwak’wala speakers
- Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

Produced by Franz Boas in 1921
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

• Tremendous cultural and linguistic value!
Impact case study: Kwak’wala

- Collaborating with documentary linguists and Kwak’wala speakers
- Identify documents that would be most useful to extract text from

**Boas texts:** 10 volumes of Kwak’wala language and community documentation

- Tremendous cultural and linguistic value!
- Minimally accessible to researchers
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from

**Boas texts:** 10 volumes of Kwak’wala language and community documentation

- Tremendous cultural and linguistic value!
- Minimally accessible to researchers
- Manual search in scanned images
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

• Tremendous cultural and linguistic value!
• Minimally accessible to researchers
• Manual search in scanned images
• Legacy orthography that is hard to read
Impact case study: Kwak’wala

- Collaborating with documentary linguists and Kwak’wala speakers
- Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation
Impact case study: Kwak’wala

- Collaborating with documentary linguists and Kwak’wala speakers
  - Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

- 1500+ pages converted to a machine-readable format
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

• 1500+ pages converted to a machine-readable format
• Searchable!
Impact case study: Kwak’wala

• Collaborating with documentary linguists and Kwak’wala speakers
  • Identify documents that would be most useful to extract text from

Boas texts: 10 volumes of Kwak’wala language and community documentation

• 1500+ pages converted to a machine-readable format
  • Searchable!
  • Legacy orthography can be automatically transliterated to modern writing systems
Impact case study: Kwak’wala

- 1500+ pages converted to a machine-readable format
- Searchable!
- Legacy orthography can be automatically transliterated to modern writing systems
Impact and applications: beyond this talk
Impact and applications: beyond this talk

Our software is open-source and has been used on many other languages!
Impact and applications: beyond this talk

Our software is open-source and has been used on many other languages!

- Bhutia
- Sanskrit
- Quechua
- Igbo
- Tibetan
- Piaroa
- Secwepemctsín
- Pintupi-Luritja

Image credits: Ijemma Onwuzulike, Jorge Labrada, Ben Foley
Impact and applications: beyond this talk

Our software is open-source and has been used on many other languages!

Bhutia
Sanskrit
Quechua
Igbo
Tibetan
Piaroa
Secwepemctsin
Pintupi-Luritja

Extracting text to train machine translation for Pintupi-Luritja

Image credits: Ijemma Onwuzulike, Jorge Labrada, Ben Foley
Our software is open-source and has been used on many other languages!

- Bhutia
- Sanskrit
- Quechua
- Igbo
- Tibetan
- Piaroa
- Secwepemcstín
- Pintupi-Luritja

Automatic extraction of handwritten speech transcriptions in Piaroa

Image credits: Ijemna Onwuzulike, Jorge Labrada, Ben Foley
Impact and applications: beyond this talk

Our software is open-source and has been used on many other languages!

Bhutia
Sanskrit
Quechua
Igbo
Tibetan
Piaroa
Secwepemctsin
Pintupi-Luritja

Print dictionaries in Igbo are high-coverage, but not digitized

Image credits: Ijemma Onwuzulike, Jorge Labrada, Ben Foley
Unlocking Bi-lingual Lexicons

Xinyi Wang, Sebastian Ruder, Graham Neubig.
Expanding Pretrained Models to Thousands More Languages via Lexicon-based Adaptation.
ACL 2022.
Multilingual Pretrained Models
Multilingual Pretrained Models

Pretrained Model → Finetune → Inference

How to adapt the model for the language T?
Adaptation: Monolingual Data

- e.g. Continued Masked Language Modeling (MLM) using monolingual data in the target language $T$
Adaptation: Parallel Data

- e.g. Parallel Data: use best NMT system available to translate English task data into the target language $T$
Languages without Conventional Data

- 77% Other
- 23% Bible
- 4% Wikipedia/CommonCrawl
- 1% Covered by mBERT
Languages without Conventional Data

- Majority of languages in the world cannot benefit from progress in NLP due to lack of data
Two Groups of Low-resource Languages
Two Groups of Low-resource Languages

- Majority of World’s languages cannot benefit from progress in NLP (Joshi et al. 2020)
Two Groups of Low-resource Languages

- Majority of World’s languages cannot benefit from progress in NLP (Joshi et al. 2020)
  - No-Text: virtually no resource
Two Groups of Low-resource Languages

- Majority of World’s languages cannot benefit from progress in NLP (Joshi et al. 2020)
  - No-Text: virtually no resource
  - Few-Text: very limited resource
Alternative Data Source

https://panlex.org/
Alternative Data Source

- Linguists have been documenting languages for years in formats such as lexicons

https://panlex.org/
Linguists have been documenting languages for years in formats such as lexicons.

PanLex: open-sourced database of lexicons with much better language coverage.

Current database coverage:

- 2,500 dictionaries
- 5,700 languages
Alternative Data Source

Linguists have been documenting languages for years in formats such as lexicons.

PanLex: open-sourced database of lexicons with much better language coverage.

Current database coverage:
- 2,500 dictionaries
- 5,700 languages

Coverage by source:
- 77% Bible
- 23% Other
- 4% Wikipedia/CommonCrawl
- 1% Covered by mBERT

https://panlex.org/
Synthesizing Data Using Lexicons
Synthesizing Data Using Lexicons

- **Pseudo Mono Data**: replace words in *English monolingual data* to its corresponding translation in the target language $T$
Synthesizing Data Using Lexicons

- **Pseudo Mono Data:** replace words in *English monolingual data* to its corresponding translation in the target language T

Anarchism calls for the abolition of the state, which it holds to be undesirable, unnecessary, and harmful.

Anarchism calls *ghal il abolition ta’ il stat*, *lima hi holds ghal tkun undesirable*, *bla bzonn*, and *u harmful*. 

---

<table>
<thead>
<tr>
<th>Eng Mono</th>
<th>Anarchism calls for the abolition of the state, which it holds to be undesirable, unnecessary, and harmful.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pseudo Mono</td>
<td>Anarchism calls <em>ghal il abolition ta’ il stat</em>, <em>lima hi holds ghal tkun undesirable</em>, <em>bla bzonn</em>, and <em>u harmful</em>.</td>
</tr>
</tbody>
</table>
Synthesizing Data Using Lexicons
Synthesizing Data Using Lexicons

- **Pseudo Task Data**: replace words in **English task data** to its corresponding translation in the target language $T$
Synthesizing Data Using Lexicons

- **Pseudo Task Data**: replace words in **English task data** to its corresponding translation in the target language $T$.
Synthesizing Data Using Lexicons

Pretrained Model → Pseudo MLM → Pseudo Fine-tune

T Mono → Pseudo Mono → Eng Mono → Lexicon → Eng Task → Pseudo Task
Synthesizing Data Using Lexicons

- Use either pseudo MLM or Pseudo Fine-tune, or both
Experiments
Experiments

• Model: mBERT
Experiments

• Model: mBERT

• Tasks:
  • NER
  • POS tagging
  • Dependency Parsing
Experiments

- Model: mBERT
- Tasks:
  - NER
  - POS tagging
  - Dependency Parsing
- Languages: 19 languages not covered by mBERT pretraining
Results
Results

F1 gain with synthetic data

NER  POS  Parsing

No-text  Few-text
Results

F1 gain with synthetic data

- **NER**
  - No-text: 4
  - Few-text: 0

- **POS**
  - No-text: 16
  - Few-text: 0

- **Parsing**
  - No-text: 8
  - Few-text: 0
Results

F1 gain with synthetic data

- NER
- POS
- Parsing

No-text

Few-text
Results

- Using synthetic data leads to significant improvements for both no-text and few-text setting
I suspect the streets of Baghdad will look as if a war is looming this week.

Pseudo Task: jien iddubita il streets ta’ Bagdad xewqa hares kif jekk a gwerra is looming dan gimgha.
Label Noise

Eng Task: I suspect the streets of Baghdad will look as if a war is looming this week.

Pseudo Task: jien iddubita il streets ta’ Bagdad xewqa hares kif jekk a gwerra is looming dan gimgha.

- “xewqa” is a noun meaning “desire, will”
Label Noise

<table>
<thead>
<tr>
<th>Eng Task</th>
<th>Pseudo Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>I suspect the streets of Baghdad <strong>will</strong> look as if a war is looming this week.</td>
<td>jien iddubita il streets ta’ Bagdad <strong>xewqa</strong> hares kif jekk a gwerra is looming dan gimgha.</td>
</tr>
<tr>
<td>PRON VERB DET NOUN ADP PROPN AUX VERB SCONJ SCONJ DET NOUN AUX VERB DET NOUN PUNCT</td>
<td>PRON VERB DET NOUN ADP PROPN AUX VERB SCONJ SCONJ DET NOUN AUX VERB DET NOUN PUNCT</td>
</tr>
</tbody>
</table>

- **“xewqa”** is a noun meaning “desire, will”
- But the original English POS tag is inconsistent with the replaced word
Label Noise

• Use the fine-tuned model to “correct” the labels for the Pseudo task data
• Use the fine-tuned model to “correct” the labels for the Pseudo task data
• Use the fine-tuned model to “correct” the labels for the Pseudo task data
Label Noise

<table>
<thead>
<tr>
<th>Eng Task</th>
<th>I suspect the streets of Baghdad <strong>will</strong> look as if a war is looming this week.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pseudo Task</td>
<td>jien iddubita il streets ta' Bagdad <strong>xewqa</strong> hares kif jekk a gwerra is looming dan gimgha.</td>
</tr>
</tbody>
</table>

- “**xewqa**” is a noun meaning “desire, will”
I suspect the streets of Baghdad will look as if a war is looming this week.

“xewqa” is a noun meaning “desire, will”
Label Noise

• “xewqa” is a noun meaning “desire, will”
• The model is able to assign the correct label of noun
Label Noise
Label Noise

No-text

Few-text

NER

POS

Parsing
Label Noise

F1 gain by Label Distillation

- **No-text**
- **Few-text**

<table>
<thead>
<tr>
<th>Task</th>
<th>No-text</th>
<th>Few-text</th>
</tr>
</thead>
<tbody>
<tr>
<td>NER</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>POS</td>
<td>1.50</td>
<td>0.00</td>
</tr>
<tr>
<td>Parsing</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Label Noise

- Label Distillation is especially helpful for syntactic tasks
Comparison to Few-shot Learning
Comparison to Few-shot Learning

F1 gain over mBERT on NER

-7.5 0 7.5 15 22.5 30

hau wol lug ibo

- Best Adapted
- 10-shot
- 100-shot
- Best Adapted+100-shot
Comparison to Few-shot Learning

-7.5  0  7.5  15  22.5  30

F1 gain over mBERT on NER

hau  wol  lug  ibo

Best Adapted  10-shot  100-shot  Best Adapted+100-shot
Comparison to Few-shot Learning

F1 gain over mBERT on NER

- hau
- wol
- lug
- ibo

-7.5 0 7.5 15 22.5 30

- Best Adapted
- 10-shot
- 100-shot
- Best Adapted+100-shot
Comparison to Few-shot Learning

F1 gain over mBERT on NER

-7.5 0 7.5 15 22.5 30

hau  wol  lug  ibo

Best Adapted  10-shot  100-shot  Best Adapted+100-shot
Comparison to Few-shot Learning

- Few-shot learning needs more annotated data for languages with limited text
Comparison to Few-shot Learning

- Few-shot learning needs more annotated data for languages with limited text

![Bar Chart]

- Best Adapted
- 10-shot
- 100-shot
- Best Adapted+100-shot
Comparison to Few-shot Learning

- Few-shot learning needs more annotated data for languages with limited text
- Combining adaptation and few-shot doesn’t bring consistent improvements
Conclusion
Conclusion
Conclusion

- Methods to **unlock new resources** for human or machine use in under-resourced languages
Conclusion

- Methods to **unlock new resources** for human or machine use in under-resourced languages
- What’s next?
Conclusion

- Methods to **unlock new resources** for human or machine use in under-resourced languages
- What’s next?
- Should we **put some linguistics in the models?**
Conclusion

- Methods to **unlock new resources** for human or machine use in under-resourced languages
- What’s next?
- Should we **put some linguistics in the models**?
  - Morphologically aware soft constraints for OCR?
Conclusion

• Methods to **unlock new resources** for human or machine use in under-resourced languages

• What’s next?

• Should we put **some linguistics in the models**?
  • Morphologically aware soft constraints for OCR?
  • Morphologically/syntactically aware data synthesis using lexicons?
Conclusion

• Methods to **unlock new resources** for human or machine use in under-resourced languages
• What’s next?
• Should we **put some linguistics in the models**?
  • Morphologically aware soft constraints for OCR?
  • Morphologically/syntactically aware data synthesis using lexicons?
• Should we **use the models in language learning or linguistics**?
Conclusion

• Methods to **unlock new resources** for human or machine use in under-resourced languages
• What’s next?
• Should we **put some linguistics in the models?**
  • Morphologically aware soft constraints for OCR?
  • Morphologically/syntactically aware data synthesis using lexicons?
• Should we **use the models in language learning or linguistics?**
  • Large-scale extraction of text or inter-linear glosses for use in developing language materials?