Intro to Flair: Open Source NLP Framework

Alan Akbik
Zalando Research

Berlin ML Meetup, December 2018
TEXT DATA IN FASHION
TEXT DATA IN FASHION

[...] How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. [...]
TEXT DATA IN FASHION

[...] How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. [...]
How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. [...]
[...] How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. [...]
DOCUMENT CLASSIFICATION

Document 1

[...] quick delivery as always. Thank you very much! [...] 

Document 2

[...] waited for three days until the package finally arrived! [...] 

Task: automatically categorize your documents into one or more classes
DOCUMENT CLASSIFICATION

Task: automatically categorize your documents into one or more classes

Document 1

[...] quick delivery as always. Thank you very much! [...]  

Classes

DELIVERY-FAST

Document 2

[...] waited for three days until the package finally arrived! [...]  

Classes

DELIVERY-SLOW

Classes:

- Document Classification
- Fashion NER
- NamedLocation
- NamedPerson
- NominalProduct
- NamedProduct
- Look
- NamedEvent
Task: automatically categorize your documents into one or more classes
How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. For my evening shoot in downtown Winnipeg with Christa Wong, I chose all of my current wardrobe favourites, including Dior-inspired pumps from Zara and marble statement earrings from Olive + Piper. [...]

Fashion Entity Types
How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. For my evening shoot in downtown Winnipeg with Christa Wong, I chose all of my current wardrobe favourites, including Dior-inspired pumps from Zara and marble statement earrings from Olive + Piper. [...]

**Fashion Entity Types**

**NamedLocation**
How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. For my evening shoot in downtown Winnipeg with Christa Wong, I chose all of my current wardrobe favourites, including Dior-inspired pumps from Zara and marble statement earrings from Olive + Piper. [...]

Fashion Entity Types

NamedLocation    NominalProduct
How I style the basics that fill my wardrobe changes from season to season. And city to city, too, come to think of it. In Berlin, I paired this dress with a moto jacket and ankle boots, while in Paris, I added an oversized hat and classic pumps. For my evening shoot in downtown Winnipeg with Christa Wong, I chose all of my current wardrobe favourites, including Dior-inspired pumps from Zara and marble statement earrings from Olive + Piper. [...]

Fashion Entity Types

- NamedLocation
- NominalProduct
- NamedPerson
- NamedOrganizationRetailer
FLAIR FRAMEWORK

a very simple framework for state-of-the-art natural language processing (NLP)
FLAIR FRAMEWORK

a very simple framework for state-of-the-art natural language processing (NLP)

- current state-of-the-art across many NLP tasks
FLAIR FRAMEWORK

a very simple framework for state-of-the-art natural language processing (NLP)

- current state-of-the-art across many NLP tasks
- very simple to use
THIS TALK
THIS TALK

1. New type of word embeddings
1. New type of word embeddings

2. New state-of-the-art scores across sequence labeling tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Our approach</th>
<th>Previous best</th>
</tr>
</thead>
<tbody>
<tr>
<td>NER English</td>
<td><strong>93.09 ± 0.12</strong></td>
<td>92.22±0.1 (Peters et al., 2018)</td>
</tr>
<tr>
<td>NER German</td>
<td><strong>88.32 ± 0.2</strong></td>
<td>78.76 (Lample et al., 2016)</td>
</tr>
<tr>
<td>Chunking</td>
<td><strong>96.72 ± 0.05</strong></td>
<td>96.37±0.05 (Peters et al., 2017)</td>
</tr>
<tr>
<td>PoS tagging</td>
<td><strong>97.85 ± 0.01</strong></td>
<td>97.64 (Choi, 2016)</td>
</tr>
</tbody>
</table>
THIS TALK

1. New type of word embeddings

2. New state-of-the-art scores across sequence labeling tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Our approach</th>
<th>Previous best</th>
</tr>
</thead>
<tbody>
<tr>
<td>NER English</td>
<td>93.09 ± 0.12</td>
<td>92.22±0.1 (Peters et al., 2018)</td>
</tr>
<tr>
<td>NER German</td>
<td>88.32 ± 0.2</td>
<td>78.76 (Lample et al., 2016)</td>
</tr>
<tr>
<td>Chunking</td>
<td>96.72 ± 0.05</td>
<td>96.37±0.05 (Peters et al., 2017)</td>
</tr>
<tr>
<td>PoS tagging</td>
<td>97.85 ± 0.01</td>
<td>97.64 (Choi, 2016)</td>
</tr>
</tbody>
</table>

3. Introduce Flair framework
Talk Outline

Overview

Flair Embeddings

Limitations of classic word embeddings
Character-level neural language models
Comparative evaluation

Flair Framework

Usage Example
Talk Outline

Overview

Flair Embeddings

Limitations of classic word embeddings
Character-level neural language models
Comparative evaluation

Flair Framework

Usage Example
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.
WORD EMBEDDINGS

Problem: Words are just strings

Vocabulary:
Man, woman, boy, girl, prince, princess, queen, king, monarch

Each word gets a 1x9 vector representation
WORD EMBEDDINGS

Try to build a lower dimensional embedding

Vocabulary:
Man, woman, boy, girl, prince, princess, queen, king, monarch
WORD EMBEDDINGS

Try to build a lower dimensional embedding

Vocabulary:
Man, woman, boy, girl, prince, princess, queen, king, monarch

<table>
<thead>
<tr>
<th></th>
<th>Femininity</th>
<th>Youth</th>
<th>Royalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Man</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Woman</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Boy</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Girl</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Prince</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Princess</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Queen</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>King</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Monarch</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
</tr>
</tbody>
</table>

Each word gets a 1x3 vector
Similar words... similar vectors

@share_a_lynn | @TeamEdgeTier
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary

Problem 1: Word ambiguity
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - …
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - ...

- Classic word embeddings conflate all meanings into single vector
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - …

- Classic word embeddings conflate all meanings into single vector

- *Contextualized* embeddings?
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - …

- Classic word embeddings conflate all meanings into single vector

Problem 2: Fixed vocabulary

- Contextualized embeddings?
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - …

- Classic word embeddings conflate all meanings into single vector

- Contextualized embeddings?

Problem 2: Fixed vocabulary

- What is a word? Tokenizer decides?
  - “48-year-old”
  - “Hotelzimmer” (hotel room)
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.

Problem 1: Word ambiguity

- “Washington”
  - Last name
  - State / city
  - Sports team
  - …

- Classic word embeddings conflate all meanings into single vector

- *Contextualized* embeddings?

Problem 2: Fixed vocabulary

- What is a word? Tokenizer decides?
  - “48-year-old”
  - “Hotelzimmer” (*hotel room*)

- Long-tailed distribution of words
  - Rare words?
  - Out of vocabulary words?
  - “coooooool”
WORD EMBEDDINGS

Classic word embeddings learn a vector representation for each word in a fixed vocabulary.

Problem 1: Word ambiguity

- "Washington"
  - Last name
  - State / city
  - Sports team
  - …

- Classic word embeddings conflate all meanings into single vector

- Contextualized embeddings?

Problem 2: Fixed vocabulary

- What is a word? Tokenizer decides?
  - “48-year-old”
  - “Hotelzimmer” (hotel room)

- Long-tailed distribution of words
  - Rare words?
  - Out of vocabulary words?
  - “coooooooool”

- Meaningful embeddings for any word?
CONTEXTUAL STRING EMBEDDINGS

We propose **contextual string embeddings** that are:

- *Contextualized* by their usage in text
CONTEXTUAL STRING EMBEDDINGS

We propose contextual string embeddings that are:

- Contextualized by their usage in text
- Fundamentally model words as strings of characters
We propose **contextual string embeddings** that are:

- *Contextualized* by their usage in text
- Fundamentally model words as *strings of characters*
- Pre-trained on very large corpora
CONTEXTUAL STRING EMBEDDINGS

We propose **contextual string embeddings** that are:

- *Contextualized* by their usage in text
- Fundamentally model words as *strings of characters*
- Pre-trained on very large corpora

We produce these embeddings using **neural character-level language modeling**
Language modeling:

- Train recurrent neural network (RNN) to predict the next word in a sequence of words
NEURAL LANGUAGE MODELING

Language modeling:
● Train recurrent neural network (RNN) to predict the next word in a sequence of words

Character-level language modeling:
● Train RNN to predict the next character in a sequence of characters
● No tokenization
● Small vocabulary
NEURAL LANGUAGE MODELING

*what is the next word?*

because it was hungry, the cat ___
NEURAL LANGUAGE MODELING

what is the next word?

because it was hungry, the cat ____ ate
NEURAL LANGUAGE MODELING

what is the next word?

because it was hungry, the cat ___ \textbf{ate}

what is the next word?

because it was hungry, the cat ate ____
NEURAL LANGUAGE MODELING

what is the next word?

because it was hungry, the cat ___ ate

what is the next word?

because it was hungry, the cat ate ____ the

what is the next word?

because it was hungry, the cat ate the ____
NEURAL LANGUAGE MODELING

what is the next word?
because it was hungry, the cat ___ ate

what is the next word?
because it was hungry, the cat ate ____ the

what is the next word?
because it was hungry, the cat ate the ____

The model learns

- Shallow syntax
  - nouns, verbs, adjectives
  - tense, number

- Sentence-level syntax
  - constituents
  - subordinate clauses
  - punctuation, capitalization

- Shallow semantics
  - sentiment
  - topic
WHAT DOES THIS NEURAL LANGUAGE MODEL KNOW?

We can sample the LM to **generate text**:

(1) According to a giant external film crew, the visible food contained "weirdness or unknown" firestorms.

(2) According to ADA attorney Stacy Baileil, prosecutors have requested that all of the county bend to him rather than require him to accept the legal fees.

(3) Iran's Deputy Marine Ministry inspector general last week criticised security forces for testing changes in a military base when attackers began putting metal plates in, he said.
INTERNAL LM REPRESENTATIONS

Model represents syntactic and semantic properties!
(Radfort et. al, 2017)
PROPOSED APPROACH
PROPOSED APPROACH
PROPOSED APPROACH

- Pass sentence as sequence of characters into two character-level language models
- Retrieve the internal states before first and after last character for each word
- Combine forward and backward states to form embedding
TRANSFER LEARNING

Supervised task

Unsupervised task
COMPARATIVE EVALUATION

Tasks:

- CoNLL-03 Named Entity Recognition for English and German
- CoNLL-2000 Chunking
- WSJ Part-of-Speech Tagging
COMPARATIVE EVALUATION

Tasks:
● CoNLL-03 Named Entity Recognition for English and German
● CoNLL-2000 Chunking
● WSJ Part-of-Speech Tagging

Setup:
● BiLSTM-CRF architecture (Huang et. al, 2015)
  ○ Only classic word embeddings (Huang et. al, 2015)
  ○ Word and character embeddings (Lample et. al, 2016)
  ○ ELMo embeddings (Peters et. al, 2017; 2018)
## RESULTS

<table>
<thead>
<tr>
<th>Approach</th>
<th>NER-English F1-score</th>
<th>NER-German F1-score</th>
<th>Chunking F1-score</th>
<th>POS Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>proposed</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROPOSED</td>
<td>91.97±0.04</td>
<td>85.78 ± 0.18</td>
<td>96.68±0.03</td>
<td>97.73±0.02</td>
</tr>
<tr>
<td>PROPOSED+WORD</td>
<td>93.07±0.10</td>
<td>88.20 ± 0.21</td>
<td>96.70±0.04</td>
<td>97.82±0.02</td>
</tr>
<tr>
<td>PROPOSED+CHAR</td>
<td>91.92±0.03</td>
<td>85.88 ± 0.20</td>
<td><strong>96.72±0.05</strong></td>
<td>97.8±0.01</td>
</tr>
<tr>
<td>PROPOSED+WORD+CHAR</td>
<td><strong>93.09±0.12</strong></td>
<td><strong>88.32 ± 0.20</strong></td>
<td>96.71±0.07</td>
<td>97.76±0.01</td>
</tr>
<tr>
<td>PROPOSED+ALL</td>
<td>92.72±0.09</td>
<td>n/a</td>
<td>96.65±0.05</td>
<td><strong>97.85±0.01</strong></td>
</tr>
<tr>
<td><strong>baselines</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HUANG</td>
<td>88.54±0.08</td>
<td>82.32 ± 0.35</td>
<td>95.4±0.08</td>
<td>96.94±0.02</td>
</tr>
<tr>
<td>LAMPLE</td>
<td>89.3±0.23</td>
<td>83.78 ± 0.39</td>
<td>95.34±0.06</td>
<td>97.02±0.03</td>
</tr>
<tr>
<td>PETERS</td>
<td>92.34±0.09</td>
<td>n/a</td>
<td>96.69±0.05</td>
<td>97.81±0.02</td>
</tr>
<tr>
<td><strong>best published</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Peters et al., 2018)</td>
<td>92.22±0.10</td>
<td>78.76</td>
<td>96.37±0.05</td>
<td>97.64</td>
</tr>
<tr>
<td>(Peters et al., 2017)</td>
<td>91.93±0.19</td>
<td>77.20</td>
<td>95.96±0.08</td>
<td>97.55</td>
</tr>
<tr>
<td>(Seyler et al., 2017)</td>
<td>91.71±0.10</td>
<td>76.22</td>
<td>95.77</td>
<td>97.53±0.03</td>
</tr>
<tr>
<td>(Liu et al., 2017)</td>
<td>91.21</td>
<td>75.72</td>
<td>95.56</td>
<td>97.30</td>
</tr>
<tr>
<td>(Lample et al., 2016)</td>
<td>(Qi et al., 2009)</td>
<td>Søgaard et al. (2016)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
EVALUATION RESULTS

Takeaways [1]:

- Combination of Contextual String Embeddings and Classic Word Embeddings consistently gives us state-of-the-art results

EVALUATION RESULTS

Takeaways [1]:

- Combination of Contextual String Embeddings and Classic Word Embeddings consistently gives us state-of-the-art results
- Task-trained character-level features not necessary

EVALUATION RESULTS

Takeaways [1]:

● Combination of Contextual String Embeddings and Classic Word Embeddings consistently gives us state-of-the-art results

● Task-trained character-level features not necessary

● Character-level LM embeddings match or outperform word-level LM embeddings (ELMo)

EVALUATION RESULTS

Takeaways [1]:

● Combination of Contextual String Embeddings and Classic Word Embeddings consistently gives us state-of-the-art results

● Task-trained character-level features not necessary

● Character-level LM embeddings match or outperform word-level LM embeddings (ELMo)

● Also state-of-the-art for Polish [2]


Talk Outline

Overview

Contextual String Embeddings

Limitations of classic word embeddings

Character-level neural language models

Sequence Labeling Experiments

Baselines and experimental setup

Results of comparative evaluation
OPEN SOURCE RELEASE

**flair** - a very simple framework for state-of-the-art NLP

```
pip install flair
```

Flair is:

- A Python library installable through pip
- Built on Pytorch
- Currently at version 0.3.2

Use Flair to:

- Apply our pre-trained taggers on your text
- Train your own NLP models
from flair.data import Sentence
from flair.models import SequenceTagger

# make a sentence
sentence = Sentence('I love Berlin .

# load the NER tagger
tagger = SequenceTagger.load('ner')

# run NER over sentence
tagger.predict(sentence)

print(sentence)
Sentence: "I love Berlin ." - 4 Tokens

print(sentence.to_tagged_string())
I love Berlin <S-LOC> .
SPAN ANNOTATIONS

# make a sentence
sentence = Sentence('George Washington was born in Washington .')

# run NER over sentence
tagger.predict(sentence)

for entity in sentence.get_spans('ner'):
    print(entity)

PER-span [1,2]: "George Washington"
LOC-span [5]: "Washington"
from flair.embeddings import WordEmbeddings

# init embedding
glove_embedding = WordEmbeddings('glove')

# create sentence.
sentence = Sentence('The grass is green .')

# embed a sentence using glove.
glove_embedding.embed(sentence)
# contextual string embeddings
flair_embedding = FlairEmbeddings('news-forward')

# ELMo embeddings (Peters et. al, 2018)
elmo_embedding = ELMoEmbeddings('medium')

# Google’s BERT embeddings (Devlin et. al, 2018)
bert_embedding = BertEmbeddings('large-uncased')

# stacked embeddings
embedding = StackedEmbeddings([flair_embedding, elmo_embedding, bert_embedding])
TRAIN YOUR OWN MODELS

Data fetchers
- Automatically download publicly available NLP datasets
- Data readers for common NLP formats

Model trainer
- Training mechanisms: annealing, checkpointing, restarts, etc.
- Automatic hyperparameter selection

Tutorials online to get you started
JOIN THE TEAM!

flair is on github

Use it

● Install through pip or clone

Help develop it

● Growing numbers of contributors
● New features / bug fixes / languages
● Frequent releases
THANK YOU!

Questions?

(BTW: we’re hiring!)

zalando research
Backup Slides
ZALANDO AT A GLANCE

~4.4 billion EURO net sales 2017

~16,000 employees in Europe
>50% return rate across all categories

~214 million visits per month

~24 million active customers
~300,000 product choices
>2,000 brands
17 countries

>2,000 brands
17 countries
TRAINING CHARACTER LANGUAGE MODELS

Hidden states, layers

- 1 GPU, 1 week

ELMo model:

- 32 GPUs, 5 weeks
### QUALITATIVE INSPECTION

<table>
<thead>
<tr>
<th>word</th>
<th>context</th>
<th>selected nearest neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Washington</td>
<td>(a) <em>Washington</em> to curb support for ...</td>
<td>1) <em>Washington</em> would also take ... action ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2) <em>Russia</em> to clamp down on barter deals ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3) <em>Brazil</em> to use hovercrafts for ...</td>
</tr>
<tr>
<td>Washington</td>
<td>(b) [...] Anthony <em>Washington</em> (U.S.) ...</td>
<td>1) [...] Carla Sacramento (Portugal) ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2) [...] Charles Austin (U.S.) ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3) [...] Steve Backley (Britain) ...</td>
</tr>
<tr>
<td>Washington</td>
<td>(c) [...] flown to <em>Washington</em> for ...</td>
<td>1) [...] while visiting <em>Washington</em> to ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2) [...] journey to New York City and <em>Washington</em> ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14) [...] lives in <em>Chicago</em> ...</td>
</tr>
<tr>
<td>Washington</td>
<td>(d) [...] when <em>Washington</em> came charging back ...</td>
<td>1) [...] point for victory when <em>Washington</em> found ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4) [...] before <em>England</em> struck back with ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6) [...] before <em>Ethiopia</em> won the spot kick decider ...</td>
</tr>
<tr>
<td>Washington</td>
<td>(e) [...] said <em>Washington</em> ...</td>
<td>1) [...] subdue the never-say-die <em>Washington</em> ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4) [...] a private school in <em>Washington</em> ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9) [...] said <em>Florida</em> manager John Boles ...</td>
</tr>
</tbody>
</table>
## DIRECT PROJECTION

<table>
<thead>
<tr>
<th>Embedding + Architecture</th>
<th>NER-English F1-score</th>
<th>NER-German F1-score</th>
<th>Chunking F1-score</th>
<th>POS Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PROPOSED+WORD</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+BiLSTM-CRF</td>
<td>93.07 ± 0.10</td>
<td>88.20 ± 0.21</td>
<td>96.70 ± 0.04</td>
<td>97.82 ± 0.02</td>
</tr>
<tr>
<td>+Map-CRF</td>
<td>90.17 ± 0.06</td>
<td>85.17 ± 0.04</td>
<td>96.05 ± 0.04</td>
<td>97.62 ± 0.01</td>
</tr>
<tr>
<td>+Map</td>
<td>79.86 ± 0.12</td>
<td>76.97 ± 0.16</td>
<td>90.55 ± 0.05</td>
<td>97.35 ± 0.01</td>
</tr>
<tr>
<td><strong>PROPOSED</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+BiLSTM-CRF</td>
<td>91.97 ± 0.04</td>
<td>85.78 ± 0.18</td>
<td>96.68 ± 0.03</td>
<td>97.73 ± 0.02</td>
</tr>
<tr>
<td>+Map-CRF</td>
<td>88.62 ± 0.15</td>
<td>82.27 ± 0.22</td>
<td>95.96 ± 0.05</td>
<td>97.53 ± 0.02</td>
</tr>
<tr>
<td>+Map</td>
<td>81.42 ± 0.16</td>
<td>73.90 ± 0.09</td>
<td>90.50 ± 0.06</td>
<td>97.26 ± 0.01</td>
</tr>
<tr>
<td><strong>CLASSIC WORD EMBEDDINGS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+BiLSTM-CRF</td>
<td>88.54 ± 0.08</td>
<td>82.32 ± 0.35</td>
<td>95.40 ± 0.08</td>
<td>96.94 ± 0.02</td>
</tr>
<tr>
<td>+Map-CRF</td>
<td>66.53 ± 0.03</td>
<td>72.69 ± 0.12</td>
<td>91.26 ± 0.04</td>
<td>94.06 ± 0.02</td>
</tr>
<tr>
<td>+Map</td>
<td>48.79 ± 0.27</td>
<td>57.43 ± 0.12</td>
<td>65.01 ± 0.50</td>
<td>89.58 ± 0.02</td>
</tr>
</tbody>
</table>
DISCLAIMER

This presentation and its contents are strictly confidential. It may not, in whole or in part, be reproduced, redistributed, published or passed on to any other person by the recipient.

The information in this presentation has not been independently verified. No representation or warranty, express or implied, is made as to the accuracy or completeness of the presentation and the information contained herein and no reliance should be placed on such information. No responsibility is accepted for any liability for any loss howsoever arising, directly or indirectly, from this presentation or its contents.