

NATURAL LANGUAGE UNDERSTANDING AT SCALE

WITH SPARK NLP

David Talby

Claudiu Branzan

Alex Thomas

Setup your laptop: `github.com/JohnSnowLabs/spark-nlp-workshop`

CONTENTS

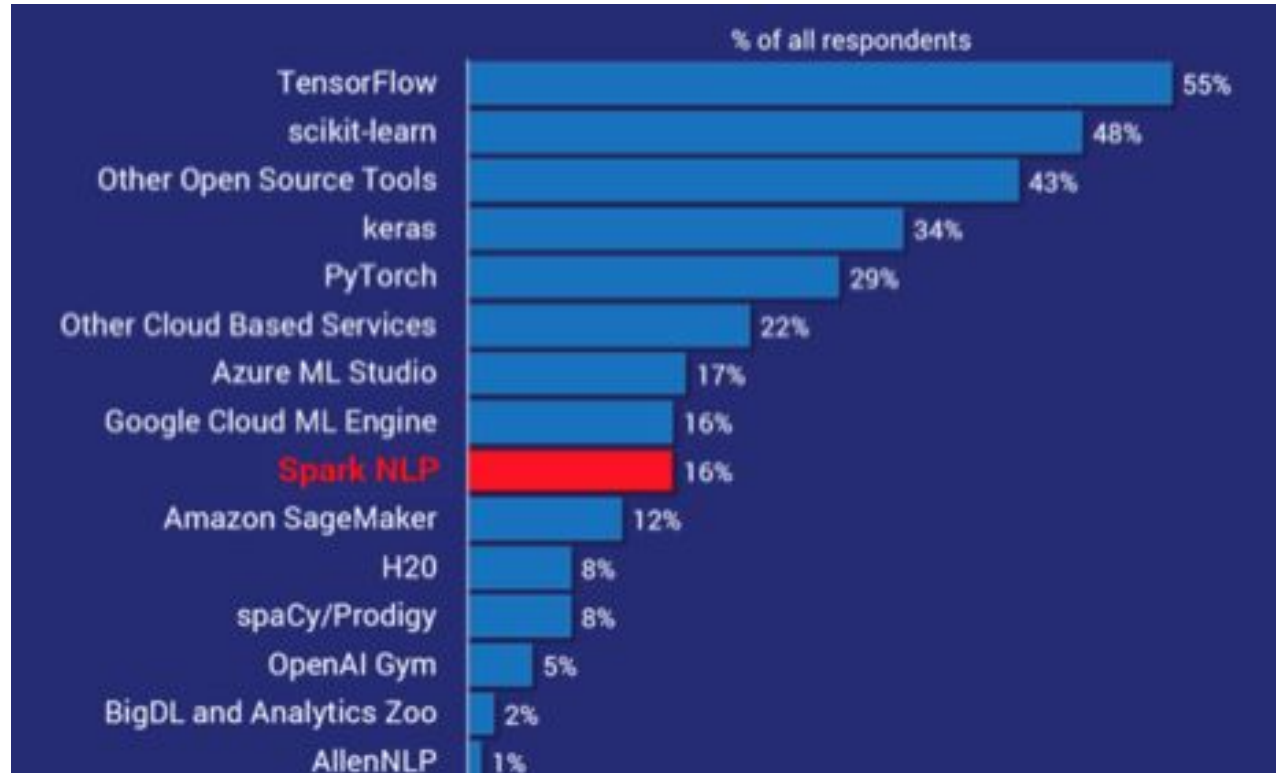
1. **INTRODUCING SPARK NLP**
2. **GETTING THINGS DONE**
3. **THE ANNOTATORS**
4. **TRAINING CUSTOM MODELS**

INTRODUCING SPARK NLP

STATE OF THE ART NLP FOR PYTHON, JAVA & SCALA

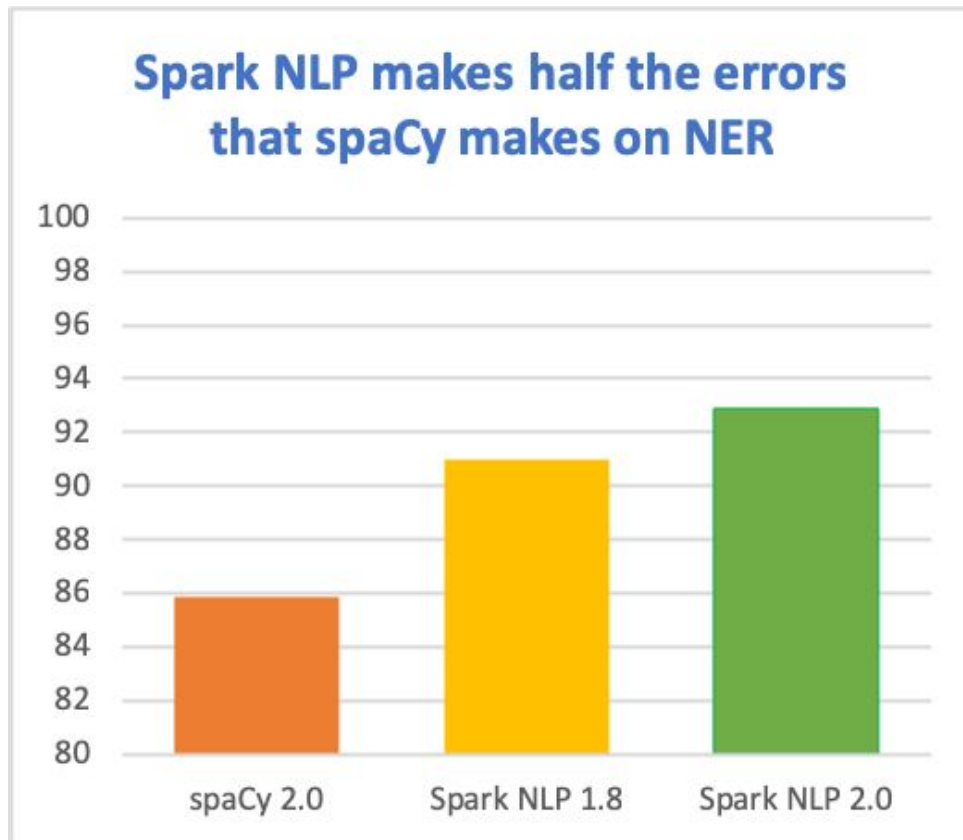
1. ACCURACY
2. SPEED
3. SCALABILITY

SPARK NLP IN THE ENTERPRISE



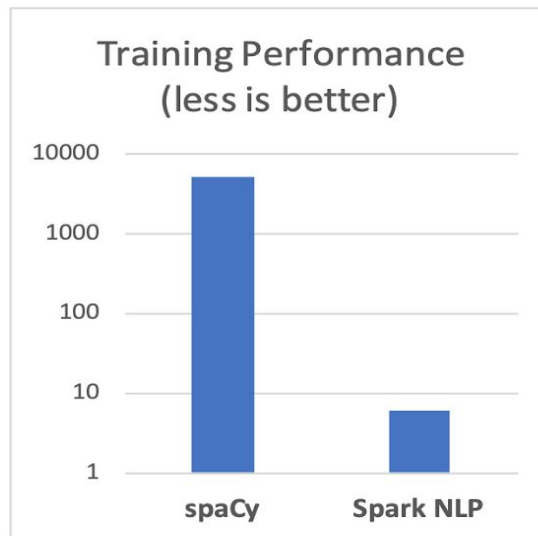
ACCURACY

- "State of the art" means the best performing academic peer-reviewed results
- NER Benchmark on right is on en_core_web_lg dataset, micro-averaged F1 score
- Why is it more accurate?
 - Deep learning models, trainable at scale with GPU's
 - TF graph based on 2017 paper (bi-LSTM+CNN+CRF)
 - BERT embeddings
 - Contrib LSTM cells



SPEED

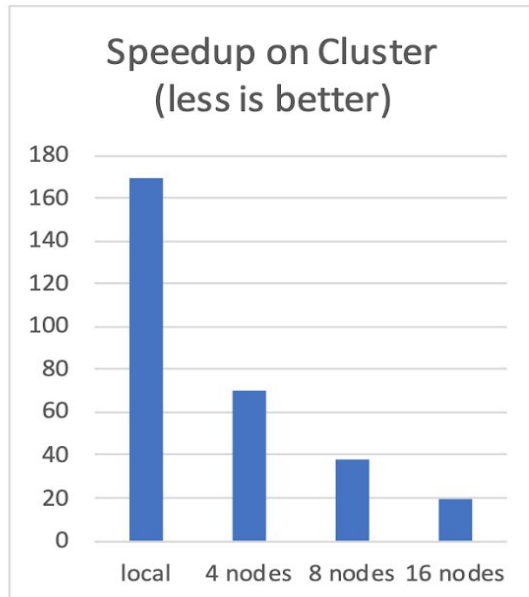
- Benchmark for training a pipeline with sentence bouncer, tokenizer, and POS tagger
- Trained on single Intel i5 machine with 4 cores, 16GB RAM, SSD
- Why is it faster?
 - 2nd gen Tungsten engine: whole stage code generation, vectorized in-memory columnar data
 - No copying of text in memory
 - Extensive profiling, config & code optimization of Spark and TensorFlow
 - Optimized for training and inference



Spark NLP trains 80x faster than spaCy on one machine

SCALE

- Zero code changes to scale a pipeline to any Spark cluster
- Only natively distributed open-source NLP library
- Spark provides execution planning, caching, serialization, shuffling
- Caveats
 - Speedup depends heavily on what you actually do
 - Not all algorithms scale well
 - Spark configuration matters



**Spark NLP natively scales
on any Spark cluster**

SPARK NLP

Permissive Open Source License

Apache 2.0

versus Stanford CoreNLP & spaCy models

SPARK NLP

Production Grade & Actively Supported

In production in [multiple Fortune 500's](#)

25 new releases in 2018

Full-time development team

Active [Slack community](#)

THAT'S NICE, BUT WHAT DOES IT ACTUALLY DO?

(Everything that other NLP
libraries do and more.)

Name	NLTK	spaCy	CoreNLP	Spark NLP
Sentence Detection	Yes	Yes	Yes	Yes
Tokenization	Yes	Yes	Yes	Yes
Stemming	Yes	Yes	Yes	Yes
Lemmatization	Yes	Yes	Yes	Yes
POS Tagger	Yes	Yes	Yes	Yes
NER	Yes	Yes	Yes	Yes
Dependency Parse	Yes	Yes	Yes	Yes
Text Matcher	No	No	Yes	Yes
Date Matcher	No	No	Yes	Yes
Chunking	Yes	Yes	Yes	Yes
Spell Checker	No	No	No	Yes
Sentiment Detector	No	No	Yes	Yes
Pre-trained Models	Yes	Yes	Yes	Yes
Training Models	Yes	Yes	Yes	Yes

WHERE CAN I USE IT?

Features	Spark NLP	NLTK	spaCy	CoreNLP	OpenNLP
Provides full Java API	Yes	No	No	Yes	Yes
Provides full Scala API	Yes	No	No	No	No
Provides full Python API	Yes	Yes	Yes	No	No
Supports training on GPU	Yes	No	Yes	No	No
Supports user-defined deep learning networks	Yes	No	No	No	No
Supports Spark Natively	Yes	No	No	No	No
Supports Hadoop (YARN and HDFS)	Yes	No	No	No	No



BUILT ON THE SHOULDERS OF SPARK ML

- Reusing the Spark ML Pipeline
 - Unified NLP & ML pipelines
 - End-to-end execution planning
 - Serializable
 - Distributable
- Reusing NLP Functionality
 - TF-IDF calculation
 - String distance calculation
 - Stop word removal
 - Topic modeling
 - Distributed ML algorithms

High Performance Natural Language Understanding at Scale



Part of Speech Tagger
Named Entity Recognition
Sentiment Analysis
Spell Checker
Tokenizer
Stemmer
Lemmatizer
Entity Extraction



Topic Modeling
Word2Vec
TF-IDF
String distance calculation
N-grams calculation
Stop word removal
Train/Test & Cross-Validate
Ensembles

Spark ML API (Pipeline, Transformer, Estimator)

Spark SQL API (DataFrame, Catalyst Optimizer)

Spark Core API (RDD's, Project Tungsten)

Data Sources API

CONTENTS

1. INTRODUCING SPARK NLP
2. **GETTING THINGS DONE**
3. THE ANNOTATORS
4. TRAINING CUSTOM MODELS

SENTIMENT ANALYSIS

```
import sparknlp
sparknlp.start()

from sparknlp.pretrained import PretrainedPipeline
pipeline = PretrainedPipeline('analyze_sentiment_ml', 'en')
result = pipeline.annotate('Harry Potter is a great movie')

print(result['sentiment'])    # will print ['positive']
```

NAMED ENTITY RECOGNITION

Chandler PERSON and Monica PERSON met in Central Perk Loc .

```
pipeline = PretrainedPipeline('recognize_entities_bert', 'en')
result = pipeline.annotate('Harry Potter is a great movie')

print(result['ner'])
# prints ['I-PER', 'I-PER', '0', '0', '0', '0']
```

SPELL CHECKING & CORRECTION

Now in Scala:

```
val pipeline = PretrainedPipeline("spell_check_ml", "en")
val result = pipeline.annotate("Harry Potter is a graet movie")

println(result("spell"))
/* will print Seq[String](..., "is", "a", "great", "movie") */
```

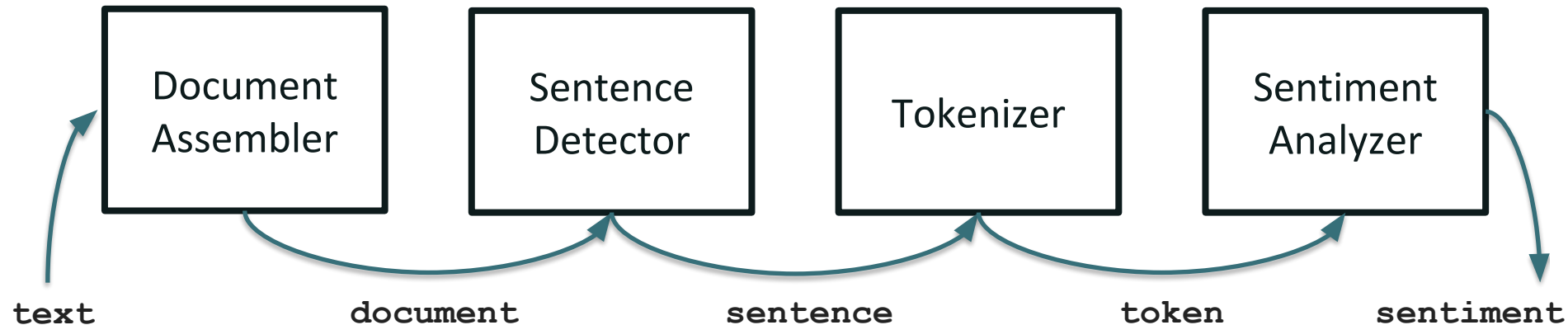

PLAY TIME!

Using pretrained pipelines

Open “How to use Light Pipelines” notebook

KEY CONCEPT #1: PIPELINE

A list of text processing steps.
Each step has input and output columns.



KEY CONCEPT #2: ANNOTATOR

An object encapsulating one text processing step.

```
sentiment_detector = SentimentDetector()  
    .setInputCols(["sentence"])  
    .setOutputCol("sentiment_score")  
    .setDictionary(resource_path+"sent.txt")
```

KEY CONCEPT #3: RESOURCE

An external file that an annotator needs.

- Trained ML models
- Trained DL networks
- Dictionaries
- Embeddings
- Rules
- Pretrained pipelines

Resources can be shared, cached, and locally stored.

KEY CONCEPT #4: PRETRAINED PIPELINE

A pre-built pipeline, with all the annotators and resources it needs.

```
pipeline = PretrainedPipeline('explain_document_d1', 'en')  
result   = pipeline.annotate('a sentence to analyze')  
results  = pipeline.transform(spark_dataframe)
```

UNDER THE HOOD

1. `sparknlp.start()` starts a new Spark session if there isn't one, and returns it.
2. `PretrainedPipeline()` loads the English version of the `explain_document_dl` pipeline, the pre-trained models, and the embeddings it depends on.
3. These are stored and cached locally.
4. TensorFlow is initialized, within the same JVM process that runs Spark.
The pre-trained embeddings and deep-learning models (like NER) are loaded.
Models are automatically distributed and shared if running on a cluster.
5. The `annotate()` call runs an NLP inference pipeline which activates each stage's algorithm (tokenization, POS, etc.).
6. The NER stage is run on TensorFlow – applying a neural network with bi-LSTM layers for tokens and a CNN for characters.
7. Embeddings are used to convert *contextual* tokens into vectors during the NER inference process.
8. The `result` object is a plain old local Python dictionary.

PLAY TIME!

Entity Recognition with Deep Learning

Open “Entity Recognizer DL” notebook

THREE KINDS OF PIPELINES

- **Spark Pipeline**

- Efficiently run on a whole Spark Dataframe
- Distributable on a cluster
- Uses Spark tasks, optimizations & execution planning
- Used by `PretrainedPipeline.transform()`

- **Light Pipeline**

- Efficiently run on a single sentence
- Faster than a Spark pipeline for up to 50,000 local documents
- Easiest way to publish a pipeline as an API
- Used by `PretrainedPipeline.annotate()`

- **Recursive Pipeline**

- Give annotators access to other annotators in the same pipeline
- Required when training your own models

FRICTIONLESS REUSE OF SPARK ML PIPELINES

```
pipeline = pyspark.ml.Pipeline(stages=[
    document_assembler,
    tokenizer,
    stemmer,
    normalizer,
    stopword_remover,
    tf,
    idf,
    lda])
```

```
topic_model = pipeline.fit(df)
```

Spark NLP annotators

Spark ML featurizers

Spark ML LDA implementation

Single execution plan for the given data frame



CONTENTS

1. INTRODUCING SPARK NLP
2. GETTING THINGS DONE
3. **THE ANNOTATORS**
4. TRAINING CUSTOM MODELS

WHICH ANNOTATORS ARE AVAILABLE?

OCR & PDF
Parsing



Named Entity
Recognition

I love Lucy PERSON



State of the Art
Natural Language Processing

Spelling
Correction

Tabel -> Table

Sentiment
Analysis



Core

Tokenizer
Normalizer
Stemmer
Lemmatizer

Document Assembler
Sentence Detector
Part of Speech Tagger
Dependency Parser

Text Matcher
Regex Matcher
Date Matcher
Chunker



Word2Vec
Topic Modeling
TF / IDF
Stop words
n-grams
String distance
ML Pipelines



Prebuilt Graphs
DL Pipelines

Pipelines



Spark



Light



Recursive



Data



Models

SENTENCE BOUNDARY DETECTION

Can't splitting text into sentences be done with regular expressions?

Yes on professionally written text (books, newspapers) but not on social media posts, mobile messaging, and on speech:

no what is that I have not heard of that
no / what is that / I have not heard of that /

SENTENCE BOUNDARY DETECTION

1.2 Confidential Information shall not include any information that:

- (a) was rightfully known to the Recipient Party without restriction before receipt from Pacific AI;
- (b) is rightfully disclosed to the Recipient Party without restriction by a third party;
- (c) is or becomes generally known to the public without violation of this Agreement by the Recipient Party, or
- (d) is independently developed by the Recipient Party or its employees without access to or reliance on such information.

1.3 Pacific AI represents and warrants to the Recipient Party that it is authorized to disclose any and all Confidential Information made available to the Recipient Party under this Agreement.

1

2. Restrictions.

2.1 The Recipient Party agrees:

- (a) to use the Confidential Information only for the purpose set out in this Agreement and for its consideration internally of a business relationship or transaction between the parties, and

- Sentence boundary is also hard for text coming from long documents (DOCX, PDF, OCR)
- Challenging to handle:
 - Page breaks
 - Paragraph breaks
 - Headers & footers
 - Lists
 - Callouts
 - Columns

TOKENIZATION

“identifying the words”

from:

he didn't arrive.

to:

He

did

n't

arrive

.

NORMALIZATION

Remove or replace undesirable characters or regular expressions:

from: `<h1 style="color: #5e9ca0;">Have a great birth day!</h1>`
to: Have a great birth day!

Spark NLP also comes with a Slang normalizer:

Original tweet

@USER, r u cuming 2 MidCorner dis Sunday?

Normalized tweet

@USER, are you coming to MidCorner this Sunday?



STEMMING

Find the **stem** of each word.

Uses rules: For example, remove common suffixes.

Form	Suffix	Stem
studie s	-es	studi
study ing	-ing	study
niñ as	-as	niñ
niñ ez	-ez	niñ

LEMMATIZATION

Find the **lemma** of each word: How does it show in the dictionary?

Uses a lookup from a full dictionary.

am, are, is → be

liver → liver

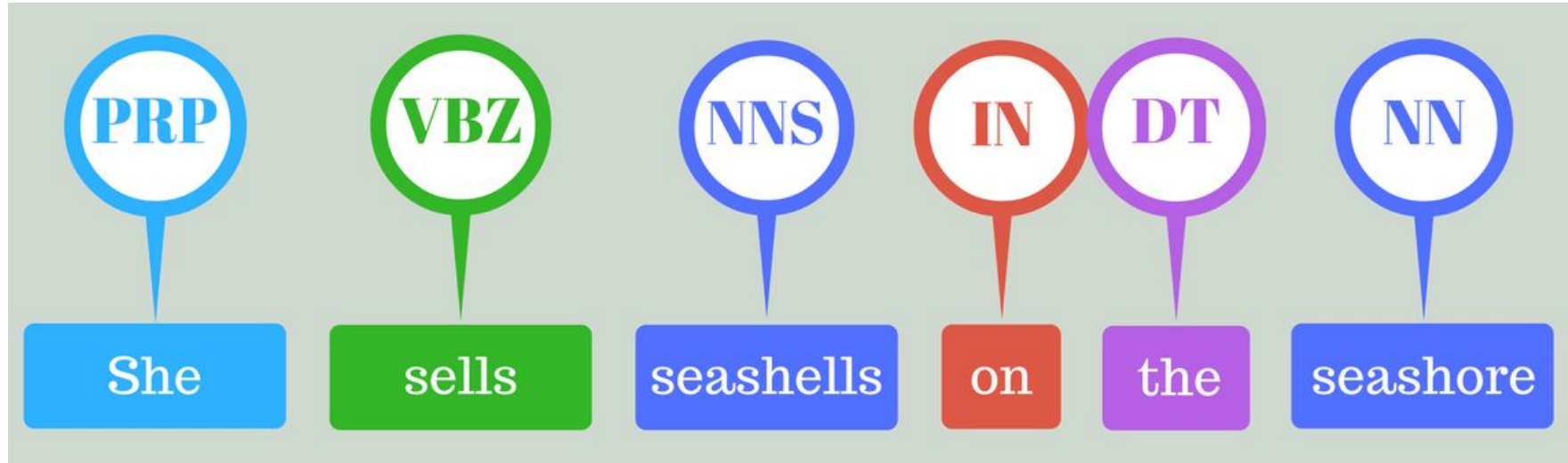
lives → live

STEMMING VS. LEMMATIZATION

- Using one of them is usually required in pipelines that perform topic modeling, text indexing, query expansion, or classification
- Choose based on your use case:
 - Lemmatization always returns real words, stemming doesn't
 - Lemmatization misses words that aren't in the dictionary
 - Lemmatization is faster (single lookup), at least in English
 - Lemmatization requires more memory (the dictionary)

PART OF SPEECH TAGGING

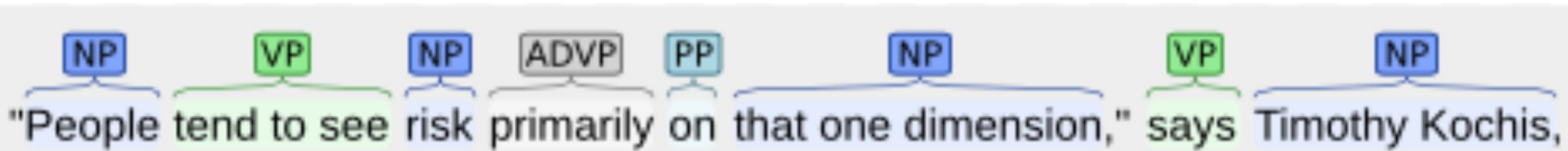
Often useful for recognizing named entities or word relationships.



CHUNKING

Combining words that form a single part of speech.

Required as part of an entity recognition or fact extraction pipeline.



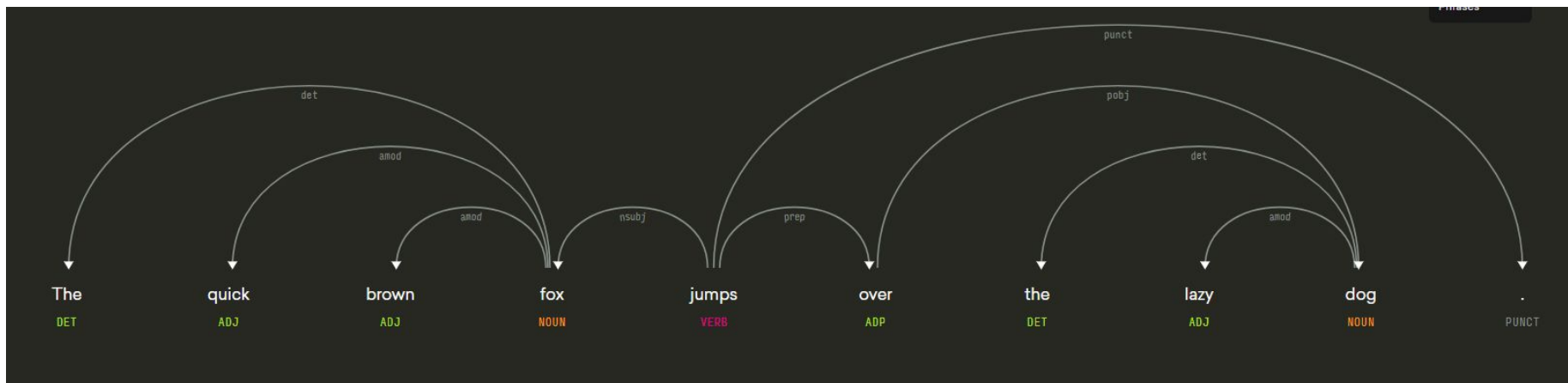
PLAY TIME!

Topic Modeling

Open “How to use Spark NLP and Spark ML Pipelines” notebook

DEPENDENCY PARSING

Useful for extracting relationships (i.e. building knowledge graphs):



NAMED ENTITY RECOGNITION

Chandler PERSON and Monica PERSON met in Central Perk LOC .

Spark NLP comes with two NER annotators:

- `NerCrfApproach` – machine learning based NER
- `NerDLApproach` – deep learning based NER
- Deep learning NER is more accurate *if given more training data*

BERT vs GloVe

Bidirectional Encoder Representations from Transformers is the first *deeply bidirectional, unsupervised* language representation, pre-trained using only a plain text corpus (in this case, Wikipedia).

Pre-trained representations can either be *context-free* or *contextual*, and *contextual* representations can further be *unidirectional* or *bidirectional*. Context-free models such as **word2vec** or **GloVe** generate **a single word embedding representation for each word** in the vocabulary. For example, the word “*bank*” would have the same context-free representation in “*bank account*” and “*bank of the river*.” Contextual models instead generate a representation of each word that is based on the other words in the sentence. For example, in the sentence “*I accessed the bank account*,” a unidirectional contextual model would represent “*bank*” based on “*I accessed the*” but not “*account*.” However, **BERT represents “bank” using both its previous and next context** — “*I accessed the ... account*” — starting from the very bottom of a deep neural network, making it **deeply bidirectional**

http://www.davidsbatista.net/blog/2018/12/06/Word_Embeddings/

NAMED ENTITY RECOGNITION

- Pretrained models find **person**, **location**, or **organization** entities
- It's more important that you can train your own:

around the left eye . <test>CT of the brain</test> showed no <problem>acute changes </problem> , <problem>left periorbital soft tissue swelling </problem> . <test> CT of the maxillofacial area</test> showed no <problem>facial bone fracture </problem> . <test> Echocardiogram </test> showed normal left ventricular function , <test>ejection fraction</test> estimated greater than 65% . She was set up with a skilled nursing facility , which took several days to arrange , where she was to be given <treatment>daily physical therapy</treatment> and <treatment> rehabilitation </treatment> until appropriate .

SENTIMENT ANALYSIS

Spark NLP comes with two:

- `SentimentDetector`
Define a set of tokens for positive, negative, increment, decrement & reverse multipliers
- `ViveknSentimentApproach`
Train an ML model on a set of positive and negative examples
- Trainable for different emotions



SPELL CHECKING & CORRECTION



- 3 trainable approaches
- **Norvig Approach:**
 - Retrieves tokens and auto-corrects based on a given dictionary
- **Symmetric Delete:**
 - Uses distance metrics to find possible words
- **Context Aware:**
 - Most accurate: Judges words in context
 - Deep learning based

WHAT ELSE IS AVAILABLE?

- Some models are available in *Italian* and *French* (more coming)
- *GloVe* and *BERT* embeddings are pretrained
- There are *Enterprise* and *Healthcare* commercial editions

Community Edition

Open Source: Apache 2.0 License

Highly active: 25 released in 2018

Support at spark-nlp.slack.com

Full Python, Java and Scala API's

Enterprise Edition

Same-day email & phone support

Personalized onboarding help

Patent-pending entity resolution

Personal data de-identification

Healthcare Edition

Biomedical deep learning NER

Map to medical terminologies

Negation detection

Clinical spell check, sentiment,
OCR, POS & de-identify models

PLAY TIME!

Notebook #4: Document Classification

Open “Sarcasm Classifier” notebooks

ONE MORE THING: OCR

- Object Character Recognition is included with Spark NLP
- It can automatically read and extract text from PDF files, reading either digital text or scanned images
- Requires installing an additional JAR file
- Based on a custom build of Tesseract
- Replaces `DocumentAssembler` as a source of text of pipelines

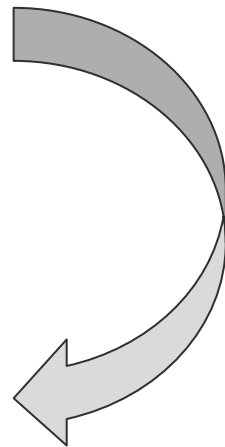
OCR IN SPARK NLP: BENEFITS

1. Image pre-processing
2. Layout detection
3. Distributed OCR
4. Sentence detection
5. OCR spell correction

OCR: IMAGE PRE-PROCESSING

Patient is an 84-year-old male with a past medical history of hypertension, HFpEF last known EF 55%, mild to moderate TR, pulmonary hypertension, permanent atrial fibrillation on Eliquis, history of GI bleed, CK-MB, and anemia who presents with full weeks of generalized fatigue and feeling unwell. He also notes some shortness of breath and worsening dyspnea with minimal exertion. His major complaints are shoulder and joint pains, diffusely. He also complains of "bone pain". He denies having any fevers or chills. He denies having any chest pain, palpitations. He denies any worse extremity swelling than his baseline. He states he's been compliant with his medications. Although he states he ran out of his Eliquis a few weeks ago. He denies having any blood in his stools or melena, although he does take iron pills and states his stools are frequently black. His hemoglobin is at baseline.

Patient is an 84-year-old male with a past medical history of hypertension, HFpEF last known EF 55%, mild to moderate TR, pulmonary hypertension, permanent atrial fibrillation on Eliquis, history of GI bleed, CK-MB, and anemia who presents with full weeks of generalized fatigue and feeling unwell. He also notes some shortness of breath and worsening dyspnea with minimal exertion. His major complaints are shoulder and joint pains, diffusely. He also complains of "bone pain". He denies having any fevers or chills. He denies having any chest pain, palpitations. He denies any worse extremity swelling than his baseline. He states he's been compliant with his medications. Although he states he ran out of his Eliquis a few weeks ago. He denies having any blood in his stools or melena, although he does take iron pills and states his stools are frequently black. His hemoglobin is at baseline.



- Rotation
- Scaling
- Erosion

OCR: LAYOUT DETECTION

Review of Systems

A 10 system review of systems was completed and negative except as documented in HPI.

Physical Exam

Vitals & Measurements

T: 36.8 °C (Oral) TMIN: 36.8 °C (Oral) TMAX: 37.0 °C (Oral) HR: 54 RR: 17

BP: 140/63 WT: 100.3 KG

Pulse Ox: 100 % Oxygen: 2 L/min via Nasal Cannula

GENERAL: no acute distress

HEAD: normocephalic

EYES/EARS/NOSE/THROAT: pupils are equal, normal oropharynx

NECK: normal inspection

RESPIRATORY: no respiratory distress, no rales on my exam

CARDIOVASCULAR: irregular, brady, no murmurs, rubs or gallops

ABDOMEN: soft, non-tender

EXTREMITIES: Bilateral chronic venous stasis changes

NEUROLOGIC: alert and oriented x 3, no gross motor or sensory deficits

Assessment/Plan

Acute on chronic diastolic CHF (congestive heart failure)

Acute on chronic diastolic heart failure exacerbation. Small pleural effusions bilaterally with mild pulmonary vascular congestion on chest x-ray, slight elevation in BNP. We'll continue 1 more day of IV diuresis with 80 mg IV Lasix. He may have had a viral infection which precipitated this. We'll add Tylenol for his joint pains. Continue atenolol and chlorthalidone.

AF - Atrial fibrillation

Permanent atrial fibrillation. Rates bradycardic in the 50s. Continue atenolol with hold parameters. Continue Eliquis for stroke prevention. No evidence of bleeding, hemoglobin at baseline.

Home Medications

Home

allopurinol 300 mg oral tablet, 300 MG= 1

TAB, PO, Daily

atenolol 25 mg oral tablet, 25 MG= 1 TAB,

PO, Daily

chlorthalidone 25 mg oral tablet, 25 MG=

1 TAB, PO, M/W/F

Combigan 0.2%-0.5% ophthalmic

solution, 1 DROP, Both Eyes, Q12H

Eliquis 5 mg oral tablet, 5 MG= 1 TAB,

PO, BID

ferrous sulfate 325 mg (65 mg elemental

iron) oral tablet, 325 MG= 1 TAB, PO,

Daily

Lasix 80 mg oral tablet, 80 MG= 1 TAB,

PO, BID

omeprazole 20 mg oral delayed release

capsule, 20 MG= 1 CAP, PO, BID

Percocet 5/325 oral tablet, 1 TAB, PO,

QAM

potassium chloride 20 mEq oral tablet,

extended release, 20 MEQ= 1 TAB, PO,

Daily

sertraline 50 mg oral tablet, 75 MG= 1.5

TAB, PO, Daily

triamcinolone 0.1% topical cream, 1 APP,

Topical, Daily

triamcinolone 0.1% topical ointment, 1

APP, Topical, Daily

OCR-SPECIFIC SENTENCE DETECTION & SPELL CHECK

1.2 Confidential Information shall not include any information that:

- (a) was rightfully known to the Recipient Party without restriction before receipt from Pacific AI;
- (b) is rightfully disclosed to the Recipient Party without restriction by a third party;
- (c) is or becomes generally known to the public without violation of this Agreement by the Recipient Party, or
- (d) is independently developed by the Recipient Party or its employees without access to or reliance on such information.

1.3 Pacific AI represents and warrants to the Recipient Party that it is authorized to disclose any and all Confidential Information made available to the Recipient Party under this Agreement.

1

2. Restrictions.

2.1 The Recipient Party agrees:

- (a) to use the Confidential Information only for the purpose set out in this Agreement and for its consideration internally of a business relationship or transaction between the parties, and

- Sentence detection handles:

- Page breaks
- Paragraph breaks
- Headers & footers
- Lists

- Spell correction handles:

- e <-> c
- n <-> m
- l <> 1
- O <> 0



PLAY TIME!

Notebook #5: OCR in Spark NLP

Open “Hardcore DL” notebook

CONTENTS

1. INTRODUCING SPARK NLP
2. GETTING THINGS DONE
3. THE ANNOTATORS
- 4. TRAINING CUSTOM MODELS**

WHY CAN'T I REUSE AN OFF-THE-SHELF NLP MODEL?

Because it won't work for you.

Try it!

[Google Cloud Natural Language](#)

[IBM Watson NLU](#)


[Azure Text Analytics](#)

[spaCy Named Entity Visualizer](#)

[Amazon Comprehend](#) (offline)

[Stanford Core NLP](#)

EXAMPLE 1: EMERGENCY ROOM LANGUAGE

	Triage Notes
states started last night, upper abd, took alka seltzer approx 0500, no relief. nausea no vomiting	
Since yeatreday 10/10 "constant Tylenol 1 hr ago. +nausea. diaphoretic. Mid abd radiates to back	
Generalized abd radiating to lower x 3 days accompanied by dark stools. Now with bloody stool this am. Denies dizzy, sob, fatigue.	



Features	
Type of Pain	Symptoms
Intensity of Pain	Onset of symptoms
Body part of region	Attempted home remedy

Of the 6 engines from the previous slide, the only medical term that only 2 of them recognized was Tylenol as a product.

EXAMPLE 2: SENTIMENT ANALYSIS



Square stock hits record as Cash app reportedly passes Venmo in downloads

Published: Aug 14, 2018 2:53 p.m. ET




A

Analyst tracking downloads of the two payment apps says Cash surpassed Venmo in July



Citron Research
@CitronResearch

Follow



Citron short [\\$TWTR](#). Near-Term target \$25
Of all social media, they are most vulnerable to privacy regulation Wait until Senate finds out what Citron has published.

- News surveillance for traders
- Can you trade quickly based on sentiment of breaking news?
- Started with Watson Sentiment Analyzer & got zero lift
- ... because sentiment analysis models are based on emotions
- Instead of reputation & change

SO, TRAIN YOUR OWN DOMAIN-SPECIFIC MODELS

Next, pair your NLP data scientist with a domain expert who is a master of the language the model will understand.

How good is this review, from 1 to 5?

If you come for the ambiance, you'll be disappointed. But if you go for good, inexpensive and authentic Mexican food, then you're in the right place.

EXAMPLE 3: E-DISCOVERY

Da Silva Moore v. Publicis Groupe, 287 F.R.D. 182, 183-84 (S.D.N.Y. 2012)

By computer-assisted coding, I mean tools... that use sophisticated algorithms to enable the computer to determine relevance, based on interaction with (*i.e.*, training by) a human reviewer.

Rio Tinto PLC v. Vale S.A., 306 F.R.D. 125, 127 (S.D.N.Y. 2015)

In the three years since *Da Silva Moore*, the case law has developed to the point that it is now black letter law that where the producing party wants to utilize TAR for document review, courts will permit it.



SPARK NLP FOR HEALTHCARE

- **Annotators:** entity resolution, negation detection, de-identification
- **Models:** biomedical NER, POS tagger, spell checker, sentiment analysis
- **Resources:** clinical terminologies

HOW MUCH LABELED DATA DO I NEED?

1. Get representative data

“Inception v3 was trained on 1.28 million images”

“used over 120,000 retinal images to train a neural network to detect diabetic retinopathy”

2. Get consistent labels

“In the study, the algorithm went head-to-head against 21 board-certified dermatologists”

“All images were graded by 3 to 7 different ophthalmologists, from a panel of 54 US-licensed senior residents & ophthalmologists”

3. Get pretrained datasets & embeddings

Facebook open sourced pre-trained word vectors for 294 languages, trained on Wikipedia using fastText

UMLS has over 1 million biomedical concepts and 5 million concept names, from over 100 controlled vocabularies

PLAY TIME!

Training NLP Models

Open “Build your own French POS Tagger” notebook

NEXT STEPS

1. READ THE DOCS & JOIN SLACK

[HTTPS://NLP.JOHNSNOWLABS.COM](https://nlp.johnsnowlabs.com)

2. STAR & FORK THE REPO

[GITHUB.COM/JOHNSNOWLABS/SPARK-NLP](https://github.com/johnsnowlabs/spark-nlp)

3. QUESTIONS? GET IT TOUCH



THANK YOU!

Alex Thomas

Claudiu Branzan

David Talby