

Spark NLP for Data Scientists

Training & Certification

Day 1: October 9, 2023 – 9:00 to 13:00 PST

Day 2: October 10, 2023 – 9:00 to 13:00 PST



Healthcare NLP for Data Scientists

Training & Certification

Day 1: October 11, 2023 – 9:00 to 13:00 PST

Day 2: October 12, 2023 – 9:00 to 13:00 PST



**4 days, 16 hrs live training
State-Of-The-Art NLP**

Certification exams in 2 weeks



Spark NLP for Data Scientists

October 9 & 10, 2023

David Amore Cecchini
cecchini@johnsnowlabs.com

Halil Saglamlar
halil@johnsnowlabs.com

Merve Ertas
merve@johnsnowlabs.com

Welcome - We have a lot of things ahead of us

Day-1	50 min	<ul style="list-style-type: none">- Intro to John Snow Labs, Spark NLP and NLP Theory- Spark NLP Pretrained Pipelines basics and JVM/Spark concepts
	10 min	Break
	50 min	<ul style="list-style-type: none">- Text Preprocessing and composing custom pipeline in Spark NLP
	10 min	Break
	50 min	<ul style="list-style-type: none">- Showcase of the 10000+ pretrained Models for 300+ languages in Spark NLP
		Break
	50 min	<ul style="list-style-type: none">- Showcase of the 10000+ pretrained Models for 300+ languages in Spark NLP- Train Named Entity Recognizers (NER)

Welcome - We have a lot of things ahead of us

Day-2	50 min	<ul style="list-style-type: none">- Train Text Classifiers- Upload models to the Models Hub
	10 min	Break
	50 min	<ul style="list-style-type: none">- Spell Checking- Keyword Extraction with YAKE- Rule-based Entity Recognition with EntityRuler- Graph triplet extraction
	10 min	Break
	50 min	<ul style="list-style-type: none">- Token Classification with Transformers- Sequence Classification with Transformers- Image Classification with ViT- Speech to Text with Wav2Vec2
	10 min	Break
	50 min	<ul style="list-style-type: none">- Table Question Answering with TAPAS- Question Answering, Summarization and other T5 applications- Multilingual NLP - Train only on English data and predict for 100+ languages

Session 1 (Day 1)

- ❖ NLP Theory and Introduction to John Snow Labs
- ❖ Pretrained Pipelines
- ❖ Spark, JVM and Spark NLP basic concepts

**Spark NLP
for Data Scientists**

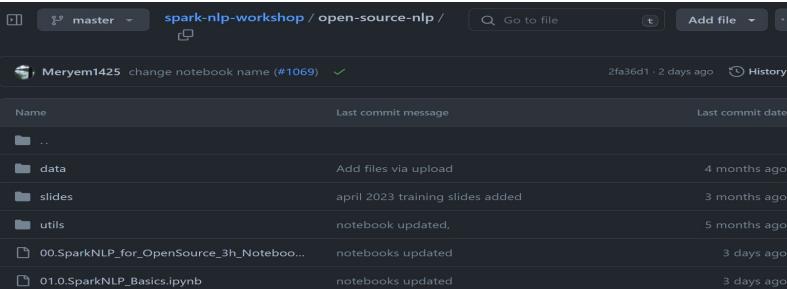


Resources

CODE:

- <https://github.com/JohnSnowLabs/spark-nlp-works/tree/master/open-source-nlp>

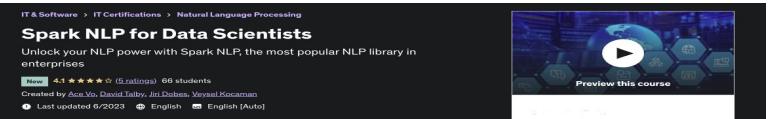
 Open in Colab



A screenshot of a GitHub repository page for 'spark-nlp-workshop / open-source-nlp'. The repository has 1 star and 66 students. It was created by Alaa Yoo, David Talby, Irfan Dabas, Vayasel Kecman and last updated on 6/2/2023. The codebase includes 'data', 'slides', and 'utils' directories, along with '00.SparkNLP_for_OpenSource_3h_Notebook.ipynb' and '01.0.SparkNLP_Basics.ipynb' notebooks. The commit history shows activity from April 2023 to May 2023.

BOOKMARK:

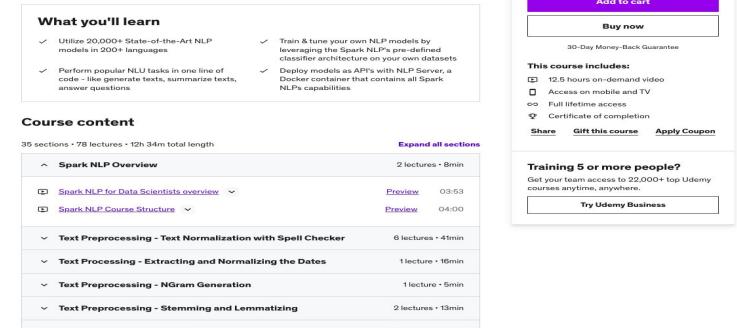
- <https://sparknlp.org/>
- <https://sparknlp.org/docs/en/quickstart>
- <https://sparknlp.org/api/python/reference/index.html>
- Spark-nlp.slack.com



A screenshot of the Udemy course page for 'Spark NLP for Data Scientists'. The course has a 4.5-star rating from 5 ratings, 66 students, and was created by Alaa Yoo, David Talby, Irfan Dabas, Vayasel Kecman. It was last updated on 6/2/2023. The course includes English subtitles and is in English. A 'Preview this course' button is visible.

MOOC:

- <https://www.udemy.com/course/spark-nlp-for-data-scientists/>
- https://github.com/JohnSnowLabs/spark-nlp-workshop/tree/master/Spark_NLP_Udemy_MOOC/Open_Source



A detailed screenshot of the 'Spark NLP for Data Scientists' course page on Udemy. The course has 36 sections and 75 lectures totaling 12h 34m. Key sections include 'Spark NLP Overview', 'Text Preprocessing - Text Normalization with Spell Checker', 'Text Processing - Extracting and Normalizing the Dates', 'Text Preprocessing - NGram Generation', 'Text Preprocessing - Stemming and Lemmatizing', and 'Text Preprocessing Models'. The course includes a 30-day money-back guarantee, access on mobile and TV, full lifetime access, and a certificate of completion. It also offers a 'Buy now' button and a 'Try Udemy Business' button.

Introducing Spark NLP

Introducing Spark NLP



Total downloads

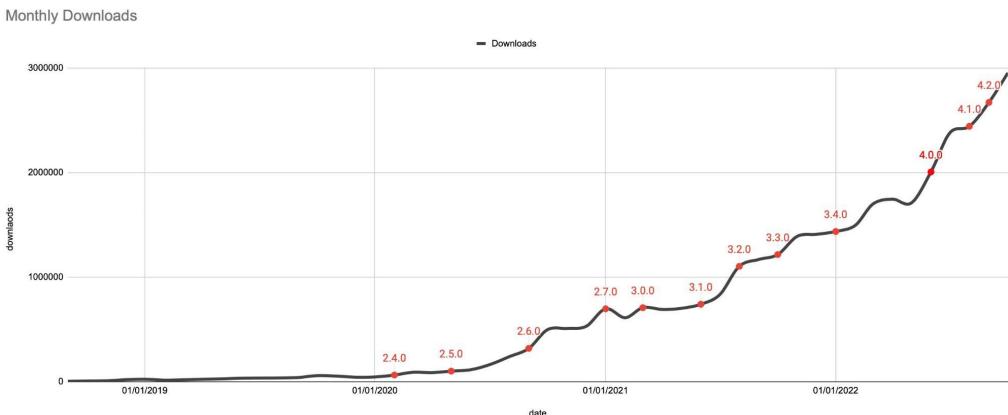
54,315,590

Total downloads - 30 days

2,848,599

Total downloads - 7 days

653,272



Spark NLP is an open-source natural language processing library, built on top of **Apache Spark** and **Spark ML**. (first release: July 2017)

- A single unified solution for all your NLP needs
- Take advantage of transfer learning and implementing the [latest and greatest SOTA algorithms and models](#) in NLP research
- The most widely used NLP library in industry (5 years in a row)
- The most scalable, accurate and fastest library in NLP history
- 121 total releases, every two weeks for the past years

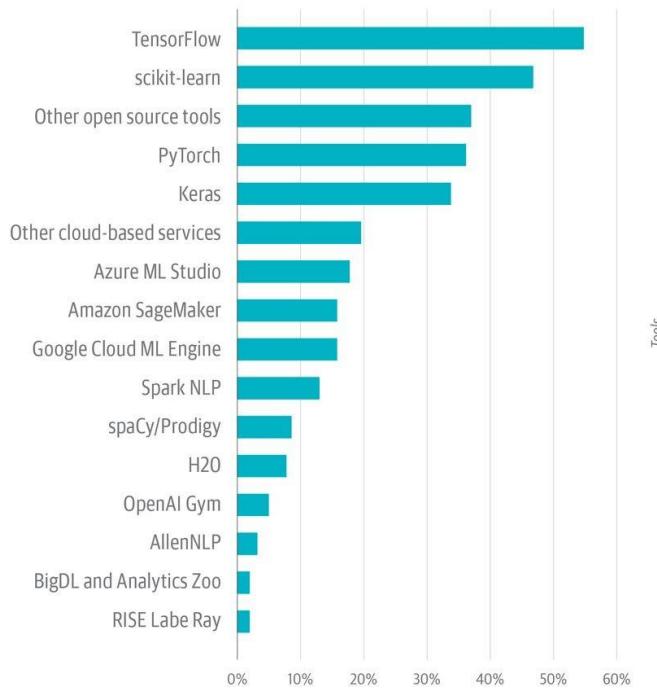
Spark NLP Modules

Entity Recognition  I love Lucy PERSON	Text Classification  She become the first... She became the first	Spelling & Grammar 	Information Extraction They met Last week DATE -> 29-04-2020
Question Answering  ?	Speech to Text 	Image Classification 	Reading Comprehension 
Translation  [[je t'aime -> i love you]]	Summarization 	Paraphrasing You bet! > For sure.	Emotion Detection 

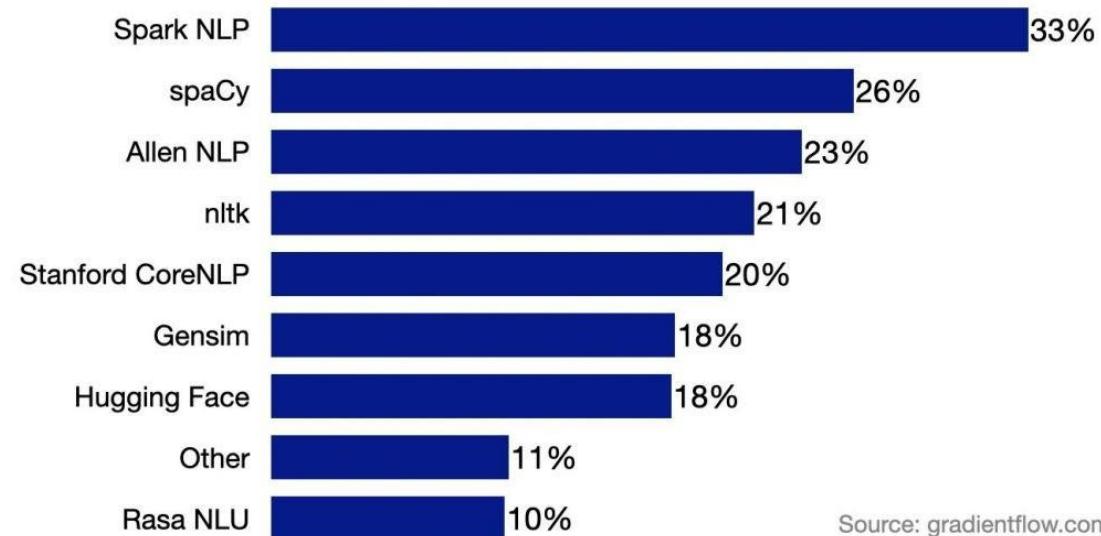
Split Text	Clean Text	20,000+ Pre-trained Pipelines, Models & Transformers	250+ Languages
<ul style="list-style-type: none"> Sentence Detector Tokenizer Normalizer nGram Generator Word Segmentation 	<ul style="list-style-type: none"> Spell Checker Grammar Checker Writing Style Checker Stopword Cleaner Summarization 	<div style="display: flex; justify-content: space-around;"> <div>BERT</div> <div>ELMO</div> <div>TAPAS</div> </div> <div style="display: flex; justify-content: space-around;"> <div>ALBERT</div> <div>DeBERTa</div> <div>USE</div> </div> <div style="display: flex; justify-content: space-around;"> <div>Longformer</div> <div>ELECTRA</div> </div> <div style="display: flex; justify-content: space-around;"> <div>T5</div> <div>NMT</div> <div>VIT</div> </div> <div style="display: flex; justify-content: space-around;"> <div>DistilBERT</div> <div>RoBERTa</div> </div> <div style="text-align: center;">XLM-RoBERTa</div> <div style="display: flex; justify-content: space-around;"> <div>Wav2Vec2</div> <div>XLNet</div> </div>	
Understand Grammar	Find in Text		
<ul style="list-style-type: none"> Stemmer Lemmatizer Part of Speech Tagger Dependency Parser Translation 	<ul style="list-style-type: none"> Text Matcher Regex Matcher Date Matcher Chunker Question Answering 		
Trainable & Tunable	Scalable	Fast Inference	Hardware Optimized

Spark NLP in Industry

Which of the following AI tools do you use?



Which NLP libraries does your organization use?



Source: gradientflow.com

NLP Industry Survey by Gradient Flow,
an independent data science research & insights company, September 2021

TRUSTED BY



Imperial College
London



STANFORD
UNIVERSITY

Biomedical Named Entity Recognition at Scale

Veysel Kocaman
John Snow Labs Inc.
16192 Coastal Highway
Lewes, DE , USA 19958
veysel@johnsnowlabs.com

Abstract—Named entity recognition (NER) is a widely applicable natural language processing task and building block of question answering, topic modeling, information retrieval, etc. In the medical domain, NER plays a crucial role by extracting meaningful chunks from clinical notes and reports, which are then fed to downstream tasks like assertion status detection, entity resolution, relation extraction, and de-identification. Reimplementing a Bi-LSTM-CNN-Char deep learning architecture on top of Apache Spark, we present a single trainable NER model that obtains new state-of-the-art results on seven public biomedical benchmarks without using heavy contextual embeddings like BERT. This includes improving BC4CHEMD to 93.72% (4.1% gain), Species800 to 80.91% (4.6% gain), and JNLPBA to 81.29% (5.2% gain). In addition, this model is freely available within a production-grade code base as part of the open-source Spark NLP library; can scale up for training and inference in any Spark cluster; has GPU support and libraries for popular programming languages such as Python, R, Scala and Java; and can be extended to support other human languages with no code changes.

I. INTRODUCTION

Electronic health records (EHRs) are the primary source of information for clinicians tracking the care of their patients. Information fed into these systems may be found in structured fields for which values are inputted electronically (e.g. laboratory test orders or results) [1] but most of the time information in these records is unstructured making it largely inaccessible

Accurate Clinical and Biomedical Named Entity Recognition at Scale

Anonymous NAACL-HLT 2021 submission

Abstract

Named entity recognition (NER) is one of the most important building blocks of NLP tasks in the medical domain by extracting meaningful chunks from clinical notes and reports, which are then fed to downstream tasks like assertion status detection, entity resolution, relation extraction, and de-identification. Due to the growing volume of healthcare data in unstructured format, an increasingly important challenge is providing high accuracy implementations of state-of-the-art deep learning (DL) algorithms at scale. In this study, we introduce a production-grade clinical and biomedical NER algorithm based on a modified BiLSTM-CNN-Char DL architecture built on top of Apache Spark. This algorithm establishes new state-of-the-art accuracy on 7 of 8 well-known biomedical NER benchmarks and 3 clinical concept extraction challenges: 2010 i2b2/VA clinical concept extraction, 2014 n2c2 de-identification, and 2018 n2c2 medication extraction. Moreover, clinical NER models trained using this implemen-

Spark NLP: Natural Language Understanding at Scale

Veysel Kocaman, David Talby

John Snow Labs Inc.
16192 Coastal Highway
Lewes, DE , USA 19958
eysel, david}@johnsnowlabs.com

Improving Clinical Document Understanding on COVID-19 Research with Spark NLP

Veysel Kocaman, David Talby

John Snow Labs Inc.
16192 Coastal Highway
Lewes, DE , USA 19958
{eysel, david}@johnsnowlabs.com

Abstract

Following the global COVID-19 pandemic, the number of scientific papers studying the virus has grown massively, leading to increased interest in automated literature review. We present a clinical text mining system that improves on previous efforts in three ways. First, it can recognize over 100 different entity types including social determinants of health, anatomy, risk factors, and adverse events in addition to other commonly used clinical and biomedical entities. Second, the text processing pipeline includes assertion status detection, to distinguish between clinical facts that are present, absent, conditional, or about someone other than the patient. Third, the deep learning models used are more accurate than previously available, leveraging an integrated pipeline of state-of-the-art pre-trained named entity recognition models, and improving on the previous best performing benchmarks for assertion status detection. We illustrate extracting trends and insights - e.g. most frequent disorders and symptoms, and most common vital signs and EKG findings – from the COVID-19 Open Research Dataset (CORD-19). The system is built using the Spark NLP library which natively supports scaling to use distributed clusters, leveraging GPU's, configurable and reusable NLP pipelines, healthcare-specific embeddings, and the ability to train models to support new entity types or human languages with no code changes.

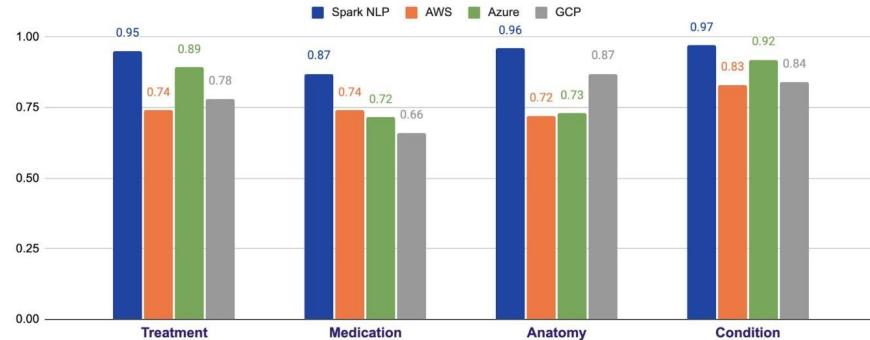
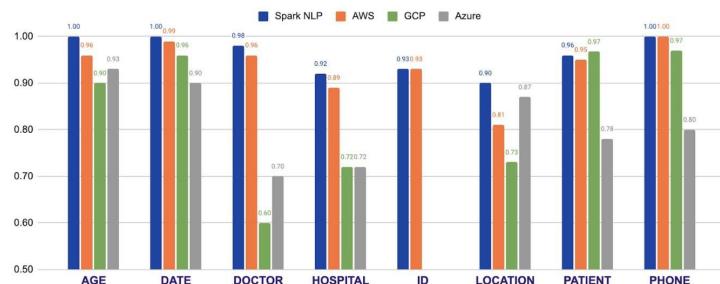
be found in structured fields for which values are inputted electronically (e.g. laboratory test orders or results) (Liede et al. 2015) but most of the time information in these records is unstructured making it largely inaccessible for statistical analysis (Murdoch and Detsky 2013). These records include information such as the reason for administering drugs, previous disorders of the patient or the outcome of past treatments, and they are the largest source of empirical data in biomedical research, allowing for major scientific findings in highly relevant disorders such as cancer and Alzheimer's disease (Perera et al. 2014).

A primary building block in such text mining systems is named entity recognition (NER) - which is regarded as a critical precursor for question answering, topic modelling, information retrieval, etc (Yadav and Bethard 2019). In the medical domain, NER recognizes the first meaningful chunks out of a clinical note, which are then fed down the processing pipeline as an input to subsequent downstream tasks such as clinical assertion status detection (Uzuner et al. 2011), clinical entity resolution (Tzitzivacos 2007) and de-identification of sensitive data (Uzuner, Luo, and Szolovits 2007) (see Figure 1). However, segmentation of clinical and drug entities is considered to be a difficult task in biomedical NER systems because of complex orthographic structures of named entities

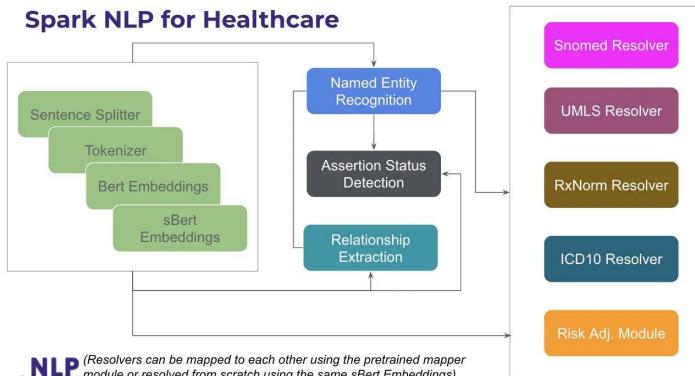
Peer-reviewed papers on
Spark NLP NER

Most Accurate in the Industry

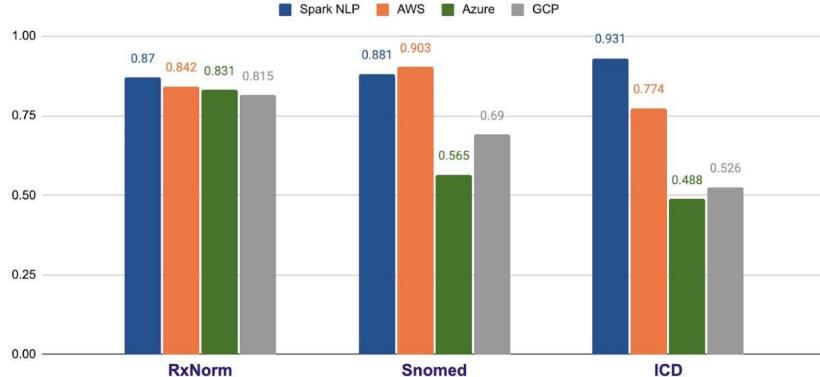
De-Identification Benchmarks (en)



Spark NLP for Healthcare



Top - 5 Results



Light Pipelines for Fast Inference



Faster inference

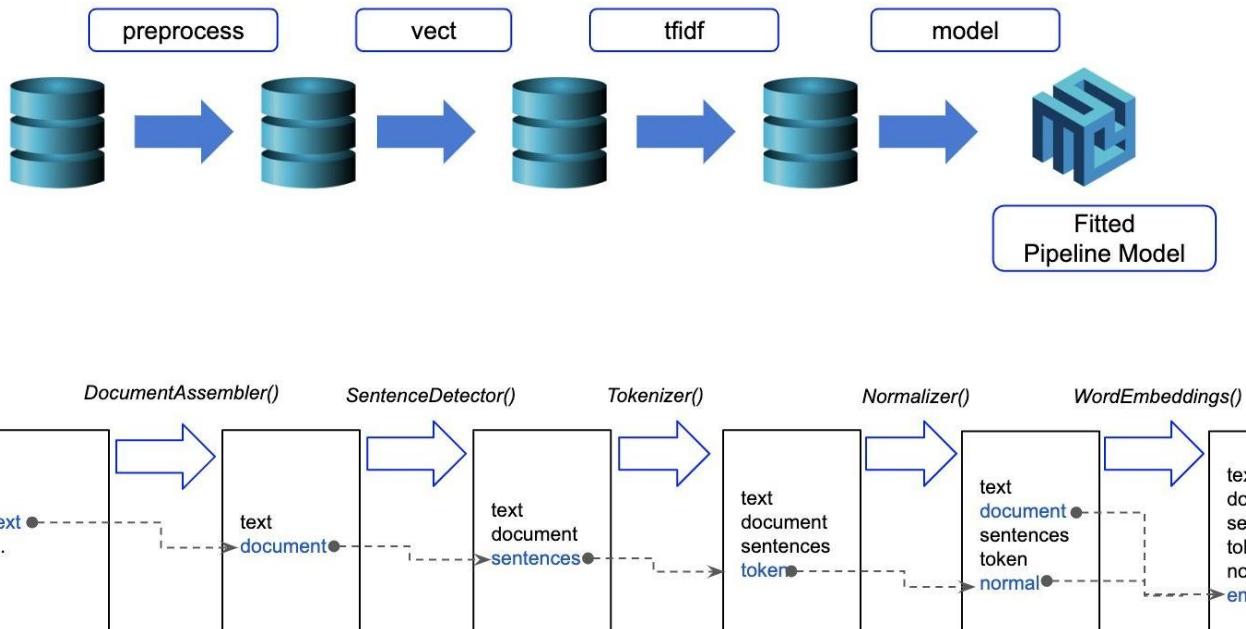
```
from sparknlp.base import LightPipeline
LightPipeline(someTrainedPipeline).annotate(someStringOrArray)
```

Spark is like a **locomotive racing a bicycle**. The **bike** will win if the load is light, it is quicker to accelerate and more agile, but with a heavy load the **locomotive** might take a while to get up to speed, but **it's going to be faster in the end**.

LightPipelines are Spark ML pipelines converted into a single machine but multithreaded task, becoming more than 10x times faster for smaller amounts of data (small is relative, but 50k sentences is roughly a good maximum).

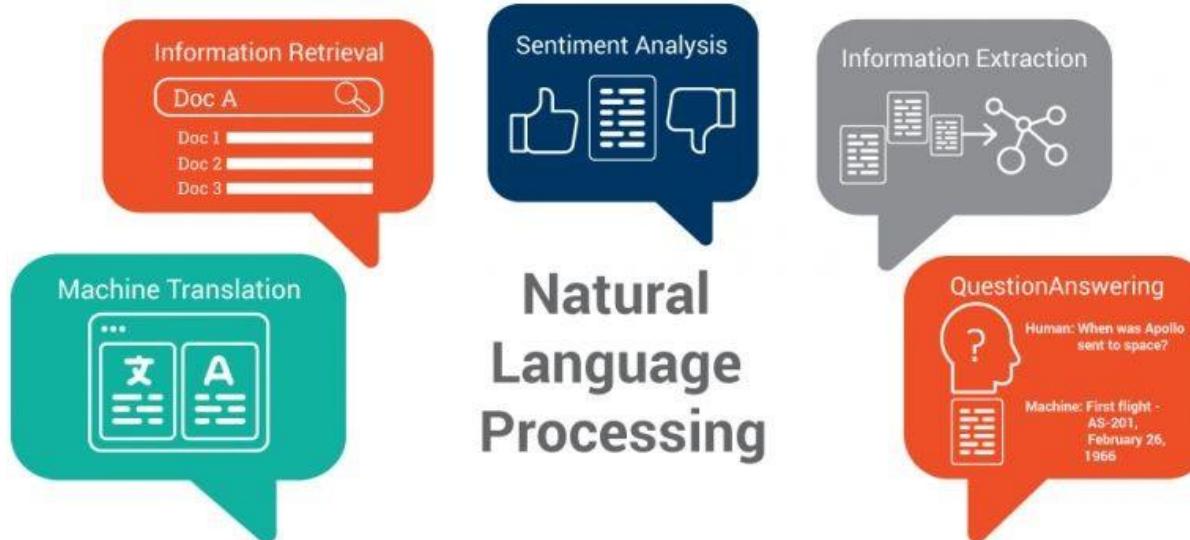
Spark Pipelines

Pipeline of annotators



```
from pyspark.ml import Pipeline
documentAssembler = DocumentAssembler()\
    .setInputCol("text")\
    .setOutputCol("document")
sentenceDetector = SentenceDetector()\
    .setInputCols(["document"])\
    .setOutputCol("sentences")
tokenizer = Tokenizer() \
    .setInputCols(["sentences"]) \
    .setOutputCol("token")
normalizer = Normalizer()\
    .setInputCols(["token"])\
    .setOutputCol("normal")
word_embeddings=WordEmbeddingsModel.pretrained()\
    .setInputCols(["document","normal"])\ 
    .setOutputCol("embeddings")
nlpPipeline = Pipeline(stages=[documentAssembler,
    sentenceDetector,
    tokenizer,
    normalizer,
    word_embeddings,
])
nlpPipeline.fit(df).transform(df)
```

NLP Basics



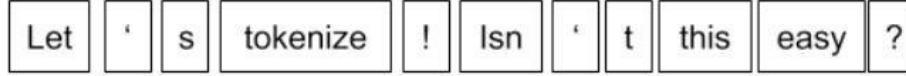
NLP Basics

Tokenization

Tokenize on
rules



Tokenize on
punctuation



Tokenize on
white spaces



- First step in NLP pipelines, raw text is broken down into smaller, more manageable pieces. Without tokenization, NLP models would have to process the text as one large unit.
- Many other tokenization methods (character, sub-word, sentence).

NLP Basics

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

Find the stem of each word.

Uses rules: e.g, remove common suffixes.

Form	Suffix	Stem
studies	-es	studi
study ^{ing}	-ing	study
niñas	-as	niñ
niñez	-ez	niñ

- The goal of both **stemming** and **lemmatization** is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form for normalization purposes.
- Lemmatization always returns real words, **stemming** doesn't.

NLP Basics

Stopword Removal

it was a bright cold day in april
and the clocks **were** striking
thirteen winston smith **his** chin
nuzzled **into his** breast in an
effort **to** escape **the** vile wind
slipped quickly **through the** glass
doors **of** victory mansions though
not quickly enough **to** prevent a
swirl **of** gritty dust **from** entering
along **with him**



bright cold day april clocks
striking thirteen winston smith
chin nuzzled breast effort
escape vile wind slipped quickly
glass doors victory mansions
though quickly enough prevent
swirl gritty dust entering along

- For tasks like text classification, where the text is to be classified into different categories, **stopwords** are **removed** or excluded from the given text so that more focus can be given to those words which define the meaning of the text. Stopwords are considered insignificant in this case.

Stopwords

a
able
about
above
according
accordingly
across
actually
after
afterwards
again
against
ain
all
allow
allows
almost
alone
along
already
also

(520 stopwords)

Pre-Processing

Spell Checking & Correction



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

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

- 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

Pre-Processing

Context Spell Checker

The Spell Checker can leverage the context of words for ranking different correction sequences. Let's take a look at some examples,

```
# check for the different occurrences of the word "siter"
example1 = ["I will call my siter.", \
            "Due to bad weather, we had to move to a different siter.", \
            "We travelled to three siter in the summer."]
beautify(lp.annotate(example1))
```

```
['I will call my sister .\n',
 'Due to bad weather , we had to move to a different site .\n',
 'We travelled to three sites in the summer .\n']
```

```
# check for the different occurrences of the word "ueather"
example2 = ["During the summer we have the best ueather.", \
            "I have a black ueather jacket, so nice.", \
            "I introduce you to my sister, she is called ueather."]
beautify(lp.annotate(example2))
```

```
['During the summer we have the best weather .\n',
 'I have a black leather jacket , so nice .\n',
 'I introduce you to my sister , she is called Heather .\n']
```

Notice that in the first example, 'siter' is indeed a valid English word,

<https://www.merriam-webster.com/dictionary/siter>

Normalization

Remove or replace undesirable characters or regular expressions:

from: @Have a\$ #2great birth) day>!

to: Have a great birth day!

Spark NLP also comes with a Slang normalizer:

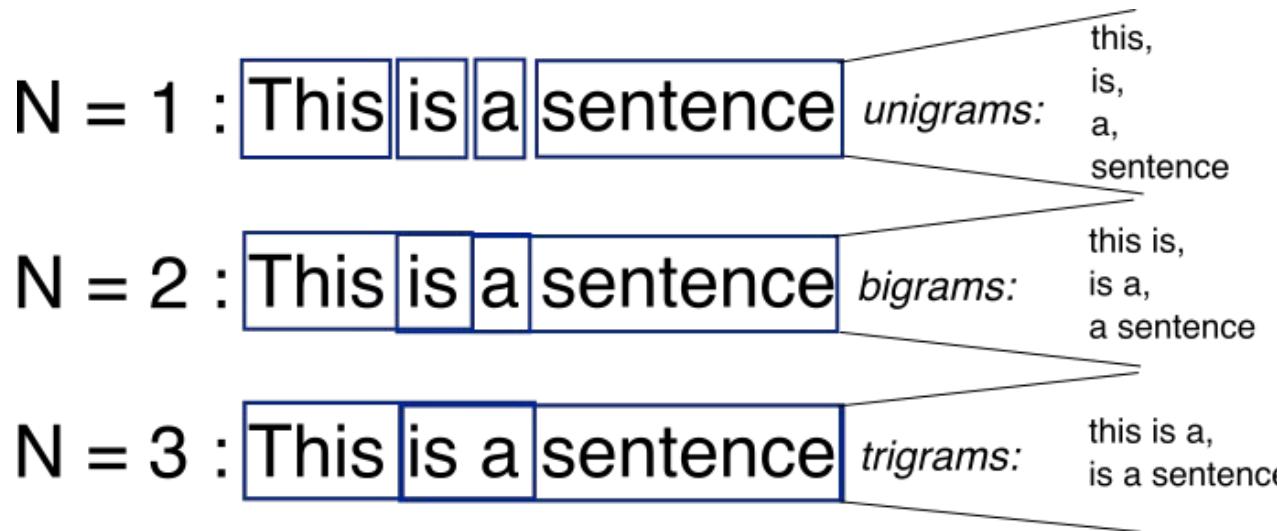
Original tweet

@USER, r u cuming 2 MidCorner dis Sunday?

Normalized tweet

@USER, are you coming to MidCorner this Sunday?

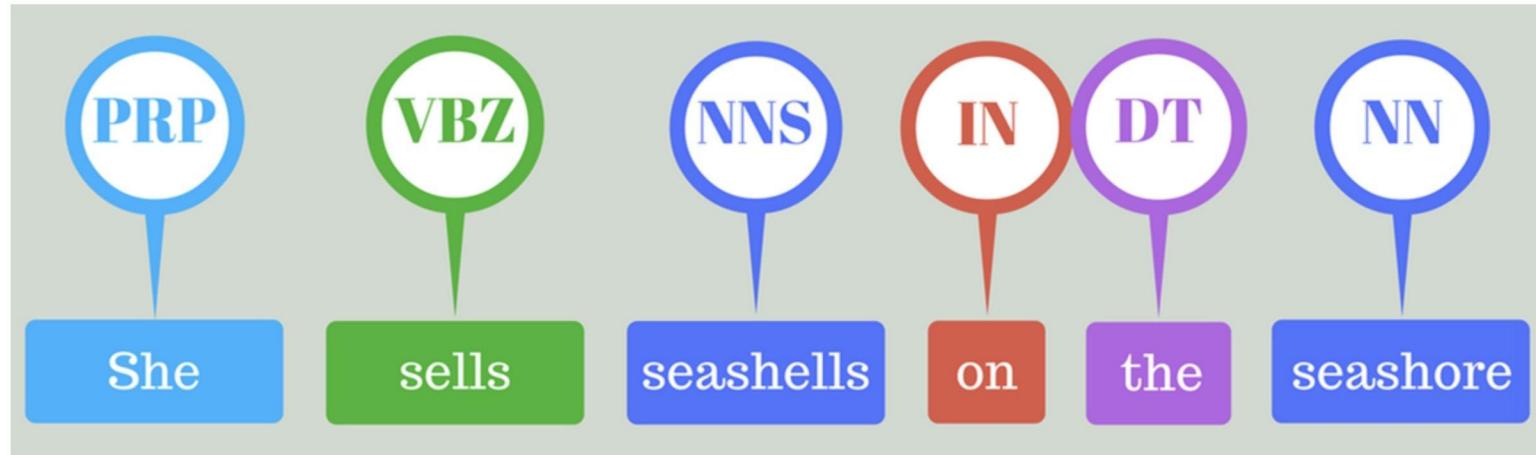
N-gram Tokenization



- Kind of tokenizers which split words or sentences into several tokens
- Each token has certain number of words
- Number of words depends on the type of n-gram tokenizer
- Unigram, bigram, trigram, etc.

Part Of Speech Tagging

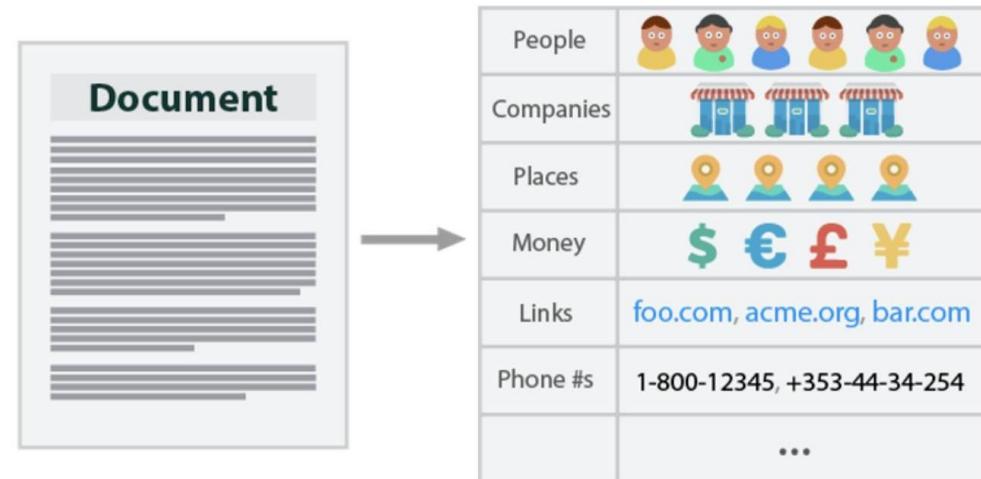
Often useful for recognizing named entities or word relationships.



A **POS tag** (or **part-of-speech tag**) is a special label assigned to each token (word) in a text corpus to indicate the **part of speech** and often also other grammatical categories such as tense, number (plural/singular), case etc.

Named Entity Recognition (NER)

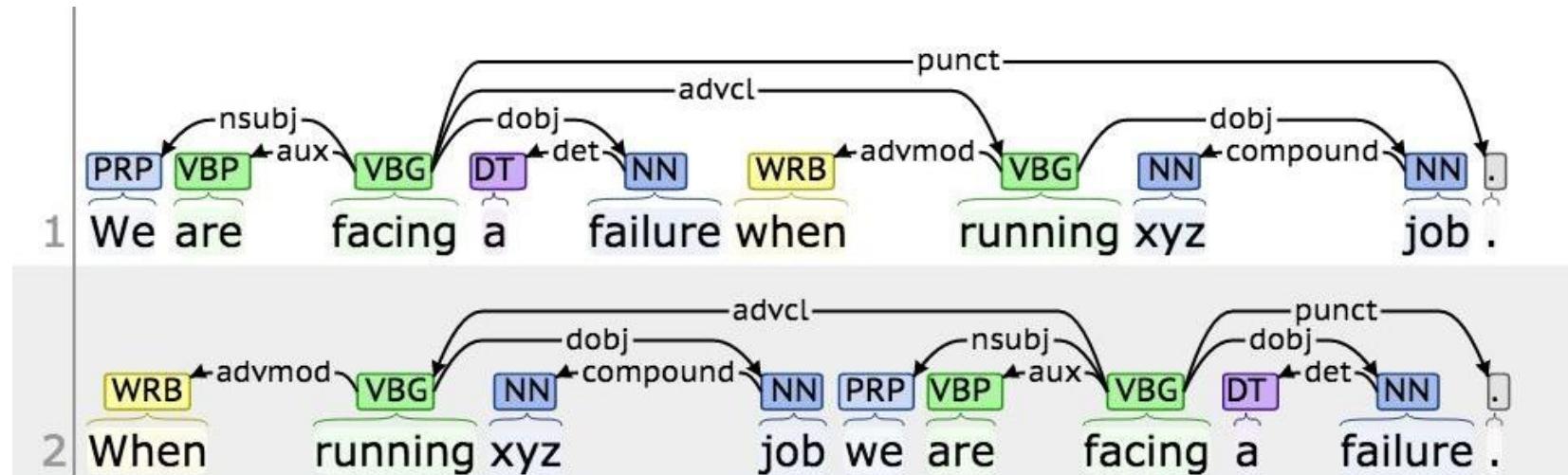
NER is a subtask of information extraction that seeks to **locate and classify named entity** mentioned in unstructured text into pre-defined categories such as **person** names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc.



But Google **ORG** is starting from behind. The company made a late push into hardware, and Apple **ORG**'s Siri **PRODUCT**, available on iPhones **PRODUCT**, and Amazon **ORG**'s Alexa **PRODUCT** software, which runs on its Echo **PRODUCT** and Dot **PRODUCT** devices, have clear leads in consumer adoption.

Dependency Parsing

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



Session 1 (Day 1) - Coding Time

- ❖ [Notebook 1 - Spark NLP Basics](#)

Session 2 (Day 1)

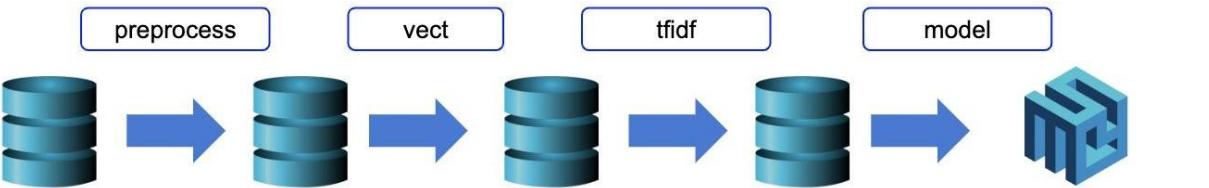
- ❖ Text Preprocessing
- ❖ Composing Pipelines
- ❖ Working with Spark Dataframes

Spark NLP
for Data Scientists

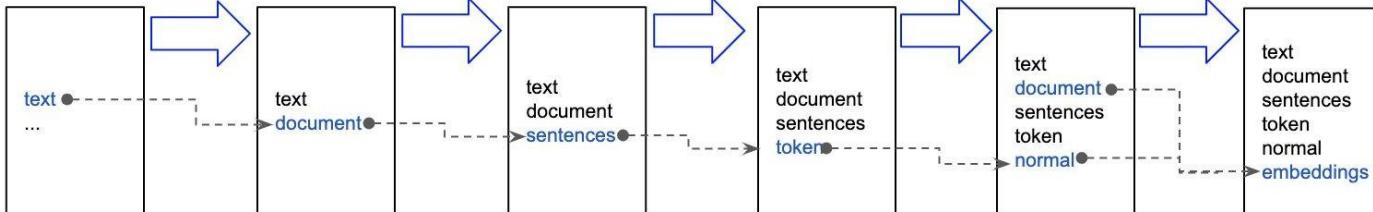


Pipeline Structure

Pipeline of annotators



DocumentAssembler() SentenceDetector() Tokenizer() Normalizer() WordEmbeddings()



DataFrame

```
from pyspark.ml import Pipeline
document_assembler = DocumentAssembler()\
    .setInputCol("text")\
    .setOutputCol("document")
sentenceDetector = SentenceDetector()\
    .setInputCols(["document"])\
    .setOutputCol("sentences")
tokenizer = Tokenizer() \
    .setInputCols(["sentences"]) \
    .setOutputCol("token")
normalizer = Normalizer()\
    .setInputCols(["token"])\
    .setOutputCol("normal")
word_embeddings=WordEmbeddingsModel.pretrained()\
    .setInputCols(["document", "normal"])\ 
    .setOutputCol("embeddings")
nlpPipeline = Pipeline(stages=[document_assembler,
    sentenceDetector,
    tokenizer,
    normalizer,
    word_embeddings,
])
nlpPipeline.fit(df).transform(df)
```

<https://spark.apache.org/docs/1.6.0/ml-guide.html#pipeline-components>

Session 2 (Day 1) - Coding Time

- ❖ Notebook 2 Text Preprocessing and composing Pipelines

Session 3 (Day 1)

- ❖ Usage and overview of the 10000+ pretrained models for 300+ languages

Spark NLP
for Data Scientists



Spark NLP Modules

Entity Recognition 	Text Classification 	Spelling & Grammar 	Information Extraction 
Question Answering 	Speech to Text 	Image Classification 	Reading Comprehension 
Translation 	Summarization 	Paraphrasing 	Emotion Detection 

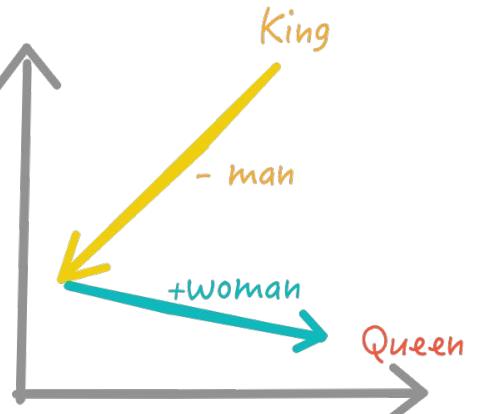
Split Text <ul style="list-style-type: none">Sentence DetectorTokenizerNormalizernGram GeneratorWord Segmentation	Clean Text <ul style="list-style-type: none">Spell CheckerGrammar CheckerWriting Style CheckerStopword CleanerSummarization	20,000+ Pre-trained Pipelines, Models & Transformers <ul style="list-style-type: none">BERTELMOTAPASALBERTDeBERTaUSELongformerELECTRAT5NMTVITDistilBERTRoBERTaXLM-RoBERTaWav2Vec2XLNet	250+ Languages <ul style="list-style-type: none">Flag icons for various languages including Chinese, English, Spanish, French, German, Indian, Korean, Japanese, Vietnamese, Russian, Turkish, etc.	
Understand Grammar <ul style="list-style-type: none">StemmerLemmatizerPart of Speech TaggerDependency ParserTranslation	Find in Text <ul style="list-style-type: none">Text MatcherRegex MatcherDate MatcherChunkerQuestion Answering			
Trainable & Tunable 	Scalable 	Fast Inference 	Hardware Optimized 	Community 

Word & Sentence Embeddings

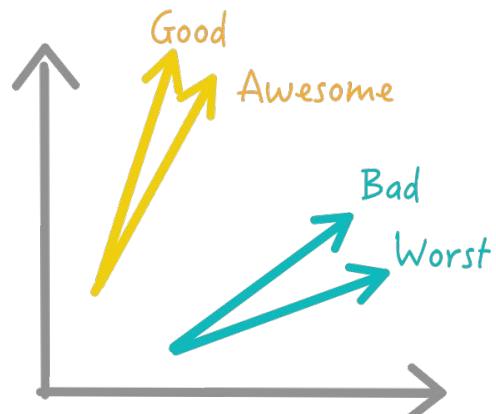
- Neural Networks and most other algorithms don't work well with strings
 - Why not convert a string into a vector with numbers!
 - Semantically similar words vectors should be **close** in the vector space learned by the model
 - Dissimilar words are **far** from each other in vector space
 - Usually high dimensional, to give model more space to encode semantics
- Useful for many downstream tasks and commonly used in NLP
- Reduce dimensionality of the input space

Raw Text	Bag-of-words vector
it	2
they	0
puppy	1
and	1
cat	0
aardvark	0
cute	1
extremely	1
...	...

it is a puppy and it
is extremely cute



a) Learns Analogy



b) Similar Words have same angles

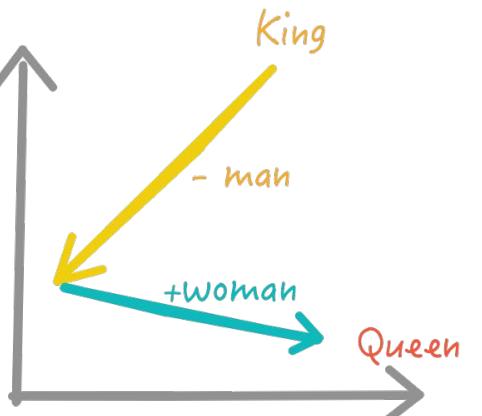
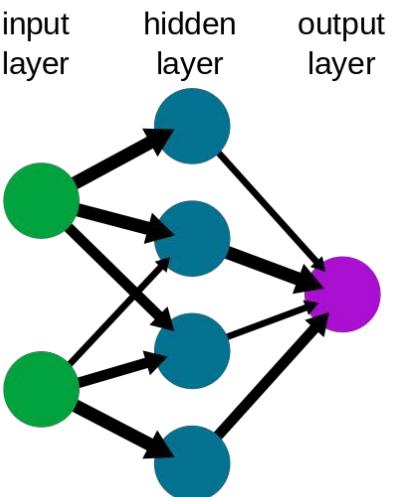
```
In [9]: doc[3].vector
Out[9]: array([-0.037103, -0.31259, -0.17857, 0.30001, 0.078154,
   -0.17958, 0.12048, -0.11879, -0.20601, 1.2849,
   -0.20409, 0.80613, 0.34344, -0.19191, -0.084511,
   0.17339, 0.042483, 2.0282, -0.16278, -0.60306,
   -0.53766, 0.35711, 0.22882, 0.1171, 0.42983,
   0.16165, 0.407, 0.036476, 0.52636, -0.13524,
   -0.016897, 0.029259, -0.079115, -0.32305, 0.052255,
   -0.3617, -0.1821, -0.09891, -0.0591, 0.16881,
   0.2118, -0.18376, -0.098909, -0.38206, -0.26935,
   0.0021152, -0.32512, 0.063977, 0.36249, 0.36249,
   -0.59341, -0.13625, 0.016425, -0.2474, -0.07498,
   0.034708, -0.01476, -0.11648, 0.25559, -0.35002,
   -0.52707, -0.21221, 0.062456, 0.26184, 0.53149,
   0.34957, -0.22692, 0.440076, 0.4438, 0.6335,
   -0.049757, -0.08134, 0.65618, -0.4716, 0.090675,
   -0.084873, 0.31455, -0.38495, -0.19247, 0.48064,
   0.26688, 0.095743, 0.13024, 0.37023, 0.46269,
   -0.32844, 0.17375, -0.36325, 0.30672, -0.075042,
   -0.64688, -0.49822, 0.12373, -0.28547, 0.61811,
   -0.19228, 0.0040473, 0.1774, 0.033154, -0.54862,
   0.34695, -0.53506, -0.013381, 0.085712, -0.054447,
   -0.64673, 0.016749, 0.47676, 0.037803, -0.10066,
   -0.4165, -0.20252, 0.2794, 0.10852, -0.40154,
```

Word & Sentence Embeddings

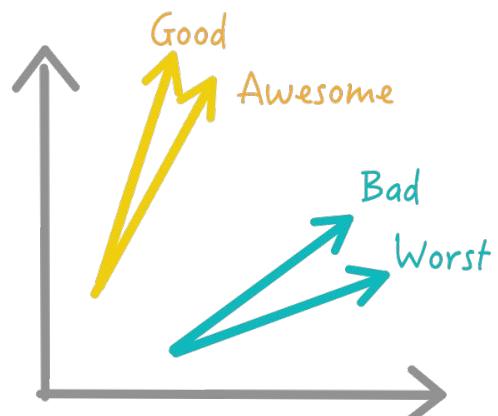
- Deep-Learning-based natural language processing systems encode **words** and **sentences** 📄 in fixed-length dense vectors to drastically improve the processing of textual data.
- Based on **The Distributional Hypothesis**: Words that occur in the same contexts tend to have similar meanings.
- Elmo and Bert-family embeddings are context-aware.



A simple neural network



a) Learns Analogy



b) Similar Words have same angles

```
In [9]: doc[3].vector
Out[9]: array([-0.037103, -0.31259, -0.17857, 0.30001, 0.078154,
   -0.17958, 0.12048, -0.11879, -0.20601, 1.2849,
   -0.20409, 0.80613, 0.34344, -0.19191, -0.084511,
   0.17339, 0.042483, 2.0282, -0.16278, -0.60306,
   -0.53766, 0.35711, 0.22882, 0.1171, 0.42983,
   0.16165, 0.407, 0.036476, 0.52636, -0.13524,
   -0.016897, 0.029259, -0.079115, -0.32305, 0.052255,
   -0.3617, -0.1821, -0.098909, -0.0591, 0.16881,
   0.21218, -0.18376, -0.098909, -0.32305, -0.26935,
   0.0021159, -0.32512, 0.063977, 0.36249,
   -0.59341, -0.13625, 0.016425, -0.2474, -0.07498,
   0.034708, -0.01476, -0.11648, 0.25559,
   -0.52707, 0.21221, 0.062456, 0.26184, 0.53149,
   0.34957, -0.22692, 0.440705, 0.4438, 0.6335,
   -0.049757, -0.08134, 0.65618, -0.4716, 0.090675,
   -0.084873, 0.31455, -0.38495, -0.19247, 0.48064,
   0.26688, 0.095743, 0.13024, 0.37023, 0.46269,
   -0.32844, 0.17375, -0.36325, 0.30672, -0.075042,
   -0.64688, -0.49822, 0.12373, -0.28547, 0.61811,
   -0.19228, 0.0040473, 0.1774, 0.033154, -0.54862,
   0.34695, -0.53506, -0.013381, 0.085712, -0.054447,
   -0.64673, 0.016749, 0.47676, 0.037803, -0.10066,
   -0.4165, -0.20252, 0.2794, 0.10852, -0.40154,
```

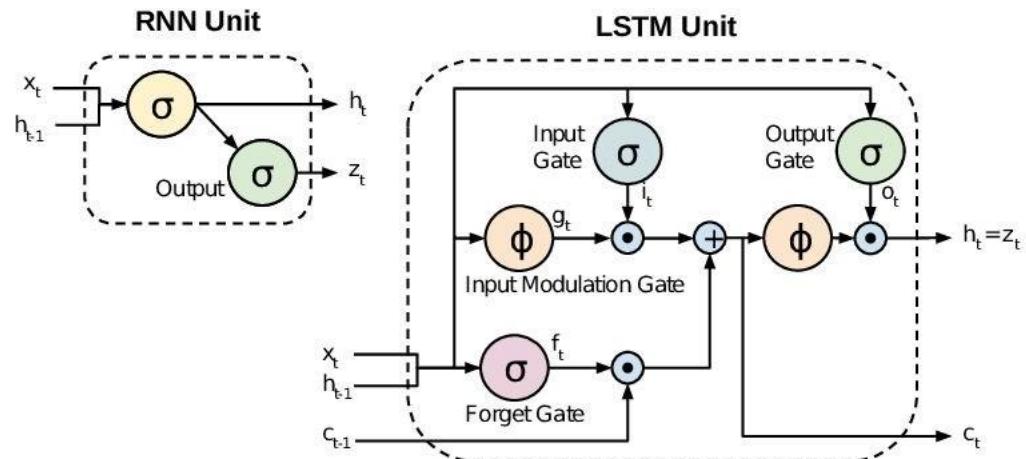
LSTM's and RNN's for Embeddings

- RNN :

- Slow to train
- Not well parallelizable
- Vanishing/Exploding Gradients

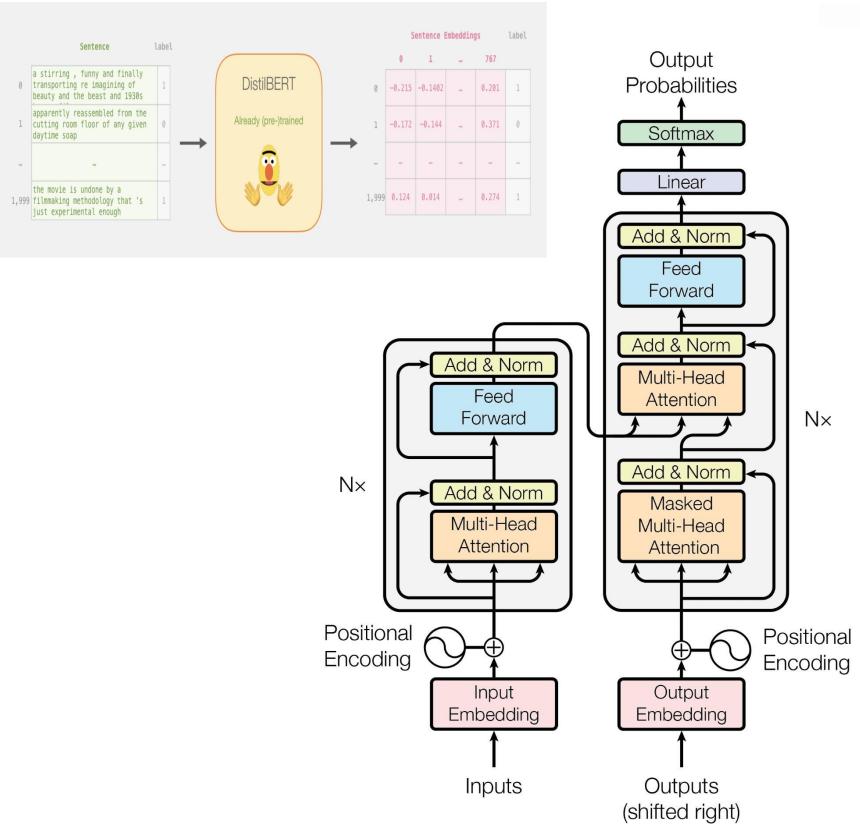
- LSTM

- Fixes vanishing/exploding gradient
- not scalable
- No transfer learning



Transformers - Attention is All you Need - Visualized

A highly parallelizable NN Architecture for Sequential Data



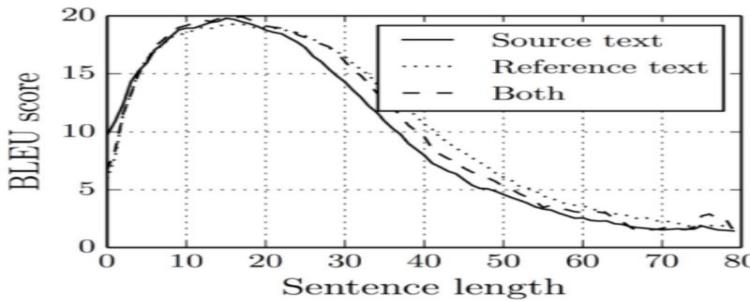
<https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

Feature-Engineering

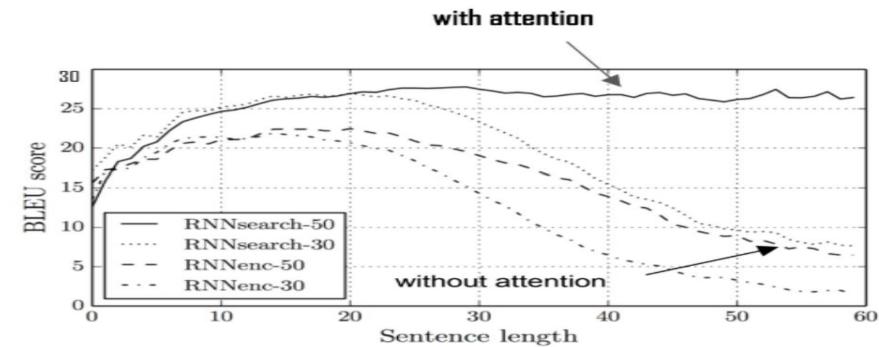
Transformers - Attention is All you Need - Why they rock

A highly parallelizable NN Architecture for Sequential Data

NMT with LSTMs



NMT with LSTMs + attention



Bahdanau et al. 2014

Slide: Text generation with attention, GTC 2017, Valentin Malykh (2017)

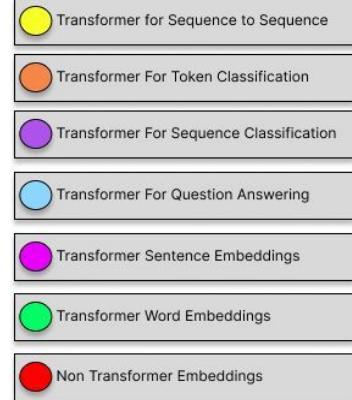
Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(\log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

The most scalable, accurate and fastest arsenal of NLP transformers in history



Core NLP Transformer Models

CamemBertEmbeddings	GloveEmbeddings	Word2VecEmbeddings	Doc2VecEmbeddings
DeBertaEmbeddings	LongformerEmbeddings	BertEmbeddings	XlmRoBERTaEmbeddings
DeBertaForQuestionAnswering	LongformerForTokenClassification	BertForTokenClassification	XlmRoBERTaForTokenClassification
GPT2Transformer	LongformerForSequenceClassification	BertForSequenceClassification	XlmRoBERTaForSequenceClassification
MarianTransformer	LongformerForQuestionAnswering	BertForQuestionAnswering	XlmRoBERTaForQuestionAnswering
T5Transformer	UniversalSentenceEncoder	BertSentenceEmbeddings	XlmRoBERTaSentenceEmbeddings
RoBERTaEmbeddings	XlnetEmbeddings	DistilBertEmbeddings	AlbertEmbeddings
RoBERTaForTokenClassification	XlnetForTokenClassification	DistilBertForTokenClassification	AlbertForTokenClassification
RoBERTaForSequenceClassification	XlnetForSequenceClassification	DistilBertForSequenceClassification	AlbertForSequenceClassification
RoBERTaForQuestionAnswering	ElmoEmbeddings	DistilBertForQuestionAnswering	AlbertForQuestionAnswering



Subflavored NLP Transformer Extensions

SciBertEmbeddings	ElectraForQuestionAnswering	MurilBertForQuestionAnswering	MurilBertEmbeddings	LapseSentenceEmbeddings
IndoBertForQuestionAnswering	ElectraSentenceEmbeddings	MurilBertSentenceEmbeddings	SciBertForQuestionAnswering	SapBertForQuestionAnswering
ElectraEmbeddings	BioBertForQuestionAnswering	CovidBertForQuestionAnswering	MiniLmForQuestionAnswering	MacBertForQuestionAnswering
BioFormerForQuestionAnswering	BioBertSentenceEmbeddings	CovidDistilBertForQuestionAnswering	LinkBertForQuestionAnswering	BioBertEmbeddings

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
0	en	glove_100d	WordEmbeddingsModel
1	xx	glove_6B_300	WordEmbeddingsModel
2	xx	glove_840B_300	WordEmbeddingsModel
3	en	embeddings_clinical	WordEmbeddingsModel
4	en	elmo	ElmoEmbeddings
5	en	embeddings_healthcare	WordEmbeddingsModel
6	en	tflite_use	UniversalSentenceEncoder
7	en	tflite_use_lg	UniversalSentenceEncoder
8	en	albert_base_uncased	AlbertEmbeddings
9	en	albert_large_uncased	AlbertEmbeddings
10	en	albert_xlarge_uncased	AlbertEmbeddings
11	en	albert_xxlarge_uncased	AlbertEmbeddings
12	en	xlnet_base_cased	XlnetEmbeddings
13	en	xlnet_large_cased	XlnetEmbeddings
14	es	embeddings_scielo_150d	WordEmbeddingsModel
15	es	embeddings_scielo_300d	WordEmbeddingsModel
16	es	embeddings_scielo_50d	WordEmbeddingsModel
17	es	embeddings_scielowiki_150d	WordEmbeddingsModel
18	es	embeddings_scielowiki_300d	WordEmbeddingsModel
19	es	embeddings_scielowiki_50d	WordEmbeddingsModel
20	es	embeddings_scifiwiki_150d	WordEmbeddingsModel
21	es	embeddings_scifiwiki_300d	WordEmbeddingsModel
22	es	embeddings_scifiwiki_50d	WordEmbeddingsModel
23	en	embeddings_healthcare_100d	WordEmbeddingsModel
24	en	embeddings_biovec	WordEmbeddingsModel
25	en	bert_base_cased	BertEmbeddings

Embeddings in Spark NLP

Part 1

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
26	en	bert_base_uncased	BertEmbeddings
27	en	bert_large_cased	BertEmbeddings
28	en	bert_large_uncased	BertEmbeddings
29	xx	bert_multi_cased	BertEmbeddings
30	en	biobert_clinical_base_cased	BertEmbeddings
31	en	biobert_discharge_base_cased	BertEmbeddings
32	en	biobert_pubmed_base_cased	BertEmbeddings
33	en	biobert_pubmed_cased	BertEmbeddings
34	en	biobert_pubmed_large_cased	BertEmbeddings
35	en	biobert_pubmed_pubmed_cased	BertEmbeddings
36	en	sent.biobert_clinical_base_cased	BertSentenceEmbeddings
37	en	sent.bert_base_cased	BertSentenceEmbeddings
38	en	sent.bert_base_uncased	BertSentenceEmbeddings
39	en	sent.bert_large_cased	BertSentenceEmbeddings
40	en	sent.bert_large_uncased	BertSentenceEmbeddings
41	xx	sent.bert_multi_cased	BertSentenceEmbeddings
42	en	sent.biobert_discharge_base_cased	BertSentenceEmbeddings
43	en	sent.biobert_pubmed_base_cased	BertSentenceEmbeddings
44	en	sent.biobert_pubmed_cased	BertSentenceEmbeddings
45	en	sent.biobert_pubmed_large_cased	BertSentenceEmbeddings
46	en	sent.biobert_pubmed_pubmed_cased	BertSentenceEmbeddings
47	en	sent.small.bert.L10.128	BertSentenceEmbeddings
48	en	sent.small.bert.L10.256	BertSentenceEmbeddings
49	en	sent.small.bert.L10.512	BertSentenceEmbeddings
50	en	sent.small.bert.L10.768	BertSentenceEmbeddings

Number	Language	Spark NLP Model Name	Model
51	en	sent_small_bert_L12_128	BertSentenceEmbeddings
52	en	sent_small_bert_L12_256	BertSentenceEmbeddings
53	en	sent_small_bert_L12_512	BertSentenceEmbeddings
54	en	sent_small_bert_L12_768	BertSentenceEmbeddings
55	en	sent_small_bert_L2_128	BertSentenceEmbeddings
56	en	sent_small_bert_L2_256	BertSentenceEmbeddings
57	en	sent_small_bert_L2_512	BertSentenceEmbeddings
58	en	sent_small_bert_L2_768	BertSentenceEmbeddings
59	en	sent_small_bert_L4_128	BertSentenceEmbeddings
60	en	sent_small_bert_L4_256	BertSentenceEmbeddings
61	en	sent_small_bert_L4_512	BertSentenceEmbeddings
62	en	sent_small_bert_L4_768	BertSentenceEmbeddings
63	en	sent_small_bert_L6_128	BertSentenceEmbeddings
64	en	sent_small_bert_L6_256	BertSentenceEmbeddings
65	en	sent_small_bert_L6_512	BertSentenceEmbeddings
66	en	sent_small_bert_L6_768	BertSentenceEmbeddings
67	en	sent_small_bert_L8_128	BertSentenceEmbeddings
68	en	sent_small_bert_L8_256	BertSentenceEmbeddings
69	en	sent_small_bert_L8_512	BertSentenceEmbeddings
70	en	sent_small_bert_L8_768	BertSentenceEmbeddings
71	en	small_bert_L10_128	BertEmbeddings
72	en	small_bert_L10_256	BertEmbeddings
73	en	small_bert_L10_512	BertEmbeddings
74	en	small_bert_L10_768	BertEmbeddings
75	en	small_bert_L12_128	BertEmbeddings

Embeddings in Spark NLP Part 2

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
76	en	small_bert_L12_256	BertEmbeddings
77	en	small_bert_L12_512	BertEmbeddings
78	en	small_bert_L12_768	BertEmbeddings
79	en	small_bert_L2_128	BertEmbeddings
80	en	small_bert_L2_256	BertEmbeddings
81	en	small_bert_L2_512	BertEmbeddings
82	en	small_bert_L2_768	BertEmbeddings
83	en	small_bert_L4_128	BertEmbeddings
84	en	small_bert_L4_256	BertEmbeddings
85	en	small_bert_L4_512	BertEmbeddings
86	en	small_bert_L4_768	BertEmbeddings
87	en	small_bert_L6_128	BertEmbeddings
88	en	small_bert_L6_256	BertEmbeddings
89	en	small_bert_L6_512	BertEmbeddings
90	en	small_bert_L6_768	BertEmbeddings
91	en	small_bert_L8_128	BertEmbeddings
92	en	small_bert_L8_256	BertEmbeddings
93	en	small_bert_L8_512	BertEmbeddings
94	en	small_bert_L8_768	BertEmbeddings
95	en	covidbert_large_uncased	BertEmbeddings
96	en	electra_base_uncased	BertEmbeddings
97	en	electra_large_uncased	BertEmbeddings
98	en	electra_small_uncased	BertEmbeddings
99	en	sent_covidbert_large_uncased	BertSentenceEmbeddings
100	en	sent_electra_base_uncased	BertSentenceEmbeddings

Number	Language	Spark NLP Model Name	Model
101	en	sent_electra_large_uncased	BertSentenceEmbeddings
102	en	sent_electra_small_uncased	BertSentenceEmbeddings
103	fi	bert_finnish_cased	BertEmbeddings
104	fi	bert_finnish_uncased	BertEmbeddings
105	de	w2v_cc_300d	WordEmbeddingsModel
106	en	biobert_clinical_base_cased	BertEmbeddings
107	en	biobert_discharge_base_cased	BertEmbeddings
108	en	biobert_pmc_base_cased	BertEmbeddings
109	en	biobert_pubmed_base_cased	BertEmbeddings
110	en	biobert_pubmed_large_cased	BertEmbeddings
111	en	biobert_pubmed_pmc_base_cased	BertEmbeddings
112	en	sent.biobert_clinical_base_cased	BertSentenceEmbeddings
113	en	sent.biobert_discharge_base_cased	BertSentenceEmbeddings
114	en	sent.biobert_pmc_base_cased	BertSentenceEmbeddings
115	en	sent.biobert_pubmed_base_cased	BertSentenceEmbeddings
116	en	sent.biobert_pubmed_large_cased	BertSentenceEmbeddings
117	en	sent.biobert_pubmed_pmc_base_cased	BertSentenceEmbeddings
118	en	labse	BertSentenceEmbeddings
119	pt	bert_portuguese_base_cased	BertEmbeddings
120	pt	bert_portuguese_large_cased	BertEmbeddings
121	en	sbiobert_base_cased_mli	BertSentenceEmbeddings
122	en	sbluebert_base_uncased_mli	BertSentenceEmbeddings
123	ur	urduvec_140M_300d	WordEmbeddingsModel
124	ar	arabic_w2v_cc_300d	WordEmbeddingsModel
125	fa	persian_w2v_cc_300d	WordEmbeddingsModel

Language	Spark NLP Model Name	Model
126	xx	tflhub_use_multi_lg
127	xx	tflhub_use_multi
128	he	hebrew_cc_300d
129	hi	hindi_cc_300d
130	bn	bengali_cc_300d
131	xx	tflhub_use_multi_lg
132	xx	tflhub_use_multi
133	en	sbert_jsl_medium_umls_uncased
134	en	sbert_jsl_medium_uncased
135	en	sbert_jsl_mini_umls_uncased
136	en	sbert_jsl_mini_uncased
137	en	sbert_jsl_tiny_umls_uncased
138	en	sbert_jsl_tiny_uncased
139	en	sbiobert_jsl_cased
140	en	sbiobert_jsl_umls_cased
141	zh	bert_base_chinese
142	nl	bert_base_dutch_cased
143	de	bert_base_german_cased
144	de	bert_base_german_uncased
145	it	bert_base_italian_cased
146	it	bert_base_italian_uncased
147	xx	bert_base_multilingual_cased
148	xx	bert_base_multilingual_uncased
149	tr	bert_base_turkish_cased
150	tr	bert_base_turkish_uncased

Embeddings in Spark NLP

Part 3

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
151	zh	chinese_bert_wwm	BertEmbeddings
152	en	distilbert_base_cased	DistilBertEmbeddings
153	xx	distilbert_base_multilingual_cased	DistilBertEmbeddings
154	en	distilbert_base_uncased	DistilBertEmbeddings
155	en	distilroberta_base	RoBERTaEmbeddings
156	en	roberta_base	RoBERTaEmbeddings
157	en	roberta_large	RoBERTaEmbeddings
158	xx	twitter_xlm_roberta_base	XLMRoBERTaEmbeddings
159	xx	xlm_roberta_base	XLMRoBERTaEmbeddings
160	en	albert_base_uncased	AlbertEmbeddings
161	en	albert_large_uncased	AlbertEmbeddings
162	en	albert_xlarge_uncased	AlbertEmbeddings
163	en	albert_xxlarge_uncased	AlbertEmbeddings
164	en	sbert_jsl_medium_umls_uncased	BertSentenceEmbeddings
165	en	sbert_jsl_medium_uncased	BertSentenceEmbeddings
166	en	sbert_jsl_mini_umls_uncased	BertSentenceEmbeddings
167	en	sbert_jsl_mini_uncased	BertSentenceEmbeddings
168	en	sbert_jsl_tiny_umls_uncased	BertSentenceEmbeddings
169	en	sbert_jsl_tiny_uncased	BertSentenceEmbeddings
170	en	sbiobert_jsl_cased	BertSentenceEmbeddings
171	en	sbiobert_jsl_umls_cased	BertSentenceEmbeddings
172	zh	chinese_xlnet_base	XlnetEmbeddings
173	en	xlnet_base_cased	XlnetEmbeddings
174	en	xlnet_large_cased	XlnetEmbeddings
175	xx	xlm_roberta_xtreme_base	XLMRoBERTaEmbeddings

Embeddings in Spark NLP

Part 4

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
176	en	sent_bert_use_cm_lm_en_base	BertSentenceEmbeddings
177	en	sent_bert_use_cm_lm_en_large	BertSentenceEmbeddings
178	xx	sent_bert_use_cm_lm_multi_base_br	BertSentenceEmbeddings
179	xx	sent_bert_use_cm_lm_multi_base	BertSentenceEmbeddings
180	en	longformer_base_4096	LongformerEmbeddings
181	en	longformer_large_4096	LongformerEmbeddings
182	xx	bert_muril	BertEmbeddings
183	en	bert_pubmed	BertEmbeddings
184	en	bert_pubmed_squad2	BertEmbeddings
185	en	bert_wiki_books	BertEmbeddings
186	en	bert_wiki_books_mnli	BertEmbeddings
187	en	bert_wiki_books_qnli	BertEmbeddings
188	en	bert_wiki_books_qqp	BertEmbeddings
189	en	bert_wiki_books_squad2	BertEmbeddings
190	en	bert_wiki_books_sst2	BertEmbeddings
191	te	sentence_detector_dl	SentenceDetectorDLModel
192	en	sent_bert_pubmed	BertSentenceEmbeddings
193	en	sent_bert_pubmed_squad2	BertSentenceEmbeddings
194	en	sent_bert_wiki_books	BertSentenceEmbeddings
195	en	sent_bert_wiki_books_mnli	BertSentenceEmbeddings
196	en	sent_bert_wiki_books_qnli	BertSentenceEmbeddings
197	en	sent_bert_wiki_books_qqp	BertSentenceEmbeddings
198	en	sent_bert_wiki_books_squad2	BertSentenceEmbeddings
199	en	sent_bert_wiki_books_sst2	BertSentenceEmbeddings
200	xx	sent_bert_muril	BertSentenceEmbeddings

Language	Spark NLP Model Name	Model
Number		
201	xx	sent_xlm_roberta_base XlmRoBERTaForTokenClassification
202	es	sent_bert_base_cased BertSentenceEmbeddings
203	nl	sent_bert_base_cased BertSentenceEmbeddings
204	sv	sent_bert_base_cased BertSentenceEmbeddings
205	el	sent_bert_base_uncased BertSentenceEmbeddings
206	es	sent_bert_base_uncased BertSentenceEmbeddings
207	en	sent_bert_base_uncased_legal BertSentenceEmbeddings
208	es	bert_base_cased BertEmbeddings
209	nl	bert_base_cased BertEmbeddings
210	sv	bert_base_cased BertEmbeddings
211	el	bert_base_uncased BertEmbeddings
212	es	bert_base_uncased BertEmbeddings
213	en	bert_base_uncased_legal BertEmbeddings
214	ja	japanese_cc_300d WordEmbeddingsModel
215	ro	bert_base_cased BertEmbeddings
216	de	sent_bert_base_cased BertSentenceEmbeddings
217	am	xlm_roberta_base_finetuned_amharic XlmRoBERTaEmbeddings
218	ha	xlm_roberta_base_finetuned_hausa XlmRoBERTaEmbeddings
219	ig	xlm_roberta_base_finetuned_igbo XlmRoBERTaEmbeddings
220	rw	xlm_roberta_base_finetuned_kinyarwanda XlmRoBERTaEmbeddings
221	lg	xlm_roberta_base_finetuned_luganda XlmRoBERTaEmbeddings
222	xx	xlm_roberta_large XlmRoBERTaEmbeddings
223	am	sent_xlm_roberta_base_finetuned_amharic XlmRoBERTaSentenceEmbeddings
224	ha	sent_xlm_roberta_base_finetuned_hausa XlmRoBERTaSentenceEmbeddings
225	ig	sent_xlm_roberta_base_finetuned_igbo XlmRoBERTaSentenceEmbeddings

Embeddings in Spark NLP Part 5

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Language	Spark NLP Model Name	Model
Number		
226	rw	sent_xlm_roberta_base_finetuned_kinyarwanda XlmRoBERTaSentenceEmbeddings
227	lg	sent_xlm_roberta_base_finetuned_luganda XlmRoBERTaSentenceEmbeddings
228	pcm	sent_xlm_roberta_base_finetuned_naija XlmRoBERTaSentenceEmbeddings
229	sw	sent_xlm_roberta_base_finetuned_swahili XlmRoBERTaSentenceEmbeddings
230	wo	sent_xlm_roberta_base_finetuned_wolof XlmRoBERTaSentenceEmbeddings
231	yo	sent_xlm_roberta_base_finetuned_yoruba XlmRoBERTaSentenceEmbeddings
232	lou	xlm_roberta_base_finetuned_luo XlmRoBERTaEmbeddings
233	pcm	xlm_roberta_base_finetuned_naija XlmRoBERTaEmbeddings
234	sw	xlm_roberta_base_finetuned_swahili XlmRoBERTaEmbeddings
235	wo	xlm_roberta_base_finetuned_wolof XlmRoBERTaEmbeddings
236	yo	xlm_roberta_base_finetuned_yoruba XlmRoBERTaEmbeddings
237	es	roberta_base_biomedical RoBERTaEmbeddings
238	en	doc2vec_gigaword_300 Doc2VecModel
239	en	doc2vec_gigaword_wiki_300 Doc2VecModel
240	te	distilbert_uncased DistilBertEmbeddings
241	en	sbert_jsl_medium_rxnorm_uncased BertSentenceEmbeddings
242	fi	bert_base_finnish_cased BertSentenceEmbeddings
243	fi	bert_base_finnish_uncased BertSentenceEmbeddings
244	en	sbert_jsl_medium_rxnorm_uncased BertSentenceEmbeddings
245	en	word2vec_gigaword_300 Word2VecModel
246	en	word2vec_gigaword_wiki_300 Word2VecModel
247	en	electra_medal_acronym BertEmbeddings
248	vi	distilbert_base_cased DistilBertEmbeddings

Language	Spark NLP Model Name	Model	Language	Spark NLP Model Name	Model
Number			Number		
0	en bert_base_sequence_classifier_dbpedia_14	BertForSequenceClassification	25	en distilbert_base_sequence_classifier_imdb	DistilBertForSequenceClassification
1	en bert_base_sequence_classifier_imdb	BertForSequenceClassification	26	ur distilbert_base_sequence_classifier_imdb	DistilBertForSequenceClassification
2	en bert_large_sequence_classifier_imdb	BertForSequenceClassification	27	fr distilbert_multilingual_sequence_classifier_al...	DistilBertForSequenceClassification
3	fr bert_multilingual_sequence_classifier.allocine	BertForSequenceClassification	28	en distilbert_sequence_classifier_banking77	DistilBertForSequenceClassification
4	en bert_base_sequence_classifier_ag_news	BertForSequenceClassification	29	en distilbert_sequence_classifier_emotion	DistilBertForSequenceClassification
5	es bert_sequence_classifier_beto_emotion_analysis	BertForSequenceClassification	30	en distilbert_sequence_classifier_industry	DistilBertForSequenceClassification
6	es bert_sequence_classifier_beto_sentiment_analysis	BertForSequenceClassification	31	en distilbert_sequence_classifier_policy	DistilBertForSequenceClassification
7	en bert_sequence_classifier_dehatebert_mono	BertForSequenceClassification	32	en distilbert_sequence_classifier_sst2	DistilBertForSequenceClassification
8	en bert_sequence_classifier_firbert	BertForSequenceClassification	33	en albert_base_sequence_classifier_ag_news	AlbertForSequenceClassification
9	ja bert_sequence_classifier_japanese_sentiment	BertForSequenceClassification	34	en albert_base_sequence_classifier_imdb	AlbertForSequenceClassification
10	xx bert_sequence_classifier_multilingual_sentiment	BertForSequenceClassification	35	en longformer_base_sequence_classifier_ag_news	LongformerForSequenceClassification
11	ru bert_sequence_classifier_rubert_sentiment	BertForSequenceClassification	36	en longformer_base_sequence_classifier_imdb	LongformerForSequenceClassification
12	de bert_sequence_classifier_sentiment	BertForSequenceClassification	37	en roberta_base_sequence_classifier_ag_news	RoBERTaForSequenceClassification
13	tr bert_sequence_classifier_turkish_sentiment	BertForSequenceClassification	38	en roberta_base_sequence_classifier_imdb	RoBERTaForSequenceClassification
14	en bert_sequence_classifier_question_statement	BertForSequenceClassification	39	en xlm_roberta_base_sequence_classifier_ag_news	XlmRoBERTaForSequenceClassification
15	en bert_sequence_classifier_question_statement_cl...	BertForSequenceClassification	40	fr xlm_roberta_base_sequence_classifier_allocine	XlmRoBERTaForSequenceClassification
16	en bert_sequence_classifier_antisemitism	BertForSequenceClassification	41	en xlm_roberta_base_sequence_classifier_imdb	XlmRoBERTaForSequenceClassification
17	en bert_sequence_classifier_hatexplain	BertForSequenceClassification	42	en xlnet_base_sequence_classifier_ag_news	XlnetForSequenceClassification
18	en bert_sequence_classifier_trec_coarse	BertForSequenceClassification	43	en xlnet_base_sequence_classifier_imdb	XlnetForSequenceClassification
19	en bert_sequence_classifier_age_news	BertForSequenceClassification			
20	en bert_sequence_classifier_banking77	BertForSequenceClassification			
21	en bert_sequence_classifier_sms_spam	BertForSequenceClassification			
22	en bert_sequence_classifier_song_lyrics	BertForSequenceClassification			
23	en distilbert_base_sequence_classifier_ag_news	DistilBertForSequenceClassification			
24	en distilbert_base_sequence_classifier_amazon_pol...	DistilBertForSequenceClassification			

Transformer Sequence Classifiers in Spark NLP

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Language	Spark NLP Model Name	Model	Language	Spark NLP Model Name	Model
Number			Number		
0	en bert_base_token_classifier_conll03	BertForTokenClassification	26	en distilroberta_base_token_classifier_ontonotes	RoBertaForTokenClassification
1	en bert_base_token_classifier_ontonote	BertForTokenClassification	27	en roberta_base_token_classifier_conll03	RoBertaForTokenClassification
2	en bert_large_token_classifier_conll03	BertForTokenClassification	28	en roberta_base_token_classifier_ontonotes	RoBertaForTokenClassification
3	en bert_large_token_classifier_ontonote	BertForTokenClassification	29	en roberta_large_token_classifier_conll03	RoBertaForTokenClassification
4	fa bert_token_classifier_parsbert_armanner	BertForTokenClassification	30	en roberta_large_token_classifier_ontonotes	RoBertaForTokenClassification
5	fa bert_token_classifier_parsbert_ner	BertForTokenClassification	31	fa roberta_token_classifier_zwnj_base_ner	RoBertaForTokenClassification
6	fa bert_token_classifier_parsbert_peymanner	BertForTokenClassification	32	xx xlm_roberta_token_classifier_ner_40_lang	XlmRoBertaForTokenClassification
7	es bert_token_classifier_spanish_ner	BertForTokenClassification	33	en xlnet_base_token_classifier_conll03	XlnetForTokenClassification
8	sv bert_token_classifier_swedish_ner	BertForTokenClassification	34	en xlnet_large_token_classifier_conll03	XlnetForTokenClassification
9	tr bert_token_classifier_turkish_ner	BertForTokenClassification	35	en bert_token_classifier_ner_aide	MedicalBertForTokenClassifier
10	en distilbert_base_token_classifier_conll03	DistilBertForTokenClassification	36	en bert_token_classifier_ner_anatomy	MedicalBertForTokenClassifier
11	en distilbert_base_token_classifier_ontonotes	DistilBertForTokenClassification	37	en bert_token_classifier_ner_bacteria	MedicalBertForTokenClassifier
12	fa distilbert_token_classifier_persian_ner	DistilBertForTokenClassification	38	en xlm_roberta_base_token_classifier_conll03	XlmRoBertaForTokenClassification
13	en bert_base_token_classifier_few_nerd	BertForTokenClassification	39	en xlm_roberta_base_token_classifier_ontonotes	XlmRoBertaForTokenClassification
14	en distilbert_base_token_classifier_few_nerd	DistilBertForTokenClassification	40	en longformer_base_token_classifier_conll03	LongformerForTokenClassification
15	en bert_token_classifier_ner_clinical	MedicalBertForTokenClassifier	41	en longformer_large_token_classifier_conll03	LongformerForTokenClassification
16	en bert_token_classifier_ner_jsl	MedicalBertForTokenClassifier	42	en bert_token_classifier_ner_chemicals	MedicalBertForTokenClassifier
17	en bert_token_classifier_ner_btc	BertForTokenClassification	43	en bert_token_classifier_ner_chemprot	MedicalBertForTokenClassifier
18	ja bert_token_classifier_ner_ud_gsd	BertForTokenClassification	44	en bert_token_classifier_ner_bionlp	MedicalBertForTokenClassifier
19	en bert_token_classifier_ner_deid	MedicalBertForTokenClassifier	45	en bert_token_classifier_ner_cellular	MedicalBertForTokenClassifier
20	en bert_token_classifier_ner_jsl	MedicalBertForTokenClassifier	46	tr xlm_roberta_base_token_classifier_ner	XlmRoBertaForTokenClassification
21	en bert_token_classifier_ner_drugs	MedicalBertForTokenClassifier	47	id xlm_roberta_large_token_classification_ner	XlmRoBertaForTokenClassification
22	en bert_token_classifier_ner_jsl_slim	MedicalBertForTokenClassifier	48	is roberta_token_classifier_icelandic_ner	RoBertaForTokenClassification
23	en albert_base_token_classifier_conll03	AlbertForTokenClassification	49	xx xlm_roberta_large_token_classifier_masakhaner	XlmRoBertaForTokenClassification
24	en albert_large_token_classifier_conll03	AlbertForTokenClassification	50	zh bert_token_classifier_chinese_ner	BertForTokenClassification

Transformer Token Classifiers in Spark NLP

Part 1

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Language	Spark NLP Model Name	Model
Number		
0	en	google_t5_small_ssm_nq
1	en	google_t5_small_ssm_nq
2	en	bert_base_cased_qa_squad2
3	en	bert_qa_Bertv1_fine
4	en	bert_qa_COVID_BERTa
5	en	bert_qa_COVID_BERTb
6	en	bert_qa_COVID_BERTc
7	de	bert_qa_GBERTQnA
8	en	bert_qa_HomayounSadri_bert_base_uncased_finetuned
9	id	bert_qa_Indobert_QA
10	en	bert_qa_Klue_CommonSense_model
11	en	bert_qa_MTL_bert_base_uncased_ww_squad
12	en	bert_qa_ManuERT_for_xqua
13	en	bert_qa_MiniLM_L12_H384_uncased_finetuned_squad
14	en	bert_qa_Multi_ling_BERT
15	xx	bert_qa_Part_1_mBERT_Model_E1
16	en	bert_qa_Part_1_mBERT_Model_E2
17	en	bert_qa_Part_2_BERT_Multilingual_Dutch_Model_E1
18	en	bert_qa_Part_2_mBERT_Model_E2
19	en	bert_qa_Paul_Vinh_bert_base_multilingual_cased
20	en	bert_qa_PruebaBert
21	en	bert_qa_SciBERT_SQuAD_QuAC
22	en	bert_qa_Seongkyu_bert_base_cased_finetuned_squad
23	en	bert_qa_Shushant_BiomedNLP_PubMedBERT_base_uncased
24	en	bert_qa_Spanbert_emotion_extraction

Transformer Question Answering in Spark NLP Part 1

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Language	Spark NLP Model Name	Model
Number		
25	en bert_qa_SreyanG_NVIDIA_bert_base_cased_finetuned	BertForQuestionAnswering
26	en bert_qa_SreyanG_NVIDIA_bert_base_uncased_finetuned	BertForQuestionAnswering
27	en bert_qa_SupriyaArun_bert_base_uncased_finetuned	BertForQuestionAnswering
28	en bert_qa_Tianle_bert_base_uncased_finetuned_squad	BertForQuestionAnswering
29	en bert_qa_Trial_3_Results	BertForQuestionAnswering
30	ko bert_qa_ainize_klue_bert_base_mrc	BertForQuestionAnswering
31	en bert_qa_andresestevez_bert_base_cased_finetuned	BertForQuestionAnswering
32	en bert_qa_araspeedest	BertForQuestionAnswering
33	ar bert_qa_arap_qa_bert	BertForQuestionAnswering
34	ar bert_qa_arap_qa_bert_large_v2	BertForQuestionAnswering
35	ar bert_qa_arap_qa_bert_v2	BertForQuestionAnswering
36	en bert_qa_augmented_Squad_Translated	BertForQuestionAnswering
37	en bert_qa_augmented	BertForQuestionAnswering
38	en bert_qa_batterybert_cased_squad_v1	BertForQuestionAnswering
39	en bert_qa_batterybert_uncased_squad_v1	BertForQuestionAnswering
40	en bert_qa_batterydata_bert_base_uncased_squad_v1	BertForQuestionAnswering
41	en bert_qa_batteryonlybert_cased_squad_v1	BertForQuestionAnswering
42	en bert_qa_batteryonlybert_uncased_squad_v1	BertForQuestionAnswering
43	en bert_qa_batteryscibert_cased_squad_v1	BertForQuestionAnswering
44	en bert_qa_batteryscibert_uncased_squad_v1	BertForQuestionAnswering
45	en bert_qa_bdickson_bert_base_uncased_finetuned_s	BertForQuestionAnswering
46	en bert_qa_bert_FT_new_newsqa	BertForQuestionAnswering
47	en bert_qa_bert_FT_newsqa	BertForQuestionAnswering
48	en bert_qa_bert_all	BertForQuestionAnswering
49	en bert_qa_bert_all_squad_all_translated	BertForQuestionAnswering

Language	Spark NLP Model Name	Model
Number		
51	en bert_qa_bert_all_squad_que_translated	BertForQuestionAnswering
52	en bert_qa_bert_all_translated	BertForQuestionAnswering
53	en bert_qa_bert_base_1024_full_trivia_copied_embe...	BertForQuestionAnswering
54	en bert_qa_bert_base_2048_full_trivia_copied_embe...	BertForQuestionAnswering
55	en bert_qa_bert_base_4096_full_trivia_copied_embe...	BertForQuestionAnswering
56	en bert_qa_bert_base_512_full_trivia	BertForQuestionAnswering
57	en bert_qa_bert_base_cased_IUChatbot_ontologyDts	BertForQuestionAnswering
58	en bert_qa_bert_base_cased_chaii	BertForQuestionAnswering
59	en bert_qa_bert_base_cased_finetuned_squad_test	BertForQuestionAnswering
60	pt bert_qa_bert_base_cased_squad_v1.1_portuguese	BertForQuestionAnswering
61	en bert_qa_bert_base_cased_squad_v1	BertForQuestionAnswering
62	zh bert_qa_bert_base_chinese_finetuned_squad_colab	BertForQuestionAnswering
63	fa bert_qa_bert_base_fa_qa	BertForQuestionAnswering
64	en bert_qa_bert_base_faquad	BertForQuestionAnswering
65	en bert_qa_bert_base_finetuned_squad2	BertForQuestionAnswering
66	th bert_qa_bert_base_multilingual_cased_finetune_qa	BertForQuestionAnswering
67	en bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
68	nl bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
69	en bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
70	pl bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
71	pl bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
72	en bert_qa_bert_base_multilingual_cased_finetuned...	BertForQuestionAnswering
73	en bert_qa_bert_base_multilingual_cased_korquad	BertForQuestionAnswering
74	en bert_qa_bert_base_multilingual_cased_korquad_v1	BertForQuestionAnswering
75	en bert_qa_bert_base_multilingual_uncased_finetun...	BertForQuestionAnswering

Transformer Question Answering in Spark NLP Part 2

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Language	Spark NLP Model Name	Model
Number		
76	en bert_qa_bert_base_multilingual_xquad	BertForQuestionAnswering
77	si bert_qa_bert_base_sinhala_qa	BertForQuestionAnswering
78	en bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
79	en bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
80	en bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
81	es bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
82	es bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
83	es bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
84	es bert_qa_bert_base_spanish_wwm_cased_finetuned...	BertForQuestionAnswering
85	en bert_qa_bert_base_spanish_wwm_uncased_finetune...	BertForQuestionAnswering
86	en bert_qa_bert_base_spanish_wwm_uncased_finetune...	BertForQuestionAnswering
87	en bert_qa_bert_base_spanish_wwm_uncased_finetune...	BertForQuestionAnswering
88	en bert_qa_bert_base_squadv1	BertForQuestionAnswering
89	en bert_qa_bert_base_swedish_cased_squad_experime...	BertForQuestionAnswering
90	sv bert_qa_bert_base_swedish_squad2	BertForQuestionAnswering
91	en bert_qa_bert_base_turkish_cased_finetuned_lr_2...	BertForQuestionAnswering
92	tr bert_qa_bert_base_turkish_squad	BertForQuestionAnswering
93	en bert_qa_bert_base_uncased_coqa	BertForQuestionAnswering
94	en bert_qa_bert_base_uncased_few_shot_k_1024_fine...	BertForQuestionAnswering
95	en bert_qa_bert_base_uncased_few_shot_k_128_finet...	BertForQuestionAnswering
96	en bert_qa_bert_base_uncased_few_shot_k_16_finetu...	BertForQuestionAnswering
97	en bert_qa_bert_base_uncased_few_shot_k_256_finet...	BertForQuestionAnswering
98	en bert_qa_bert_base_uncased_few_shot_k_32_finetu...	BertForQuestionAnswering
99	en bert_qa_bert_base_uncased_few_shot_k_512_finet...	BertForQuestionAnswering

Number					
100	en	bert_qa_bert_base_uncased_few_shot_k_64_finetu...	BertForQuestionAnswering		
101	en	bert_qa_bert_base_uncased_finetuned_docvqa	BertForQuestionAnswering		
102	en	bert_qa_bert_base_uncased_finetuned_duorc_bert	BertForQuestionAnswering		
103	en	bert_qa_bert_base_uncased_finetuned_infovqa	BertForQuestionAnswering		
104	en	bert_qa_bert_base_uncased_finetuned_newsqa	BertForQuestionAnswering		
105	en	bert_qa_bert_base_uncased_finetuned_squad_froz...	BertForQuestionAnswering		
106	en	bert_qa_bert_base_uncased_finetuned_squad_v1	BertForQuestionAnswering		
107	en	bert_qa_bert_base_uncased_finetuned_squad_v2	BertForQuestionAnswering		
108	en	bert_qa_bert_base_uncased_finetuned_vi_infovqa	BertForQuestionAnswering		
109	en	bert_qa_bert_base_uncased_fiqa_film_sq_filt	BertForQuestionAnswering		
110	en	bert_qa_bert_base_uncased_qa_squad2	BertForQuestionAnswering		
111	en	bert_qa_bert_base_uncased_squad1.1_block_spars...	BertForQuestionAnswering		
112	en	bert_qa_bert_base_uncased_squad1.1_block_spars...	BertForQuestionAnswering		
113	en	bert_qa_bert_base_uncased_squad1.1_block_spars...	BertForQuestionAnswering		
114	en	bert_qa_bert_base_uncased_squad1.1_block_spars...	BertForQuestionAnswering		
115	en	bert_qa_bert_base_uncased_squad1.1_pruned_x3.2_v2	BertForQuestionAnswering		
116	en	bert_qa_bert_base_uncased_squad2_covid_qa_deepset	BertForQuestionAnswering		
117	en	bert_qa_bert_base_uncased_squad_L3	BertForQuestionAnswering		
118	en	bert_qa_bert_base_uncased_squad_L6	BertForQuestionAnswering		
119	en	bert_qa_bert_base_uncased_squad_v1_sparse0.25	BertForQuestionAnswering		
120	en	bert_qa_bert_base_uncased_squadv1.1_sparse_80_...	BertForQuestionAnswering		
121	en	bert_qa_bert_base_uncased_squadv1_x1.16_f88.1_...	BertForQuestionAnswering		
122	en	bert_qa_bert_base_uncased_squadv1_x1.84_f88.7_...	BertForQuestionAnswering		
123	en	bert_qa_bert_base_uncased_squadv1_x1.96_f88.3_...	BertForQuestionAnswering		
124	en	bert_qa_bert_base_uncased_squadv1_x2.01_f89.2_...	BertForQuestionAnswering		
125	en	bert_qa_bert_base_uncased_squadv1_x2.32_f86.6_...	BertForQuestionAnswering		

Transformer Question Answering in Spark NLP Part 3

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number	Language	Spark NLP Model Name	Model
126	en	bert_qa_bert_base_uncased_squadv1_x2.44_f87.7_...	BertForQuestionAnswering
127	zh	bert_qa_bert_chinese_finetuned	BertForQuestionAnswering
128	en	bert_qa_bert	BertForQuestionAnswering
129	en	bert_qa_bert_fa_QA_v1	BertForQuestionAnswering
130	en	bert_qa_bert_large_faquad	BertForQuestionAnswering
131	en	bert_qa_bert_large_uncased_www_squadv2_x2.63_f...	BertForQuestionAnswering
132	en	bert_qa_bert_medium_finetuned_squad	BertForQuestionAnswering
133	en	bert_qa_bert_medium_finetuned_squadv2	BertForQuestionAnswering
134	en	bert_qa_bert_medium_pretrained_finetuned_squad	BertForQuestionAnswering
135	en	bert_qa_bert_medium_squad2_distilled	BertForQuestionAnswering
136	en	bert_qa_bert_medium_wslb_finetuned_squadv1	BertForQuestionAnswering
137	en	bert_qa_bert_mini_5_finetuned_squadv2	BertForQuestionAnswering
138	en	bert_qa_bert_mini_finetuned_squadv2	BertForQuestionAnswering
139	en	bert_qa_bert_mini_wslb_finetuned_squadv1	BertForQuestionAnswering
140	en	bert_qa_bert_multi_cased_finetuned_xquad_chaii	BertForQuestionAnswering
141	en	bert_qa_bert_multi_cased_finetuned_chaii	BertForQuestionAnswering
142	en	bert_qa_bert_multi_cased_finetuned_xquadv1_fin...	BertForQuestionAnswering
143	xx	bert_qa_bert_multi_cased_finetuned_xquadv1	BertForQuestionAnswering
144	de	bert_qa_bert_multi_english_german_squad2	BertForQuestionAnswering
145	en	bert_qa_bert_multi_uncased_finetuned_chaii	BertForQuestionAnswering
146	xx	bert_qa_bert_multi_uncased_finetuned_xquadv1	BertForQuestionAnswering
147	en	bert_qa_bert_qasper	BertForQuestionAnswering
148	en	bert_qa_bert_reader_squad2	BertForQuestionAnswering
149	en	bert_qa_bert_set_date_1_lr_2e_5_bs_32_ep_4	BertForQuestionAnswering

Language	Spark NLP Model Name	Model
Number		
150	en	bert_qa_bert_small_2_finetuned_squadv2
151	en	bert_qa_bert_small_cord19_squad2
152	en	bert_qa_bert_small_cord19qa
153	en	bert_qa_bert_small_finetuned_squad
154	en	bert_qa_bert_small_finetuned_squadv2
155	en	bert_qa_bert_small_pretrained_finetuned_squad
156	en	bert_qa_bert_small_wrsib_finetuned_squadv1
157	en	bert_qa_bert_tiny_2_finetuned_squadv2
158	en	bert_qa_bert_tiny_3_finetuned_squadv2
159	en	bert_qa_bert_tiny_4_finetuned_squadv2
160	en	bert_qa_bert_tiny_5_finetuned_squadv2
161	en	bert_qa_bert_tiny_finetuned_squad
162	en	bert_qa_bert_tiny_finetuned_squadv2
163	tr	bert_qa_bert_turkish_question_answering
164	en	bert_qa_bert_uncased_L_10_H_512_A_8_cord19_200...
165	en	bert_qa_bert_uncased_L_10_H_512_A_8_cord19_200...
166	en	bert_qa_bert_uncased_L_10_H_512_A_8_squad2_cov...
167	en	bert_qa_bert_uncased_L_10_H_512_A_8_squad2
168	en	bert_qa_bert_uncased_L_2_H_512_A_8_cord19_200...
169	en	bert_qa_bert_uncased_L_2_H_512_A_8_squad2_covi...
170	en	bert_qa_bert_uncased_L_2_H_512_A_8_squad2
171	en	bert_qa_bert_uncased_L_4_H_256_A_4_cord19_200...
172	en	bert_qa_bert_uncased_L_4_H_256_A_4_cord19_200...
173	en	bert_qa_bert_uncased_L_4_H_256_A_4_squad2_covi...
174	en	bert_qa_bert_uncased_L_4_H_256_A_4_squad2

Transformer Question Answering in Spark NLP Part 4

Search them all in the Modelshub
<https://nlp.johnsnowlabs.com/models>

Number			
175	en	bert_qa_bert_uncased_L_4_H_512_A_8_cord19_200...	BertForQuestionAnswering
176	en	bert_qa_bert_uncased_L_4_H_512_A_8_cord19_200...	BertForQuestionAnswering
177	en	bert_qa_bert_uncased_L_4_H_512_A_8_squad2_covi...	BertForQuestionAnswering
178	en	bert_qa_bert_uncased_L_4_H_512_A_8_squad2	BertForQuestionAnswering
179	en	bert_qa_bert_uncased_L_4_H_768_A_12_cord19_200...	BertForQuestionAnswering
180	en	bert_qa_bert_uncased_L_4_H_768_A_12_cord19_200...	BertForQuestionAnswering
181	en	bert_qa_bert_uncased_L_4_H_768_A_12_squad2_covi...	BertForQuestionAnswering
182	en	bert_qa_bert_uncased_L_4_H_768_A_12_squad2	BertForQuestionAnswering
183	en	bert_qa_bert_uncased_L_6_H_128_A_2_squad2_covi...	BertForQuestionAnswering
184	en	bert_qa_bert_uncased_L_6_H_128_A_2_squad2	BertForQuestionAnswering
185	en	bert_qa_bertimbau_squad1.1	BertForQuestionAnswering
186	en	bert_qa_bertserini_bert_base_squad	BertForQuestionAnswering
187	en	bert_qa_bertserini_bert_large_squad	BertForQuestionAnswering
188	ko	bert_qa_bespin_global_klue_bert_base_mrc	BertForQuestionAnswering
189	es	bert_qa_beto_base_spanish_sqac	BertForQuestionAnswering
190	pt	bert_qa_bioBERTpt_squad_v1.1_portuguese	BertForQuestionAnswering
191	en	bert_qa_biobert_base_cased_v1.1_squad	BertForQuestionAnswering
192	en	bert_qa_biobert_base_cased_v1.1_squad_finetune...	BertForQuestionAnswering
193	en	bert_qa_biobert_base_cased_v1.1_squad_finetune...	BertForQuestionAnswering
194	en	bert_qa_biobert_base_cased_v1.1_squad_finetune...	BertForQuestionAnswering
195	en	bert_qa_biobert_bioasq	BertForQuestionAnswering
196	en	bert_qa_biobert_squad2_cased	BertForQuestionAnswering
197	en	bert_qa_biobert_squad2_cased_finetuned_squad	BertForQuestionAnswering
198	en	bert_qa_biobert_v1.1_pubmed_finetuned_squad	BertForQuestionAnswering
199	en	bert_qa_biobert_v1.1_pubmed_squad_v2	BertForQuestionAnswering
200	en	bert_qa_bioformer_cased_v1.0_squad1	BertForQuestionAnswering

Transformers & Embeddings

BERT is a bi-directional transformer for pre-training over a lot of unlabeled textual data to learn a language representation that can be used to fine-tune for specific machine learning tasks. While BERT outperformed the NLP state-of-the-art on several challenging tasks, its performance improvement could be attributed to the bidirectional transformer, novel pre-training tasks of Masked Language Model and Next Structure Prediction along with a lot of data and Google's compute power. It is an Auto Encoder Language Model.

XLNet is a large bidirectional transformer that uses improved training methodology, larger data and more computational power to achieve better than BERT prediction metrics on 20 language tasks. To improve the training, XLNet introduces permutation language modeling, where all tokens are predicted but in random order. This is in contrast to BERT's masked language model where only the masked (15%) tokens are predicted. It is an Autoregressive Language Model.

Albert is Google's new "ALBERT" language model and achieved state-of-the-art results on three popular benchmark tests for natural language understanding (NLU): GLUE, RACE, and SQuAD 2.0. ALBERT is a "lite" version of Google's 2018 NLU pre training method BERT. Researchers introduced two parameter-reduction techniques in ALBERT to lower memory consumption and increase training speed and the Next Sentence Prediction task is replace by Sentence Order Prediction

USE (Universal Sentence Encoder) is a Transformer-based model for encoding sentences into embedding vectors that specifically target transfer learning to other NLP tasks and outperforms previous word-embedding models on various NLP tasks

Transformers & Embeddings

DistilBert is a compressed version of BERT, which leverages knowledge distillation during the pre-training phase and shows that it is possible to reduce the size of a BERT model by 40%, while retaining 97% of its language understanding capabilities and being 60% faster. It introduces a triple loss combining language modeling, distillation and cosine-distance losses. The smaller, faster and lighter model is cheaper to pre-train and we demonstrate its capabilities for on-device computations in a proof-of-concept experiment and a comparative on-device

RoBerta is a optimized version of BERT, which has improved hyperparameters and training data size. The findings show that the original BERT was significantly undertrained and with (1) Longer training, bigger batches, (2) removing next sentence prediction objective, (3) training on longer sequences and (4) dynamically changing the masking pattern applied to the training data every previous BERT can be outperformed and new state of the art can be achieved

XlmRoBerta is a multilingual BERT model which significantly outperforms multilingual BERT on a variety of cross-lingual benchmark and large accuracy gains on various multi-lingual benchmarks for 88 languages that appear in the Wiki-100 corpus

Longformer is a Transformer-based model which improves on processing long sequences by introduces a windowed attention mechanism and a linearly scaling self-attention mechanism which scales linearly with sequence length and easily processes documents with thousands or more tokens.

NLP Transformers & Embeddings

- BERT : <https://arxiv.org/abs/1810.04805>
- XLNet : <https://arxiv.org/abs/1906.08237>
- Albert : <https://arxiv.org/abs/1909.11942>
- Elmo : <https://arxiv.org/abs/1802.05365>
- USE : <https://arxiv.org/abs/1803.11175>
- DistilBert : <https://arxiv.org/abs/1910.01108>
- RoBerta : <https://arxiv.org/abs/1907.11692>
- DeBerta : <https://arxiv.org/abs/2006.03654>
- XlmRoberta : <https://arxiv.org/abs/1911.02116>
- Longformer : <https://arxiv.org/abs/2004.05150>
- Electra : <https://arxiv.org/abs/2003.10555>
- T5 : <https://arxiv.org/abs/1910.10683>
- Marian: <https://arxiv.org/abs/1804.00344>
- GPT2: <https://openai.com/blog/better-language-models/>
- SpanBERT <https://arxiv.org/abs/1907.10529>
- CamemBERT <https://camembert-model.fr/>
- SciBert <https://www.aclweb.org/anthology/D19-1371/>
- MiniLm <https://arxiv.org/abs/2002.10957>
- CovidBERT <https://arxiv.org/abs/2005.07503>
- BioBERT <https://arxiv.org/abs/1901.08746>
- indoBERT <https://arxiv.org/abs/2011.00677>
- MuRIL <https://arxiv.org/abs/2103.10730>
- sapBERT <https://github.com/cambridgeeltl/sapbert>
- BioFormer <https://github.com/WGLab/Bioformer>
- LinkBERT <https://arxiv.org/abs/2203.15827>
- MacBERT <https://aclanthology.org/2020.findings-emnlp.58>
- DOC2Vec : <https://arxiv.org/abs/1301.3781>
- Word2Vec: <https://arxiv.org/abs/1301.3781>

100+ Languages supported by Language-agnostic BERT Sentence Embedding (LABSE) and XLM-RoBERTa

Train in 1 Language, predict in 100+ different languages



```
# Binary Class Classifier, 2 classes
nlu.load('xx.embed_sentence.labse train.sentiment').fit(train_df).predict(test_df)

# Multi Class Classifier, N classes
nlu.load('xx.embed_sentence.labse train.classifier').fit(train_df).predict(test_df)

# Multi Class Classifier with multiple labels example (i.e. Hashtags)
# N classes, where one row can be assigned up to N labels
nlu.load('xx.embed_sentence.labse train.multi_classifier').fit(train_df).predict(test_df)
```

ISO	NAME	ISO	NAME	ISO	NAME
af	AFRIKAANS	ht	HAITIAN_CREOLE	pt	PORTRUGUESE
am	AMHARIC	hu	HUNGARIAN	ro	ROMANIAN
ar	ARABIC	hy	ARMENIAN	ru	RUSSIAN
as	ASSAMESE	id	INDONESIAN	rw	KINYARWANDA
az	AZERBAIJANI	ig	IGBO	si	SINHALA/ESL
be	BELARUSIAN	is	ICELANDIC	sk	SLOVAK
bg	BULGARIAN	it	ITALIAN	sm	SAMOAN
bn	BENGALI	ja	JAPANESE	sn	SHONA
bo	TIBETAN	JV	JAVANESE	so	SOMALI
bs	BOSNIAN	ka	GEORGIAN	sq	ALBANIAN
ca	CATALAN	kk	KAZAKH	sr	SERBIAN
ceb	CEBUANO	km	KHMER	st	SESOTHO
co	CORSICAN	kn	KANNADA	su	SUNDANESE
cs	CZECH	ko	KOREAN	sv	SWEDISH
cy	WELSH	ku	KURDISH	sw	SWAHILI
da	DANISH	ky	KYRGYZ	ta	TAMIL
de	GERMAN	la	LATIN	te	TELUGU
el	GREEK	lb	LUXEMBOURGISH	tg	TAJIK
en	ENGLISH	lo	LAOTHIAN	th	THAI
eo	ESPERANTO	lt	LITHUANIAN	tk	TURKMEN
es	SPANISH	lv	LATVIAN	tl	TAGALOG
et	ESTONIAN	mg	MALAGASY	tr	TURKISH
eu	BASQUE	mi	MAORI	tt	TATAR
fa	PERSIAN	mk	MACEDONIAN	ug	UIGHUR
fi	FINNISH	ml	MALAYALAM	uk	UKRAINIAN
fr	FRENCH	mn	MONGOLIAN	ur	URDU
fy	FRISIAN	mr	MARATHI	uz	UZBEK
ga	IRISH	ms	MALAY	vi	VietNAMESE
gd	SCOTS_GAELIC	mt	MALTESE	wo	WOLOF
gl	Galician	my	BURMESE	xh	XHOSA
gu	GUARATI	ne	NEPALI	yi	YIDDISH
ha	HAUSA	nl	DUTCH	yo	YORUBA
haw	HAWAIIAN	no	NORWEGIAN	zh	Chinese
he	HEBREW	ny	NYANJA	zu	ZULU
hi	HINDI	or	ORIYA		
hmn	HMONG	pa	PUNABI		
hr	CROATIAN	pl	POLISH		

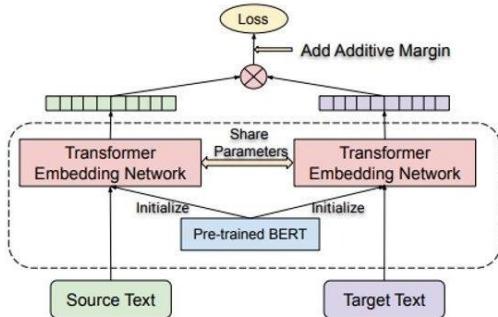


Figure 1: Dual encoder model with BERT based encoding modules.



Translate between 200+ Languages With Marian: Fast Neural Machine Translation in C++



MARIAN NMT

Fast Neural Machine Translation in C++



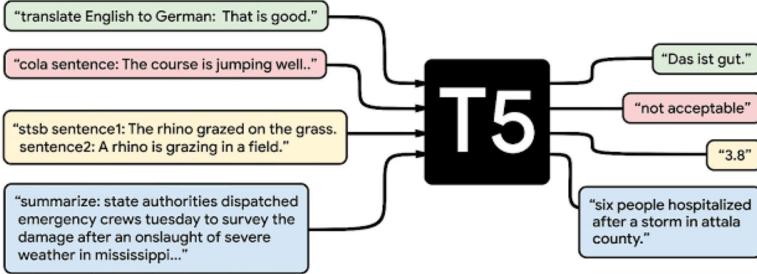
```
# Use ISO standards for the languages
nlu.load('<start_language>.translate_to.<target_language>')

#Translate Turkish to English:
nlu.load('tr.translate_to.en')

#Translate English to French:
nlu.load('en.translate_to.fr')

#Translate French to Hebrew
nlu.load('fr.translate_to.he')

#Translate English to German
nlu.load('en.translate_to.de')
```



```

# Closed book Question Answering
nlu.load('en.t5').predict('what is the capital of Germany?') # >>> Berlin
# Open Book Question answering
nlu.load('en.t5').predict('Who is president of Nigeria?') # >>> Muhammadu Buhari

# Open book Question Answering
context = 'Peters last week was terrible! He had an accident and broke his leg while skiing!'
question1 = 'Why was peters week so bad?'
question2 = 'How did peter broke his leg?'
nlu.load('answer_question').predict(question1 + context) # >>> broke his leg
nlu.load('answer_question').predict(question2 + context) # >>> skiing

# Big T5 model for Summarization, Sentiment, Text Similarity and other SQuAD/GLUE tasks
pipe = nlu.load('t5')
pipe['t5'].settask('summarize')
pipe.predict(long_text)
  
```

Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer

1. Text summarization
2. Question answering
3. Translation
4. Sentiment analysis
5. Natural Language inference
6. Coreference resolution
7. Sentence Completion
8. Word sense disambiguation



Every T5 Task with explanation:

Task Name	Explanation
1.CoLA	Classify if a sentence is grammatically correct
2.RTE	Classify whether a statement can be deduced from a sentence
3.MNLI	Classify for a hypothesis and premise whether they contradict or contradict each other or neither of both (3 class).
4.MRPC	Classify whether a pair of sentences is a re-phrasing of each other (semantically equivalent)
5.QNLI	Classify whether the answer to a question can be deducted from an answer candidate.
6.QQP	Classify whether a pair of questions is a re-phrasing of each other (semantically equivalent)
7.SST2	Classify the sentiment of a sentence as positive or negative
8.STSB	Classify the sentiment of a sentence on a scale from 1 to 5 (21 Sentiment classes)
9.CB	Classify for a premise and a hypothesis whether they contradict each other or not (binary).
10.COPA	Classify for a question, premise, and 2 choices which choice the correct choice is (binary).
11.MultiRc	Classify for a question, a paragraph of text, and an answer candidate, if the answer is correct (binary).
12.WIC	Classify for a pair of sentences and a disambiguous word if the word has the same meaning in both sentences.
13.WSC/DPR	Predict for an ambiguous pronoun in a sentence what it is referring to.
14.Summarization	Summarize text into a shorter representation.
15.SQuAD	Answer a question for a given context.
16.WMT1	Translate English to German
17.WMT2	Translate English to French
18.WMT3	Translate English to Romanian

Train Transformer Models via Hugging Face or TfHub and scale with Spark NLP



TF Hub to Spark NLP

Spark NLP	TF Hub Notebooks	Colab
BertEmbeddings	TF Hub in Spark NLP - BERT	Open in Colab
BertSentenceEmbeddings	TF Hub in Spark NLP - BERT Sentence	Open in Colab
AlbertEmbeddings	TF Hub in Spark NLP - ALBERT	Open in Colab

Spark NLP	HuggingFace Notebooks	Colab
BertEmbeddings	HuggingFace in Spark NLP - BERT	Open in Colab
BertSentenceEmbeddings	HuggingFace in Spark NLP - BERT Sentence	Open in Colab
DistilBertEmbeddings	HuggingFace in Spark NLP - DistilBERT	Open in Colab
CamemBERTEmbeddings	HuggingFace in Spark NLP - CamemBERT	Open in Colab
RoBERTaEmbeddings	HuggingFace in Spark NLP - RoBERTa	Open in Colab
DeBERTaEmbeddings	HuggingFace in Spark NLP - DeBERTa	Open in Colab
XlmRoBERTaEmbeddings	HuggingFace in Spark NLP - XLM-RoBERTa	Open in Colab
AlbertEmbeddings	HuggingFace in Spark NLP - ALBERT	Open in Colab
XlnetEmbeddings	HuggingFace in Spark NLP - XLNet	Open in Colab
LongformerEmbeddings	HuggingFace in Spark NLP - Longformer	Open in Colab
BertForTokenClassification	HuggingFace in Spark NLP - BertForTokenClassification	Open in Colab
DistilBertForTokenClassification	HuggingFace in Spark NLP - DistilBertForTokenClassification	Open in Colab
AlbertForTokenClassification	HuggingFace in Spark NLP - AlbertForTokenClassification	Open in Colab
RoBERTaForTokenClassification	HuggingFace in Spark NLP - RoBERTaForTokenClassification	Open in Colab
XlmRoBERTaForTokenClassification	HuggingFace in Spark NLP - XlmRoBERTaForTokenClassification	Open in Colab
BertForSequenceClassification	HuggingFace in Spark NLP - BertForSequenceClassification	Open in Colab
DistilBertForSequenceClassification	HuggingFace in Spark NLP - DistilBertForSequenceClassification	Open in Colab
AlbertForSequenceClassification	HuggingFace in Spark NLP - AlbertForSequenceClassification	Open in Colab
RoBERTaForSequenceClassification	HuggingFace in Spark NLP - RoBERTaForSequenceClassification	Open in Colab
XlmRoBERTaForSequenceClassification	HuggingFace in Spark NLP - XlmRoBERTaForSequenceClassification	Open in Colab
XlnetForSequenceClassification	HuggingFace in Spark NLP - XlnetForSequenceClassification	Open in Colab
LongformerForSequenceClassification	HuggingFace in Spark NLP - LongformerForSequenceClassification	Open in Colab
AlbertForQuestionAnswering	HuggingFace in Spark NLP - AlbertForQuestionAnswering	Open in Colab
BertForQuestionAnswering	HuggingFace in Spark NLP - BertForQuestionAnswering	Open in Colab
DeBERTaForQuestionAnswering	HuggingFace in Spark NLP - DeBERTaForQuestionAnswering	Open in Colab
DistilBertForQuestionAnswering	HuggingFace in Spark NLP - DistilBertForQuestionAnswering	Open in Colab
LongformerForQuestionAnswering	HuggingFace in Spark NLP - LongformerForQuestionAnswering	Open in Colab
RoBERTaForQuestionAnswering	HuggingFace in Spark NLP - RoBERTaForQuestionAnswering	Open in Colab
XlmRobertaForQuestionAnswering	HuggingFace in Spark NLP - XlmRobertaForQuestionAnswering	Open in Colab



Session 3 (Day 1) - Coding Time

- ❖ [Notebook 3 Spark NLP pretrained models](#)

Spark NLP
for Data Scientists



Session 4 (Day 1)

- ❖ (cont.) Usage and overview of the 10000+ pretrained models for 300+ languages
- ❖ Training Named Entity Recognition (NER) models

**Spark NLP
for Data Scientists**



Session 4 (Day 1) - Coding Time

- ❖ Notebook 3 Spark NLP pretrained models

Spark NLP
for Data Scientists



CoNLL 2003 (English)

The CoNLL 2003 NER task consists of newswire text from the Reuters RCV1 corpus tagged with four different entity types (PER, LOC, ORG, MISC). Models are evaluated based on span-based F1 on the test set. * used both the train and development splits for training.

Model	F1	Paper / Source	Code
CNN Large + fine-tune (Baevski et al., 2019)	93.5	Cloze-driven Pretraining of Self-attention Networks	
RNN-CRF+Flair	93.47	Improved Differentiable Architecture Search for Language Modeling and Named Entity Recognition	
LSTM-CRF+ELMo+BERT+Flair	93.38	Neural Architectures for Nested NER through Linearization	Official
Flair embeddings (Akbik et al., 2018)*	93.09	Contextual String Embeddings for Sequence Labeling	Flair framework
BERT Large (Devlin et al., 2018)	92.8	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	
CVT + Multi-Task (Clark et al., 2018)	92.61	Semi-Supervised Sequence Modeling with Cross-View Training	Official
BERT Base (Devlin et al., 2018)	92.4	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	
BILSTM-CRF+ELMo (Peters et al., 2018)	92.22	Deep contextualized word representations	AllenNLP Project AllenNLP GitHub
Peters et al. (2017) *	91.93	Semi-supervised sequence tagging with bidirectional language models	
CRF + AutoEncoder (Wu et al., 2018)	91.87	Evaluating the Utility of Hand-crafted Features in Sequence Labelling	Official
Bi-LSTM-CRF + Lexical Features (Ghadhar and Langlais 2018)	91.73	Robust Lexical Features for Improved Neural Network Named-Entity Recognition	Official
BILSTM-CRF + IntNet (Xin et al., 2018)	91.64	Learning Better Internal Structure of Words for Sequence Labeling	
Chiu and Nichols (2016) *	91.62	Named entity recognition with bidirectional LSTM-CNNs	

NER-DL in Spark NLP

SYSTEM	YEAR	LANGUAGE	ACCURACY
Spark NLP v2.4	2020	Python/Scala/Java/R	93.3 (test F1) - 95.9 (dev F1)
Spark NLP v2.x	2019	Python/Scala/Java/R	93
Spark NLP v1.x	2018	Python/Scala/Java/R	92
spaCy v2.x	2017	Python/Cython	92.6
spaCy v1.x	2015	Python/Cython	91.8
ClearNLP	2015	Java	91.7
CoreNLP	2015	Java	89.6
MATE	2015	Java	92.5
Turbo	2015	C++	92.4

The best NER score in production

93.3 %
Test Set

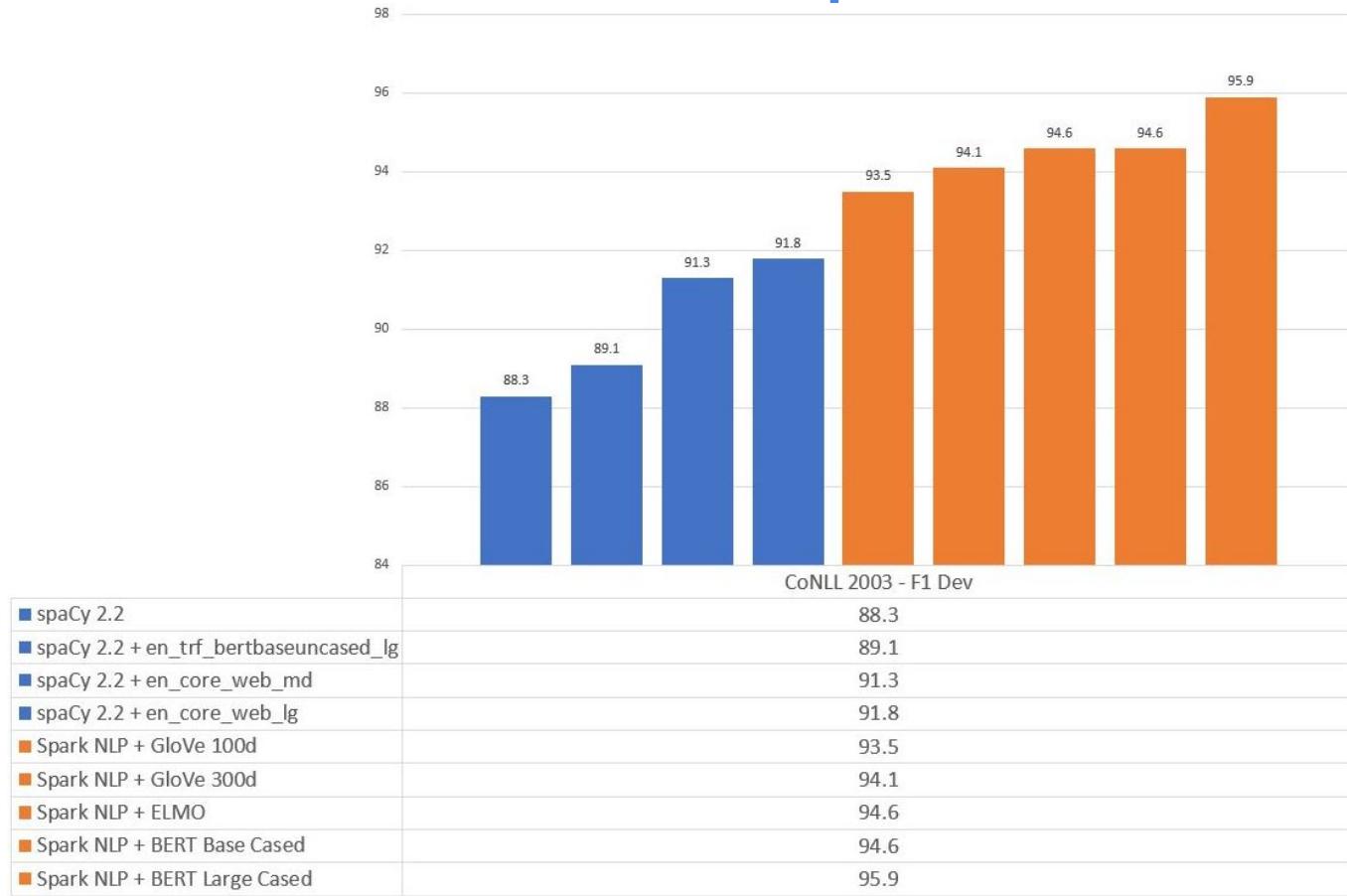


Bert



NerDLApproach

NER-DL in Spark NLP



NER Systems

Feature-engineered machine learning systems	Dict	SP	DU	EN	GE
Carreras et al. (2002) binary AdaBoost classifiers	Yes	81.39	77.05	-	-
Malouf (2002) - Maximum Entropy (ME) + features	Yes	73.66	68.08	-	-
Li et al. (2005) SVM with class weights	Yes	-	-	88.3	-
Passos et al. (2014) CRF	Yes	-	-	90.90	-
Ando and Zhang (2005a) Semi-supervised state of the art	No	-	-	89.31	75.27
Agerri and Rigau (2016)	Yes	84.16	85.04	91.36	76.42
Feature-inferring neural network word models					
Collobert et al. (2011) Vanilla NN +SLL / Conv-CRF	No	-	-	81.47	-
Huang et al. (2015) Bi-LSTM+CRF	No	-	-	84.26	-
Yan et al. (2016) Win-BiLSTM (English), FF (German) (Many fets)	Yes	-	-	88.91	76.12
Collobert et al. (2011) Conv-CRF (SENNNA+Gazetteer)	Yes	-	-	89.59	-
Huang et al. (2015) Bi-LSTM+CRF+ (SENNNA+Gazetteer)	Yes	-	-	90.10	-
Feature-inferring neural network character models					
Gillick et al. (2015) – BTS	No	82.95	82.84	86.50	76.22
Kuru et al. (2016) CharNER	No	82.18	79.36	84.52	70.12
Feature-inferring neural network word + character models					
Yang et al. (2017)	Yes	85.77	85.19	91.26	-
Luo (2015)	Yes	-	-	91.20	-
Chiu and Nichols (2015)	Yes	-	-	91.62	-
Ma and Hovy (2016)	No	-	-	91.21	-
Santos and Guimaraes (2015)	No	82.21	-	-	-
Lample et al. (2016)	No	85.75	81.74	90.94	78.76
Bharadwaj et al. (2016)	Yes	85.81	-	-	-
Dernoncourt et al. (2017)	No	-	-	90.5	-
Feature-inferring neural network word + character + affix models					
Re-implementation of Lample et al. (2016) (100 Epochs)	No	85.34	85.27	90.24	78.44
Yadav et al. (2018)(100 Epochs)	No	86.92	87.50	90.69	78.56
Yadav et al. (2018) (150 Epochs)	No	87.26	87.54	90.86	79.01

1. Classical Approaches (rule based)

2. ML Approaches

- Multi-class classification

- Conditional Random Field (CRF)

3. DL Approaches

- [Bidirectional LSTM-CRF](#)

- [Bidirectional LSTM-CNNs](#)

- [Bidirectional LSTM-CNNs-CRF](#)

- Pre-trained language models (Bert, Elmo)

4. Hybrid Approaches (DL + ML)

NER-DL in Spark NLP

Char-CNN-BiLSTM

	F1 : Tokens	F2 : Casing	F3 : POS	F4 : Char CNN	Labels
The					O
company					O
XYZ					Company
Private					Company
Limited					Company
works					O
in					O
the					O
health					Activity
sector					Activity
in					O
Europe					Location

NER-DL in Spark NLP

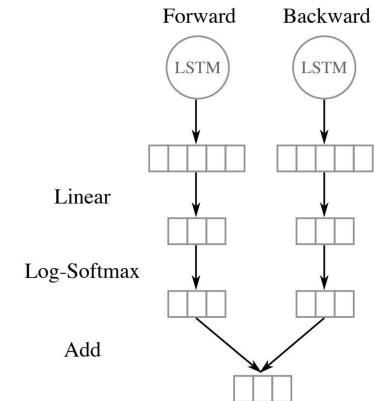
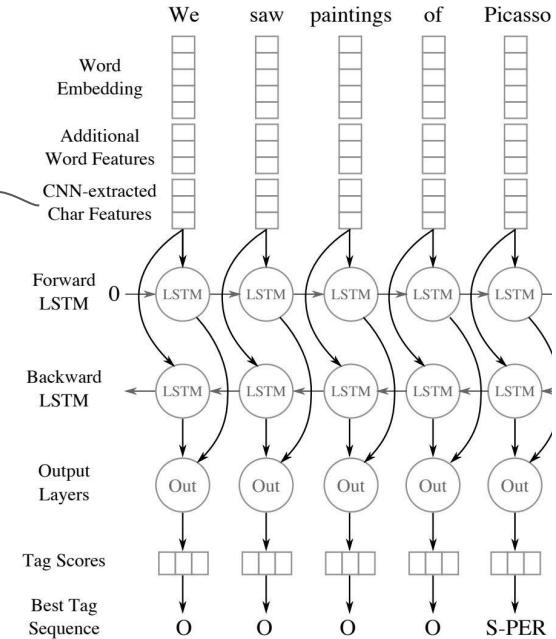
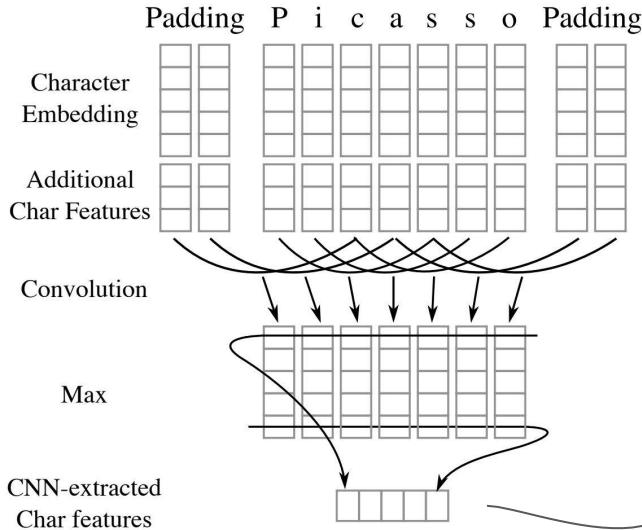


Figure 3: The output layers (“Out” in Figure 1) decode output into a score for each tag category.

Char-CNN-BiLSTM

NER-DL in Spark NLP

CoNLL2003 format

All data files contain one word per line with empty lines representing sentence boundaries. At the end of each line there is a tag which states whether the current word is inside a named entity or not. The tag also encodes the type of named entity. Here is an example sentence:

U.N.	NNP	I-NP	I-ORG
official	NN	I-NP	O
Ekeus	NNP	I-NP	I-PER
heads	VBZ	I-VP	O
for	IN	I-PP	O
Baghdad	NNP	I-NP	I-LOC
.	.	O	O

*Each line contains four fields: the word, its part-of-speech tag, its chunk tag and its named entity tag.

* CoNLL: Conference on Computational Natural Language Learning

BIO schema

John	B-PER
Smith	I-PER
lives	0
in	0
New	B-LOC
York	I-LOC

John Smith ⇒ PERSON
New York ⇒ LOCATION

Session 4 (Day 1) - Coding Time

- ❖ [Notebook 4: NERDL Training](#)

Spark NLP
for Data Scientists



Coding ...

Open 4. NERDL Training notebook in Colab

(click on Colab icon or open in a new tab)

https://github.com/JohnSnowLabs/spark-nlp-workshop/blob/master/open-source-nlp/04.0.NERDL_Training.ipynb

Test set evaluation

```
In [ ]: import pyspark.sql.functions as F

predictions.select(F.explode(F.arrays_zip('token.result','label.result','ner.result')).alias("cols")) \
    .select(F.expr("cols['0']").alias("token"),
           F.expr("cols['1']").alias("ground_truth"),
           F.expr("cols['2']").alias("prediction")).show(truncate=False)

+-----+-----+-----+
|token |ground_truth|prediction|
+-----+-----+-----+
|CRICKET|O          |O          |
|-      |O          |O          |
|LEICESTERSHIRE|B-ORG|B-ORG
|TAKE   |O          |O          |
|OVER   |O          |O          |
|AT     |O          |O          |
|TOP    |O          |O          |
|AFTER  |O          |O          |
|INNINGS|O          |O          |
|VICTORY|O          |O          |
|.     |O          |O          |
|LONDON |B-LOC     |B-LOC
|1996-08-30|O          |O          |
|West   |B-MISC    |B-MISC
|Indian |I-MISC    |I-MISC
|all-roundner|O          |O          |
|Phil   |B-PER     |B-PER
|Simmons|I-PER     |I-PER
|took   |O          |O          |
|four   |O          |O          |
+-----+-----+-----+
only showing top 20 rows
```

```
In [ ]: from sklearn.metrics import classification_report

preds_df = predictions.select(F.explode(F.arrays_zip('token.result','label.result','ner.result')).alias("cols")) \
    .select(F.expr("cols['0']").alias("token"),
           F.expr("cols['1']").alias("ground_truth"),
           F.expr("cols['2']").alias("prediction")).toPandas()

print (classification_report(preds_df['ground_truth'], preds_df['prediction']))

precision    recall   f1-score   support
B-LOC       0.88      0.93      0.90      1837
B-MISC      0.80      0.82      0.81       922
B-ORG       0.92      0.73      0.81      1341
B-PER       0.94      0.95      0.95      1842
```

End of Day 1 - see you tomorrow!
Same time, same place :)

Spark NLP
for Data Scientists



Welcome to Day-2 - We have a lot of things ahead of us

Day-2	50 min	<ul style="list-style-type: none">- Train Text Classifiers- Upload models to the Models Hub
	10 min	Break
	50 min	<ul style="list-style-type: none">- Spell Checking- Keyword Extraction with YAKE- Rule-based Entity Recognition with EntityRuler- Graph triplet extraction
	10 min	Break
	50 min	<ul style="list-style-type: none">- Token Classification with Transformers- Sequence Classification with Transformers- Image Classification with ViT- Speech to Text with Wav2Vec2
	10 min	Break
	50 min	<ul style="list-style-type: none">- Table Question Answering with TAPAS- Question Answering, Summarization and other T5 applications- Multilingual NLP - Train only on English data and predict for 100+ languages

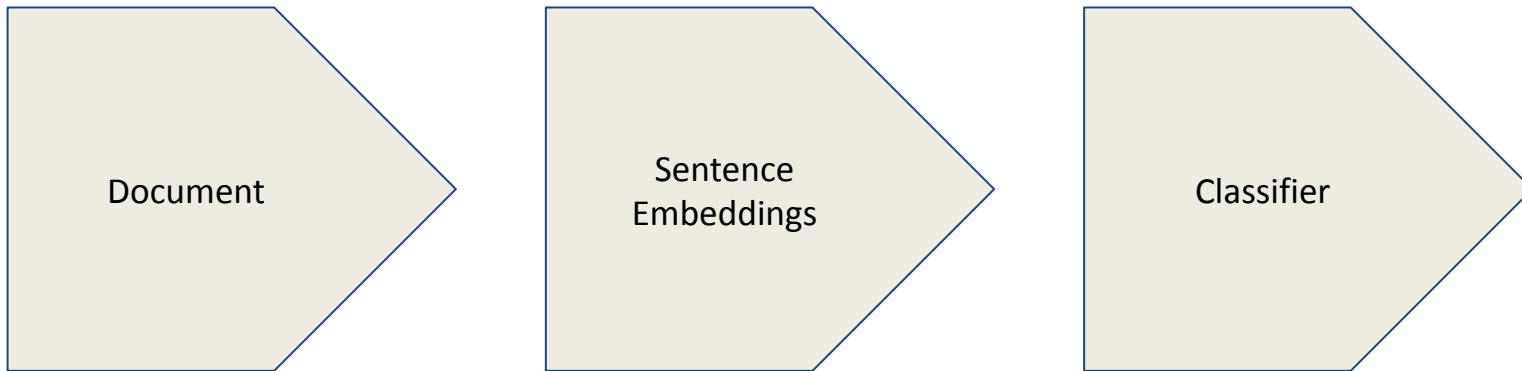
Session 1 (Day 2)

- ❖ Training a Deep Learning Text Classifier
- ❖ Upload/Download your own model via John Snow Labs Models Hub

**Spark NLP
for Data Scientists**



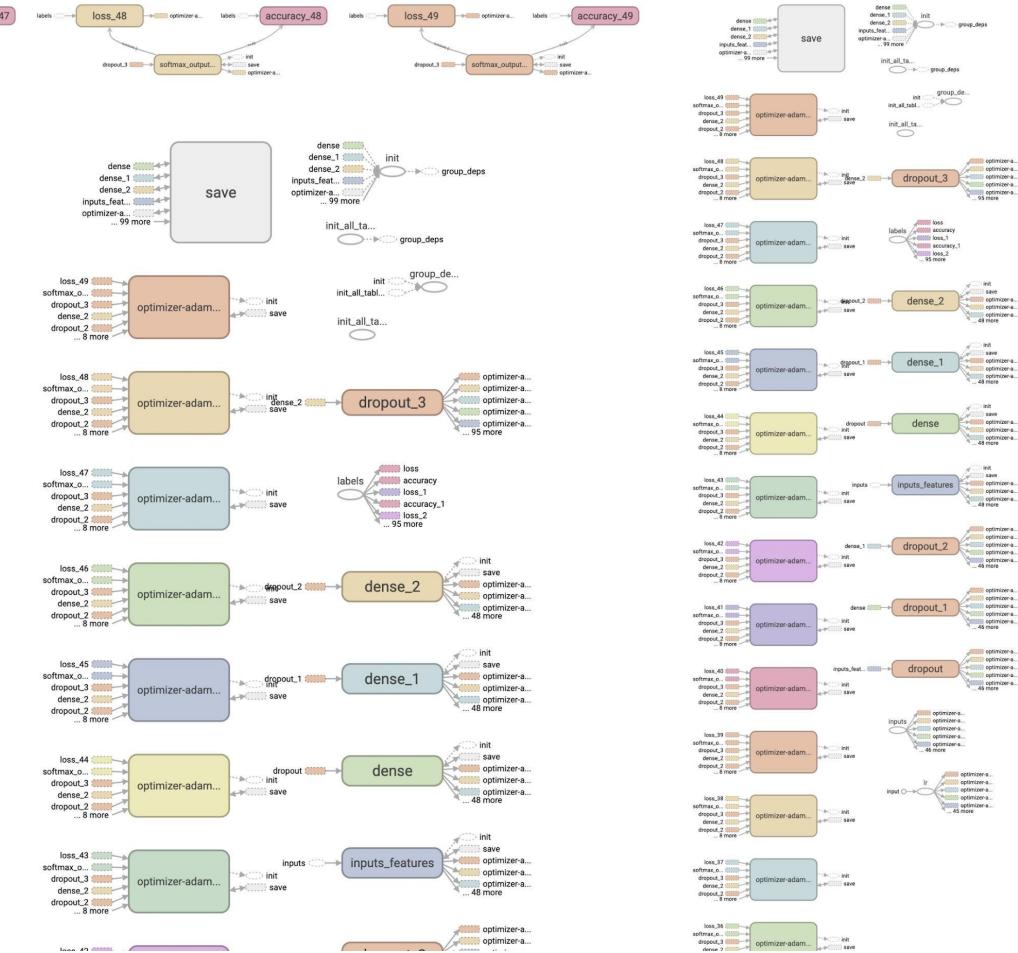
Spark NLP Classifiers



SentimentDL, ClassifierDL, and MultiClassifierDL

- BERT
 - 2 classes (positive/negative)
 - 100 dimensions
- Small BERT
 - 3 classes (0, 1, 2)
 - 200 dimensions
- BioBERT
 - 4 classes (Sports, Business, etc.)
 - 128 dimensions
- CovidBERT
 - 5 classes (1.0, 2.0, 3.0, 4.0, 5.0)
 - 256 dimensions
- LaBSE
 - ... 100 classes!
 - 300 dimensions
- ALBERT
 - 512 dimensions
- ELECTRA
 - 768 dimensions
- XLNet
 - 1024 dimensions
- ELMO
- Universal Sentence Encoder
- GloVe
- Mpnet
- E5
 - tfhub_ues
 - tfhub_use_lg
 - glove_6B_100
 - glove_6B_300
 - glove_840B_300
 - bert_base_cased
 - bert_base_uncased
 - bert_large_cased
 - bert_large_uncased
 - bert_multi_uncased
 - electra_small_uncased
 - elmo
 - ... 100+ Word & Sentence models

Classifier DL Tensorflow Architecture



Import a model from HuggingFace 😊 into Spark NLP

Example Notebooks		
HuggingFace to Spark NLP		
Spark NLP	HuggingFace Notebooks	Colab
BertEmbeddings	HuggingFace in Spark NLP - BERT	Open in Colab
BertSentenceEmbeddings	HuggingFace in Spark NLP - BERT Sentence	Open in Colab
DistilBertEmbeddings	HuggingFace in Spark NLP - DistilBERT	Open in Colab
CamemBertEmbeddings	HuggingFace in Spark NLP - CamemBERT	Open in Colab
RoBERTaEmbeddings	HuggingFace in Spark NLP - RoBERTa	Open in Colab
DeBertaEmbeddings	HuggingFace in Spark NLP - DeBERTa	Open in Colab
XlmRoBERTaEmbeddings	HuggingFace in Spark NLP - XLM-RoBERTa	Open in Colab
AlbertEmbeddings	HuggingFace in Spark NLP - ALBERT	Open in Colab
XinerEmbeddings	HuggingFace in Spark NLP - XLNet	Open in Colab
LongformerEmbeddings	HuggingFace in Spark NLP - Longformer	Open in Colab
BertForTokenClassification	HuggingFace in Spark NLP - BertForTokenClassification	Open in Colab
DistilBertForTokenClassification	HuggingFace in Spark NLP - DistilBertForTokenClassification	Open in Colab
AlbertForTokenClassification	HuggingFace in Spark NLP - AlbertForTokenClassification	Open in Colab
RoBERTaForTokenClassification	HuggingFace in Spark NLP - RobertaForTokenClassification	Open in Colab
BertForSequenceClassification	HuggingFace in Spark NLP - BertForSequenceClassification	Open in Colab
DistilBertForSequenceClassification	HuggingFace in Spark NLP - DistilBertForSequenceClassification	Open in Colab
AlbertForSequenceClassification	HuggingFace in Spark NLP - AlbertForSequenceClassification	Open in Colab
RoBERTaForSequenceClassification	HuggingFace in Spark NLP - RoBERTaForSequenceClassification	Open in Colab
XlmRoBERTaForSequenceClassification	HuggingFace in Spark NLP - XlmRoBERTaForSequenceClassification	Open in Colab
XinerForSequenceClassification	HuggingFace in Spark NLP - XinerForSequenceClassification	Open in Colab
LongformerForSequenceClassification	HuggingFace in Spark NLP - LongformerForSequenceClassification	Open in Colab
AlbertForQuestionAnswering	HuggingFace in Spark NLP - AlbertForQuestionAnswering	Open in Colab
BertForQuestionAnswering	HuggingFace in Spark NLP - BertForQuestionAnswering	Open in Colab
DeBertaForQuestionAnswering	HuggingFace in Spark NLP - DeBertaForQuestionAnswering	Open in Colab
DistilBertForQuestionAnswering	HuggingFace in Spark NLP - DistilBertForQuestionAnswering	Open in Colab
LongformerForQuestionAnswering	HuggingFace in Spark NLP - LongformerForQuestionAnswering	Open in Colab
RoBERTaForQuestionAnswering	HuggingFace in Spark NLP - RoBERTaForQuestionAnswering	Open in Colab
XlmRobertaForQuestionAnswering	HuggingFace in Spark NLP - XlmRobertaForQuestionAnswering	Open in Colab

TF Hub to Spark NLP		
Spark NLP	TF Hub Notebooks	Colab
BertEmbeddings	TF Hub in Spark NLP - BERT	Open in Colab
BertSentenceEmbeddings	TF Hub in Spark NLP - BERT Sentence	Open in Colab
AlbertEmbeddings	TF Hub in Spark NLP - ALBERT	Open in Colab



Search models, datasets, users...

Models

finiteautomata/beto-sentiment-analysis

Text Classification

PyTorch

JAX

Transformers

Spanish

arxiv:2106.09462

bert

Model card

Files and versions

Community

Edit model card

Sentiment Analysis in Spanish

beto-sentiment-analysis

Repository: <https://github.com/finiteautomata/pysentimiento/>

Model trained with TASS 2020 corpus (around ~5k tweets) of several dialects of Spanish.

Base model is BETO, a BERT model trained in Spanish.

[Import Transformers into Spark NLP](#)

Upload trained models to Models Hub

- Trained models can be uploaded and shared via the modelshub
- Zip and Download the model
- Go to <https://modelshub.johnsnowlabs.com/> and upload a zip file

The screenshot shows the 'Step 2' page of the John Snow LABS Modelshub. The top navigation bar includes links for Home, Docs, Learn, Models, Demo, GitHub, and Settings. A progress bar at the top indicates 'Step 1' is complete and 'Step 2' is in progress. On the left, a 'BACK' button is visible. The main area has a heading 'Upload from your local computer or via a link'. A 'Browse...' button shows the selected file 'beto_sentiment_analysis.zip'. Below this, there are several input fields: 'Edition' (set to 'Official'), 'Input Labels' (set to '[document, token]'), 'Output Labels' (set to '[class]'), 'Case sensitive' (set to 'true'), 'Max sentence length' (set to '128'), 'License' (radio buttons for 'Open Source' (selected) and 'Licensed'), 'Tags' (an empty input field with a plus icon), 'Name' (input field containing 'beto_sentiment_analysis'), and 'Language' (a dropdown menu).

Session 1 (Day 2) - Coding Time

- ❖ [Notebook 5 ClassifierDL, SentimentDL, MultiClassifierDL Training](#)
- ❖ [Upload your own model via John Snow Labs Models Hub](#)

Session 2 (Day 2)

- ❖ Spell Checking
- ❖ Keyword extraction with YAKE
- ❖ Rule-based Entity Recognition with EntityRuler
- ❖ Graph triplet extraction

Spark NLP
for Data Scientists



Spell Checking & Correction



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

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

- 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

Context Spell Checker

The Spell Checker can leverage the context of words for ranking different correction sequences. Let's take a look at some examples,

```
# check for the different occurrences of the word "siter"
example1 = ["I will call my siter.", \
    "Due to bad weather, we had to move to a different siter.", \
    "We travelled to three siter in the summer."]
beautify(lp.annotate(example1))
```

```
['I will call my sister .\n',
 'Due to bad weather , we had to move to a different site .\n',
 'We travelled to three sites in the summer .\n']
```

```
# check for the different occurrences of the word "ueather"
example2 = ["During the summer we have the best ueather.", \
    "I have a black ueather jacket, so nice.", \
    "I introduce you to my sister, she is called ueather."]
beautify(lp.annotate(example2))
```

```
['During the summer we have the best weather .\n',
 'I have a black leather jacket , so nice .\n',
 'I introduce you to my sister , she is called Heather .\n']
```

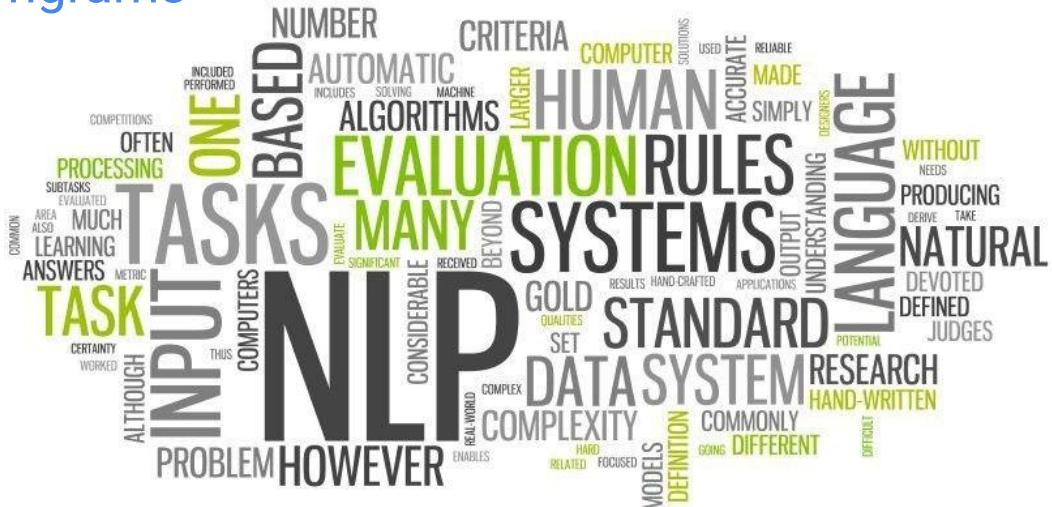
Notice that in the first example, 'siter' is indeed a valid English word,

<https://www.merriam-webster.com/dictionary/siter>

[Notebook 07.0 Spell Checker](#)

Unsupervised Keyword Extraction

YAKE! Is Yet Another Keyword Extraction Algorithm that can extract keywords without any weight by leveraging statistical properties of ngrams



Notebook 08.0

YAKE

Rule-based Entity Recognition with EntityRuler

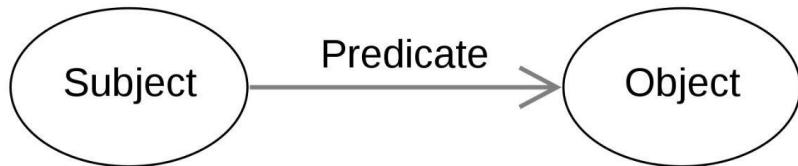
Notebook 11.0 EntityRuler

Fits an Annotator to match exact strings or regex patterns provided in a file against a Document and assigns them an named entity.

```
[  
  {  
    "id": "person-regex",  
    "label": "PERSON",  
    "patterns": ["\w+\s\w+", "\w+-\w+"]  
  },  
  {  
    "id": "locations-words",  
    "label": "LOCATION",  
    "patterns": ["Winterfell"]  
  }]
```

```
{"id": "names-with-j", "label": "PERSON", "patterns": ["Jon", "John", "John Snow"]}  
{"id": "names-with-s", "label": "PERSON", "patterns": ["Stark", "Snow"]}  
{"id": "names-with-e", "label": "PERSON", "patterns": ["Eddard", "Eddard Stark"]}
```

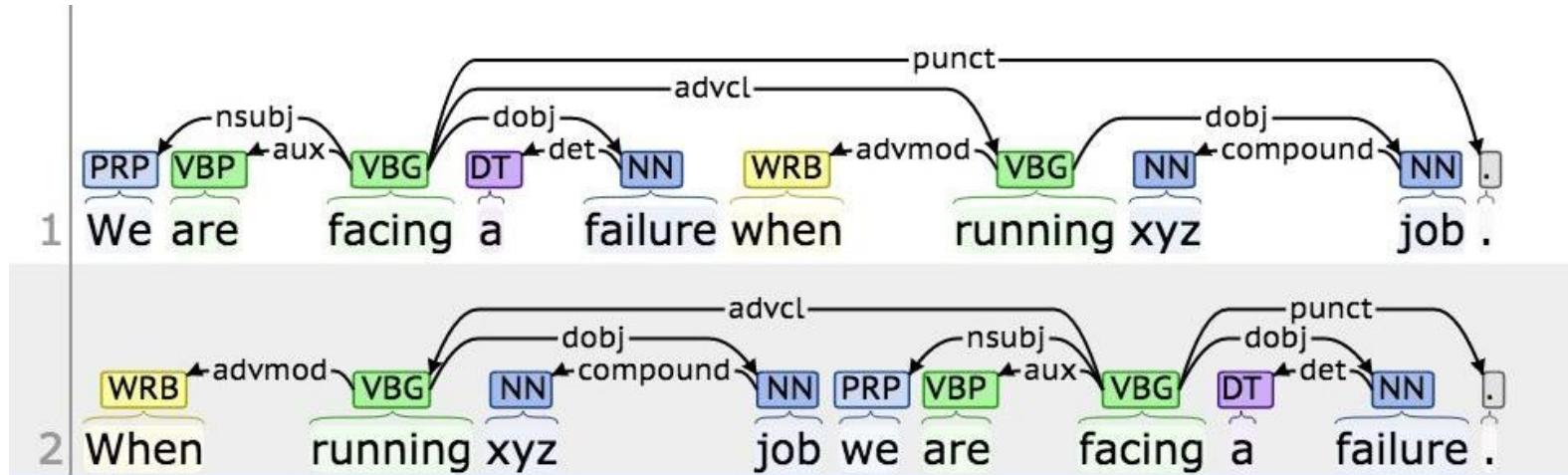
Extract RDF Semantic Triplets



Typed/Untyped Dependency Parsers define a **Predicate**

Relationship between **Subject** and **Object**

Notebook 10.0 Graph Extraction



Session 2 (Day 2) - Coding Time

- ❖ [Notebook 7 Context Spell Checker](#)
- ❖ [Notebook 8 YAKE](#)
- ❖ [Notebook 11 EntityRuler](#)
- ❖ [Notebook 10 RDF Graph Extraction](#)

Session 3 (Day 2)

- ❖ Sequence & Token Classification with Transformers
- ❖ Image Classification with ViT
- ❖ Speech to Text with Wav2Vec2

**Spark NLP
for Data Scientists**



Training & Importing Transformers

Spark NLP - Transformers

Import Transformers into Spark NLP

We have extended support for HuggingFace 😊 and TF Hub exported models since 3.1.0 to equivalent Spark NLP 🚀 annotators. Starting this release, you can easily use the `saved_model` feature in HuggingFace within a few lines of codes and import any `BERT`, `DistilBERT`, `CamemBERT`, `RoBERTa`, `DeBERTa`, `XLM-RoBERTa`, `Longformer`, `BertForTokenClassification`, `DistilBertForTokenClassification`, `AlbertForTokenClassification`, `RoBertaForTokenClassification`, `DeBertaForTokenClassification`, `XlmRoBertaForTokenClassification`, `XlnetForTokenClassification`, `LongformerForTokenClassification`, `CamemBertForTokenClassification`, `CamemBertForSequenceClassification`, `BertForSequenceClassification`, `DistilBertForSequenceClassification`, `AlbertForSequenceClassification`, `RoBertaForSequenceClassification`, `DeBertaForSequenceClassification`, `XlmRoBertaForSequenceClassification`, `XlnetForSequenceClassification`, `LongformerForSequenceClassification`, `BertForQuestionAnswering`, `DeBertaForQuestionAnswering`, `AlbertForQuestionAnswering`, `DistilBertForQuestionAnswering`, `LongformerForQuestionAnswering`, `RoBertaForQuestionAnswering`, `XlmRoBertaForQuestionAnswering`, `TapasForQuestionAnswering`, `Vision Transformers (ViT)`, `HubertForCTC`, `SwinForImageClassification`, and `ConvNextForImageClassification` models to Spark NLP. We will work on the remaining annotators and extend this support to the rest with each release 😊

Image Classification with ViT

Notebook 18 VIT for Image Classification

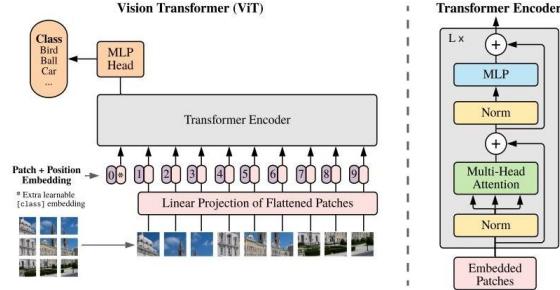
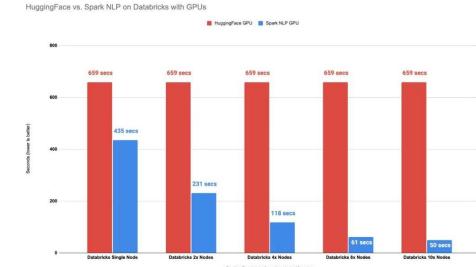


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable “classification token” to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

A screenshot of the HuggingFace interface. At the top, there's a navigation bar with links like REPO, TOP STREETS, MWE, LEARN, ABOUT, HELP, FEATURES, CRYPTO, TUTORIAL DIRECTORY, and NEWSLETTER. Below the navigation bar, a banner says "Scale Vision Transformers (ViT) Beyond Hugging Face". The main content area shows a comparison between HuggingFace and Spark NLP. It includes a 3D diagram of a mobile device displaying a dog's face, with labels for "HuggingFace", "Spark NLP", and "Databricks". Below the diagram, a table compares execution times for different node configurations:

System	1 Node	2 Nodes	4 Nodes	8 Nodes	10 Nodes
HuggingFace GPU	659 secs				
Spark NLP GPU	420 secs	231 secs	118 secs	61 secs	50 secs

Text below the table: "Speed up state-of-the-art ViT models in Hugging Face 😊 up to 2300% (25x times faster) with Databricks, Nvidia, and Spark NLP 🚀".



Spark NLP is 1200% faster than Hugging Face with 10x Nodes

Scale Vision Transformers (ViT) Beyond Hugging Face

Spark NLP vs. HuggingFace



VIT: An Image is Worth 16x16 Words:
Transformers for Image Recognition at
Scale



Spark NLP 1.3x times
faster on CPU



Spark NLP 1.5x times
faster on GPU

Spark NLP is 15% faster than HuggingFace on a single-node Databricks by using only CPUs (with oneDNN enabled)
Spark NLP is 49% faster than HuggingFace on a single-node Databricks by using only GPUs

Over 200 VIT
Models in various
languages and
sizes

Speech to Text Wav2vec 2.0

Notebook 19 Speech2Text with Wav2Vec2

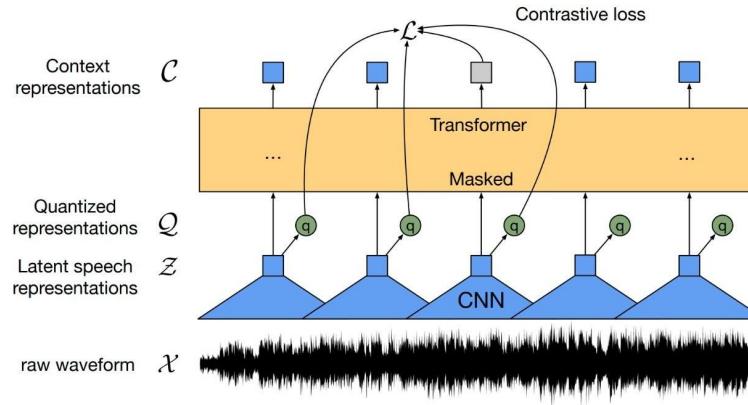


Figure 1: Illustration of our framework which jointly learns contextualized speech representations and an inventory of discretized speech units.

[wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations](#)

Meta

[Over 2000 Wav2Vec Models in various languages and sizes](#)

Session 3 (Day 2) - Coding Time

- ❖ [Notebook 5.2 Transformers for Sequence Classification](#)
- ❖ [Notebook 4.2 Transformers for Token Classification](#)
- ❖ [Notebook 16 Image Classification with Transformers](#)
- ❖ [Notebook 17 Automatic Speech Recognition](#)

Session 4 (Day 2)

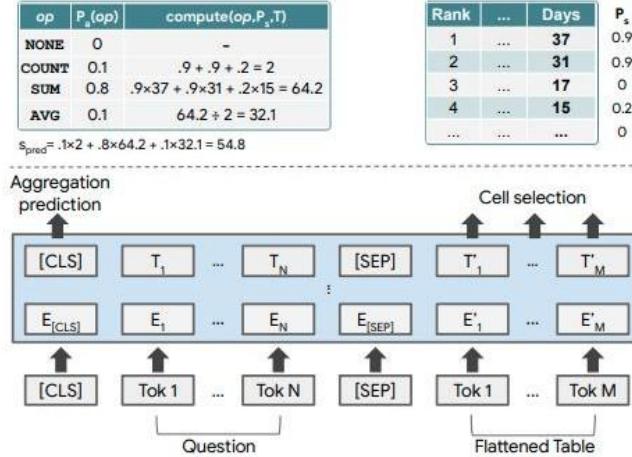
- ❖ Table Question Answering with TAPAS
- ❖ Question Answering, Summarization and other T5 applications
- ❖ Multilingual NLP - Train only on English data and predict for 100+ languages

Spark NLP
for Data Scientists



TAPAS for Table Classification

Notebook 17 Table Question Answering with TAPAS



[TAPAS: Weakly Supervised Table Parsing via Pre-training](#)

TAPAS for Table Question Answering using Spark NLP 🚀

TAPAS is a Zero-shot Question Answering architecture, based on Bert, to carry out Table Understanding. By asking a question on natural language using these models, you can retrieve the content of the cell or cells which best answer to those questions.

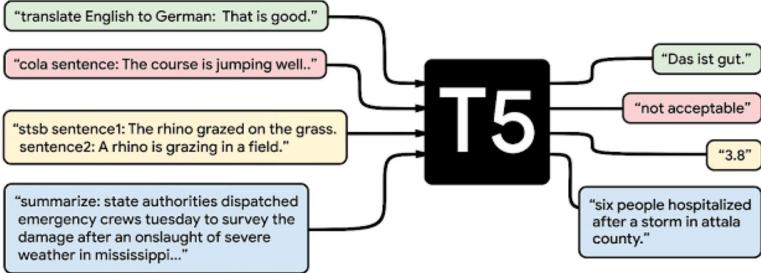
Table				Example questions			
Rank	Name	No. of reigns	Combined days	#	Question	Answer	Example Type
1	Lou Thesz	3	3,749	1	Which wrestler had the most number of reigns?	Ric Flair	Cell selection
2	Ric Flair	8	3,103	2	Average time as champion for top 2 wrestlers?	AVG(3749,3103)=3426	Scalar answer
3	Harley Race	7	1,799	3	How many world champions are there with only one reign?	COUNT(Dory Funk Jr., Gene Kiniski)=2	Ambiguous answer
4	Dory Funk Jr.	1	1,563	4	What is the number of reigns for Harley Race?	7	Ambiguous answer
5	Dan Severn	2	1,559	5	Which of the following wrestlers were ranked in the bottom 3?	{Dory Funk Jr., Dan Severn, Gene Kiniski}	Cell selection
6	Gene Kiniski	1	1,131	6	Out of these, who had more than one reign?	Dan Severn	Cell selection

TAPAS models have been trained using a combination of three datasets:

- [SQA](#), Sequential Question Answering by Microsoft (it was not trained to return aggregation operations as SUM, COUNT, etc - see below)
- [WTQ](#), Wiki Table Questions by Stanford University (with aggregation operations)
- [Wiki SQL](#), by Salesforce, also with aggregation operations.



[Table Classifiers Overview](#)



[Notebook 10 Question Answering and Summarization and more with T5](#)

Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer

1. Text summarization
2. Question answering
3. Translation
4. Generate SQL from natural language text
5. Text style transfer
6. Sentiment analysis
7. Natural Language inference
8. Coreference resolution
9. Sentence Completion
10. Word sense disambiguation



```

# Closed book Question Answering
nlu.load('en.t5').predict('what is the capital of Germany?') # >>> Berlin
# Open Book Question answering
nlu.load('en.t5').predict('Who is president of Nigeria?') # >>> Muhammadu Buhari

# Open book Question Answering
context = 'Peters last week was terrible! He had an accident and broke his leg while skiing!'
question1 = 'Why was peters week so bad?'
question2 = 'How did peter broke his leg?'
nlu.load('answer_question').predict(question1 + context) # >>> broke his leg
nlu.load('answer_question').predict(question2 + context) # >>> skiing

# Big T5 model for Summarization, Sentiment, Text Similarity and other SQuAD/GLUE tasks
pipe = nlu.load('t5')
pipe['t5'].settask('summarize')
pipe.predict(long_text)

```

Every T5 Task with explanation:

Task Name	Explanation
1.CoLA	Classify if a sentence is grammatically correct
2.RTE	Classify whether a statement can be deduced from a sentence
3.MNLI	Classify for a hypothesis and premise whether they contradict or contradict each other or neither of both (3 class).
4.MRPC	Classify whether a pair of sentences is a re-phrasing of each other (semantically equivalent)
5.QNLI	Classify whether the answer to a question can be deducted from an answer candidate.
6.QQP	Classify whether a pair of questions is a re-phrasing of each other (semantically equivalent)
7.SST2	Classify the sentiment of a sentence as positive or negative
8.STSB	Classify the sentiment of a sentence on a scale from 1 to 5 (21 Sentiment classes)
9.CB	Classify for a premise and a hypothesis whether they contradict each other or not (binary).
10.COPA	Classify for a question, premise, and 2 choices which choice the correct choice is (binary).
11.MultiRc	Classify for a question, a paragraph of text, and an answer candidate, if the answer is correct (binary).
12.WIC	Classify for a pair of sentences and a disambiguous word if the word has the same meaning in both sentences.
13.WSC/DPR	Predict for an ambiguous pronoun in a sentence what it is referring to.
14.Summarization	Summarize text into a shorter representation.
15.SQuAD	Answer a question for a given context.
16.WMT1	Translate English to German
17.WMT2	Translate English to French
18.WMT3	Translate English to Romanian

100+ Languages supported by Language-agnostic BERT Sentence Embedding (LABSE) and XLM-RoBERTa

Train in 1 Language, predict in 100+ different languages

Notebook 5.2 Training Multilingual Classifier

```
# Binary Class Classifier, 2 classes
nlu.load('xx.embed_sentence.labse train.sentiment').fit(train_df).predict(test_df)

# Multi Class Classifier, N classes
nlu.load('xx.embed_sentence.labse train.classifier').fit(train_df).predict(test_df)

# Multi Class Classifier with multiple labels example (i.e. Hashtags)
# N classes, where one row can be assigned up to N labels
nlu.load('xx.embed_sentence.labse train.multi_classifier').fit(train_df).predict(test_df)
```

ISO	NAME	ISO	NAME	ISO	NAME
af	AFRIKAANS	ht	HAITIAN_CREOLE	pt	PORTRUGUESE
am	AMHARIC	hu	HUNGARIAN	ro	ROMANIAN
ar	ARABIC	hy	ARMENIAN	ru	RUSSIAN
as	ASSAMESE	id	INDONESIAN	rw	KINYARWANDA
az	AZERBAIJANI	ig	IGBO	si	SIKHALESE
be	BELARUSIAN	is	ICELANDIC	sk	SLOVAK
bg	BULGARIAN	it	ITALIAN	sm	SAMOAN
bn	BENGALI	ja	JAPANESE	sn	SHONA
bo	TIBETAN	JV	JAVANESE	so	SOMALI
bs	BOSNIAN	ka	GEORGIAN	sq	ALBANIAN
ca	CATALAN	kk	KAZAKH	sr	SERBIAN
ceb	CEBUANO	km	KHMER	st	SESOTHO
co	CORSICAN	kn	KANNADA	su	SUNDANESE
cs	CZECH	ko	KOREAN	sv	SWEDISH
cy	WELSH	ku	KURDISH	sw	SWAHILI
da	DANISH	ky	KYRGYZ	ta	TAMIL
de	GERMAN	la	LATIN	te	TELUGU
el	GREEK	lb	LUXEMBOURGISH	th	THAI
en	ENGLISH	lo	LAOTHIAN	tg	TAJIK
eo	ESPERANTO	lt	LITHUANIAN	tk	TURKMEN
es	SPANISH	lv	LATVIAN	tl	TAGALOG
et	ESTONIAN	mg	MALAGASY	tr	TURKISH
eu	BASQUE	mi	MAORI	tt	TATAR
fa	PERSIAN	mk	MACEDONIAN	ug	UIGHUR
fi	FINNISH	ml	MALAYALAM	uk	UKRAINIAN
fr	FRENCH	mn	MONGOLIAN	ur	URDU
fy	FRISIAN	mr	MARATHI	uz	UZBEK
ga	IRISH	ms	MALAY	vi	VietNAMESE
gd	SCOTS_GAELIC	mt	MALTESE	wo	WOLOF
gl	Galician	my	BURMESE	xh	XHOSA
gu	GUARATI	ne	NEPALI	yi	YIDDISH
ha	HAUSA	nl	DUTCH	yo	YORUBA
haw	HAWAIIAN	no	NORWEGIAN	zh	Chinese
he	HEBREW	ny	NYANJA	zu	ZULU
hi	HINDI	or	ORIYA		
hmn	HMONG	pa	PUNABI		
hr	CROATIAN	pl	POLISH		

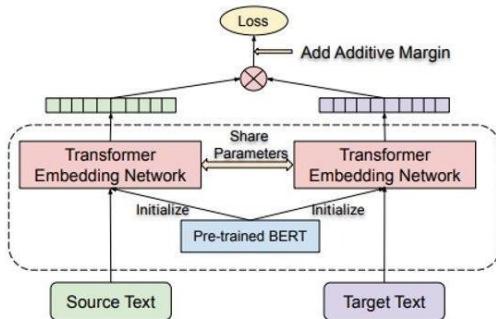


Figure 1: Dual encoder model with BERT based encoding modules.

Spark NLP Resources

[Spark NLP Official page](#)

[Spark NLP Workshop Repo](#)

[JSL Youtube channel](#)

[JSL Blogs](#)

[Introduction to Spark NLP: Foundations and Basic Components \(Part-I\)](#)

[Introduction to: Spark NLP: Installation and Getting Started \(Part-II\)](#)

[Named Entity Recognition with Bert in Spark NLP](#)

[Text Classification in Spark NLP with Bert and Universal Sentence Encoders](#)

[Spark NLP 101 : Document Assembler](#)

[Spark NLP 101: LightPipeline](#)

<https://www.oreilly.com/radar/one-simple-chart-who-is-interested-in-spark-nlp/>

<https://blog.dominodatalab.com/comparing-the-functionality-of-open-source-natural-language-processing-libraries/>

<https://databricks.com/blog/2017/10/19/introducing-natural-language-processing-library-apache-spark.html>

<https://databricks.com/fr/session/apache-spark-nlp-extending-spark-ml-to-deliver-fast-scalable-unified-natural-language-processing>

<https://medium.com/@saif1988/spark-nlp-walkthrough-powered-by-tensorflow-9965538663fd>

<https://www.kdnuggets.com/2019/06/spark-nlp-getting-started-with-worlds-most-widely-used-nlp-library-enterprise.html>

<https://www.forbes.com/sites/forbestechcouncil/2019/09/17/winning-in-health-care-ai-with-small-data/#1b2fc2555664>

<https://medium.com/hackernoon/mueller-report-for-nerds-spark-meets-nlp-with-tensorflow-and-bert-part-1-32490a8f8f12>

<https://www.analyticsindiamag.com/5-reasons-why-spark-nlp-is-the-most-widely-used-library-in-enterprises/>

<https://www.oreilly.com/ideas/comparing-production-grade-nlp-libraries-training-spark-nlp-and-spacy-pipelines>

<https://www.oreilly.com/ideas/comparing-production-grade-nlp-libraries-accuracy-performance-and-scalability>

<https://www.infoworld.com/article/3031690/analytics/why-you-should-use-spark-for-machine-learning.html>

<https://www.udemy.com/course/spark-nlp-for-data-scientists/>

Session 4 (Day 2) - Coding Time

- ❖ [Notebook 15 Table Question Answering with TAPAS](#)
- ❖ [Notebook 13 Question Answering and Summarization and more with T5](#)
- ❖ [Notebook 5.3 Training Multilingual Classifier](#)

**Thank
You!**