Detect person names in text: Part 1 (Results)

Jan Procházka / January 14, 2021 /
Deep Learning (https://pii-tools.com/category/deep-learning/),
Personal Data (https://pii-tools.com/category/personal-data/),
Whitepaper (https://pii-tools.com/category/whitepaper/)

Detecting people's names is part and parcel of PII discovery. Traditional techniques like regexps and keywords don't work, because the set of all names is too varied. How do open source Named Entity Recognition (NER) engines compare, and can we do better?

This Part 1 has NER results and benchmarks. There's also Part 2 with technical neural network details (https://pii-tools.com/detect-person-names-in-text-part-2-technical/).

named entity recognizer

A good named entity recognizer (NER) is essential for detecting names, home addresses, passport scans (https://twitter.com/PII_tools/status/1230905790745260032) etc, for purposes of compliance or breach incident management (https://pii-tools.com/pii-exclusions-data-breach/).

One could use a prepared list of names and surnames for this task, but such gazetteer obviously cannot be exhaustive and will fail on sentences like this:

> "**Calvin Klein** founded Calvin Klein Company in 1968."

Humans recognize easily what is a person's name and what isn't. For machines, the task is more difficult because they have trouble understanding context. This leads to the famous two types of errors:

- **False positives**: words detected as personal names that are not, typically because they're capitalized (`"Skin Games and Jesus Jones were one the first western` lowercased, foreign or uncommon. Example: `"bill carpenter was playing football."`

So in order to recognize person names in text it is necessary to know not only what names look like, but also in what context they're used and have a general domain knowledge.

Why not just use open source?

NER is a well-studied task in academia. Naturally, we turned to open source NER solutions first. We evaluated the most popular ready-made software: Stanford NER and Stanza from the Stanford University, FLAIR from Zalando Research, spaCy from Explosion AI.

To cut the story short, **none of these open source tools were precise and fast enough** for our purposes. While they work great on well-behaved data such as news articles or Wikipedia, their academic pedigree implodes when applied to the wild, messy documents of the real world.

Tools (https://pii-tools.com/)) but we decided to share some technical design tips and results here, in the hope they help others.

In this article, we'll compare our creation against popular open source options. Additionally, we'll benchmark a simple gazetteer-based NER that uses a predefined list of names, to serve as a **baseline**.

Description of tested NERs

	version	type	language	source
list	-	list of names (contains 5.6M unique names)	multi-language	inhouse
stanford	4.1	CRF classifier	single (each language has a separate model)	https://nlp.stanford.edu/software/CRF-NER.html (https://nlp.stanford.edu/software/CRF-NER.html)

		network	separate model)	(https://stanfordnlp.github.io/stanza/)
flair (ner-multi-fast model)	0.6.1	neural network	multi-language	https://github.com/flairNLP/flair (https://github.com/flairNLP/flair)
spacy (xx_ent_wiki_sm model)	2.3.2	neural network	multi-language	https://spacy.io/ (https://spacy.io/)
pii-tools	3.8.0	neural network	multi-language	https://pii-tools.com/ (https://pii-tools.com/)

Our requirements for the new NER were:

- **multi-language**, with special focus on English, Spanish, German and Portuguese/Brazilian;
- to accept **arbitrary document contexts**: text coming from PDFs, Word documents, Excel, email, database field, OCR…;
- **efficient**, to process **large amounts of text quickly on a CPU** (no need for specialized GPU hardware) and with a **low memory footprint**,
- **flexible** to be able to evolve the model behaviour: retraining, adding new languages, correcting detection errors;
- and of course to be **accurate**, to minimize false positives and negatives.

2003 (https://www.clips.uantwerpen.be/conll2003/ner/) (English, German) and LeNER-Br (https://cic.unb.br/~teodecampos/LeNER-Br/) (Portuguese).

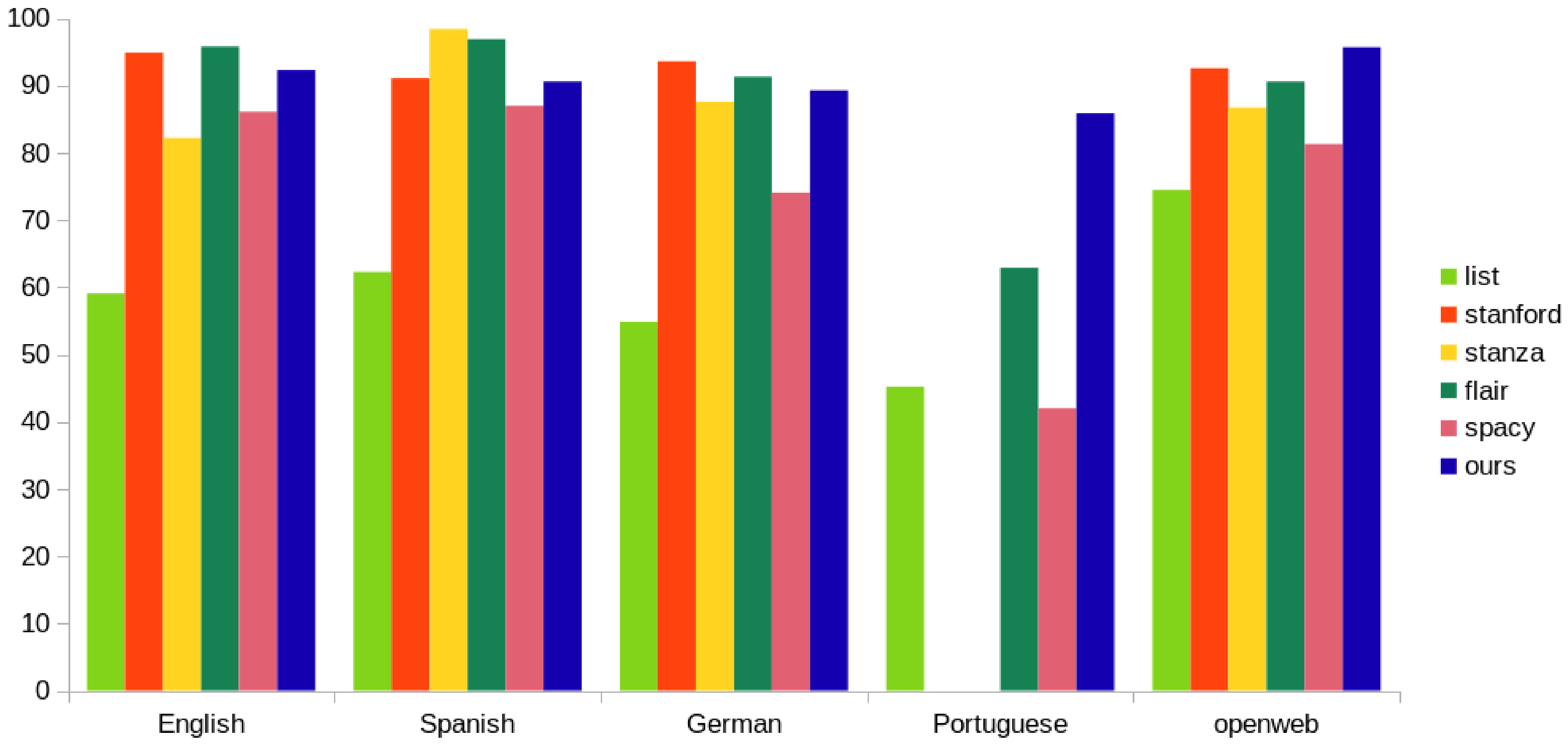
We also included a manually annotated `openweb` dataset, to make sure we test on data that no NER system (including ours!) has seen during training. Text for this dataset was randomly sampled from the English OpenWebTextCorpus (https://skylion007.github.io/OpenWebTextCorpus/). We value results on `openweb` the most, because OpenWeb reflects the real (messy) data found in real documents the closest, among these public datasets.

Accuracy

We measured F1 scores (https://en.wikipedia.org/wiki/F-score) for person names detected by each software. F1 scores range from 0 to 100, the higher the better. A "hit" (true positive) means the entire name was matched exactly, beginning to end. This is the strictest metric; we specifically don't calculate success over the number of correctly predicted tokens, or even individual characters.

Performance benchmark [F1 score – higher is better]

stanford	94.9	91.1	93.6	-- (no model available)	92.6
stanza	82.2	98.4	87.6	-- (no model available)	86.7
flair	95.8	96.9	91.3	62.9	90.6
spacy	86.1	87.0	74.1	42.0	81.3
pii-tools	92.3	90.6	89.3	85.9	95.7



In all tests our NER is among the top contenders – but keep in mind that accuracy was just one of our 5 design goals. For example, we "lost" a few F1 points on purpose by switching from Tensorflow to TF lite, trading accuracy for a much smaller & faster model.

For the Portuguese language (Brazil, LGPD) and `openweb` dataset, PII Tools is the clear winner. As mentioned above `openweb` reflects "real data" the closest, so this is great.

Let's look at some concrete examples:

Text samples and NER detections (actual expected names are **in bold**)

	stanford	**stanza**	**flair**	**spacy**	**pii-tools**
Calvin Klein founded Calvin Klein Company in 1968.	ok	ok	ok	ok	ok
Nawab Zulfikar Ali Magsi did not see **Shama Parveen Magsi** coming.	ok	ok	ok	fail	ok
bill carpenter was playing football.	fail	fail	fail	ok	ok
Llanfihangel Talyllyn (a village in Wales) is beautiful.	fail	fail	fail	fail	ok

These samples are rather edge cases, but serve to get an idea of what detectors have to deal with, and how successful they are.

Performance

Now we will look into other aspects of our requirements. Detection performance was measured on CPU, our `pii-tools` NER was artificially restricted to a single process with a single thread, others were left in default settings.

Speed and memory comparison

	speed short [kB/s] (short documents of ~500 bytes)	speed long [kB/s] (long documents of ~2.5 kB)	startup time [s] (time to initializate NER + tag a 20 character text)	RAM usage [MB] (peak RAM while tagging 270 kB of text)
list	350.0	1100.0	1.1	1420
stanford	30.0	30.0	1.5	248 (for one language only)

flair	0.1	0.1	5.2	5341 (input had to be chunked to limit RAM)
spacy	70.0	200.0	0.5	123
pii-tools	35.0	230.0	1.3	387

Overall FLAIR and Stanza are definitely out, due to their super slow speed and high RAM usage. A worthy competitor from the performance perspective is spaCy, whose authors put a great deal of effort into optimization. Unfortunately spaCy's tokenization quirks and opinionated architecture proved too inflexible for our needs.

Likewise, Stanford NER is the most accurate among the open source alternatives, but is quite rigid – it's really hard to update its models or add a new language. Plus its GNU GPL license won't be to everyone's liking.

Flexibility

appear), fixing detection errors.

In order to adapt the PII Tools NER model quickly, we built a pipeline that utilizes several "weaker" NERs and automatic translation tools to build a huge training corpus from varied sources. This focus on real-world data, along with a robust automated Tensorflow training pipeline allows adding new languages and controlling NER outputs more easily than the open source solutions.

While developing our `pii-tools` NER we implemented most components from scratch, including:

- a large scale annotated dataset (proprietary data)
- a tokenizer (critical; none of the open source variants do this part well)
- token features (input to the neural network)
- convolutional neural network (NN architecture)
- data augmentation and training pipeline (for grounded model updates)
- various tricks to push performance or to squeeze a lot of information into a smaller parameter space

Questions? Want to see a live demo? Contact us (https://pii-tools.com/schedule-a-demo/).

Download our AI whitepaper

Detecting Personal Names in Text

Your name

Your email address

□ Agree to Privacy Policy (/privacy-policy/). We never share your data with third parties. Unsubscribe

Download Whitepaper

🏷Tags:detect names (https://pii-tools.com/tag/detect-names/), named entity recognition (https://pii-tools.com/tag/named-entity-recognition/), names in text (https://pii-tools.com/tag/names-in-text/), NER (https://pii-tools.com/tag/ner/), NER benchmark (https://pii-tools.com/tag/ner-benchmark/), open source (https://pii-tools.com/tag/open-source/), person names (https://pii-tools.com/tag/person-names/), unstructured text (https://pii-tools.com/tag/unstructured-text/)

Meet the author

Jan Procházka's bio:

Data science engineer at PII Tools in one life. Accredited physiotherapist in another.

Schedule a demo

Your Name (required)

Your Email (required)

Send

Recent Posts

- PII De-Identification vs. Masking vs. Redaction (https://pii-tools.com/pii-de-identification-vs-masking-vs-redaction/)
- The New CPRA Umbrella Covers HR Data (https://pii-tools.com/the-new-cpra-umbrella-covers-hr-data/)
- Regular Audits: Do You Really Need Them? (https://pii-tools.com/regular-audits-do-you-really-need-them/)
- Do They Even Matter?—The 3 Largest GDPR Fines To Date (https://pii-tools.com/do-they-even-matter-the-3-largest-gdpr-fines-to-date/)

HOME (HTTPS://PII-TOOLS.COM/)
/ PRODUCT TOUR (HTTPS://PII-TOOLS.COM/PRODUCT-TOUR/)
/ PRICING (HTTPS://PII-TOOLS.COM/PRICING/)
/ SCHEDULE A DEMO (HTTPS://PII-TOOLS.COM/SCHEDULE-A-DEMO/)
/ TERMS AND CONDITIONS – SELF-HOSTED (HTTPS://PII-TOOLS.COM/TERMS-AND-CONDITIONS/)
/ TERMS AND CONDITIONS – SAAS (HTTPS://PII-TOOLS.COM/TERMS-AND-CONDITIONS-CLOUD/)
/ PRIVACY POLICY (HTTPS://PII-TOOLS.COM/PRIVACY-POLICY/)
/ CONTACT US (HTTPS://PII-TOOLS.COM/CONTACT-US/)