

Categorizing Comparative Sentences

Alexander Panchenko^{‡,★}, Alexander Bondarenko[†], Mirco Franzek[‡], Matthias Hagen[†], and Chris Biemann[‡]

[‡]Universität Hamburg, Germany [†] Martin-Luther-Universität Halle-Wittenberg, Germany [★] Skolkovo Institute of Science and Technology, Russia

Key Ideas

- **Task:** Identifying comparative sentences and categorizing the intended preference (e.g., "Python has better NLP libraries than MATLAB" → Python, better, MATLAB).
- **New cross-domain dataset:** CompSent-19 corpus consisting of 7,199 sentences with item pairs from three domains: "Brands", "Computer Science", and "General/Random".
- **Evaluation:** A gradient boosting model based on pre-trained sentence embeddings achieves an F1 score of 85% improving upon a strong rule-based baseline.
- **Model Application:** Extracting comparative sentences for pro/con argumentation in argument search engines or debating systems.

CompSent-19 Characteristics

Domain	BETTER	WORSE	NONE	Total
CompSci	581	248	1,596	2,425
Brands	404	167	1,764	2,335
Random	379	178	1,882	2,439
Total	1,364	593	5,242	7,199

Examples of the Label Comparative Sentences

The **first item** is BETTER/WORSE/NONE than the **second item**. The ordering of the items is important.

Domain	Sentence	Label
CompSci	This time Windows 8 was roughly 8 percent slower than Windows 7 .	WORSE
Brands	These include Motorola , Samsung and Nokia .	NONE
Random	Right now, I think tennis is easier than baseball .	BETTER

Downloads and Demo

Dataset:

<https://zenodo.org/record/3237552>

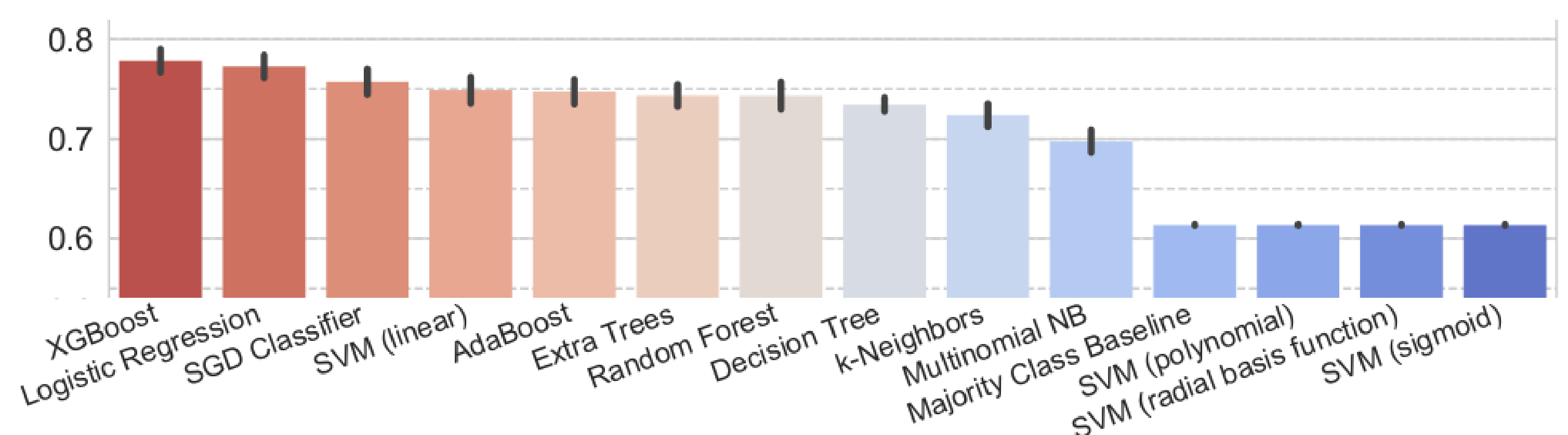
Code:

<https://github.com/uhh-lt/comparative>

Demo:

ltdemos.informatik.uni-hamburg.de/cam/

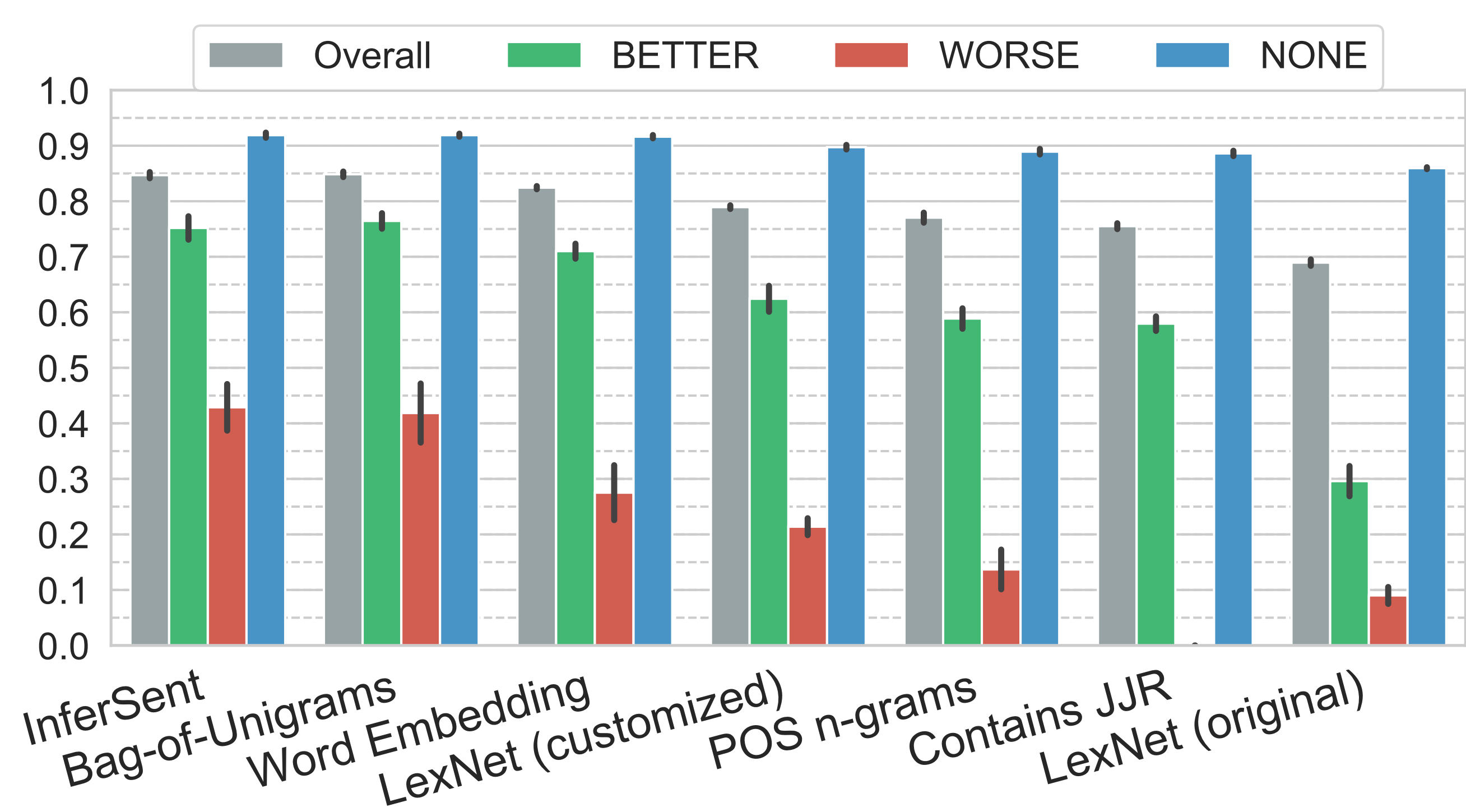
Impact of Classification Model (Bag-of-Words Representation)



Rule-based vs. XGBoost

Model	BETTER	WORSE	NONE	ALL
Rule-based Baseline	0.65	0.44	0.90	0.82
XGBoost+InferSent	0.75	0.43	0.92	0.85

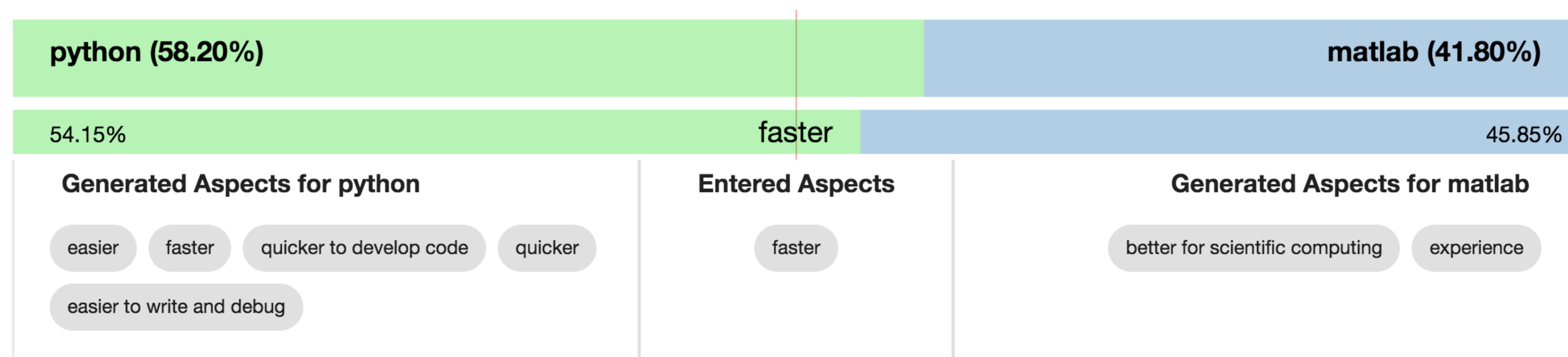
Impact of Feature Representation (XGBoost with different features)



Cross-Domain Transfer

Domain	CompSci	Brands	Random
CompSci	0.82	0.84	0.84
Brands	0.76	0.83	0.83
Random	0.79	0.84	0.86

Comparative Argument Mining: "Python vs. MATLAB"



Wow, **Python** much **faster** than **MatLab** .

RE: Wow, **Python** much **faster** than **MatLab** .

Remember that **Python** with NumPy tend to be **faster** than **Matlab**.

Python might be **faster** [Click, to show context.](#) I'm not good at **MATLAB** so I don't know how to get computational times (or in **Python**, for that matter).

As you can see from the results- **Matlab** is significantly **faster** than **python**.

Right, exactly; but "flat" **Matlab** (that is, **Matlab** with few looping constructs) has been shown to me to be **faster** than **Python**+NumPy for intensive calculations.

But I also tested with 64 bit float maxtrix and on my machine, **Matlab** 2010b is still **faster** than **Python** 3.2 with Numpy-MKL