AMuSE-WSD: An All-in-one Multilingual System for Easy Word Sense Disambiguation

Riccardo Orlando
Babelscape, Italy
orlando@babelscape.com

Simone Conia
Sapienza University of Rome
conia@di.uniroma1.it

Fabrizio Brignone
Babelscape, Italy
brignone@babelscape.com

Francesco Cecconi
Babelscape, Italy
cecconi@babelscape.com

Roberto Navigli
Sapienza University of Rome
navigli@diag.uniroma1.it

Abstract

Over the past few years, Word Sense Disambiguation (WSD) has received renewed interest: recently proposed systems have shown the remarkable effectiveness of deep learning techniques in this task, especially when aided by modern pretrained language models. Unfortunately, such systems are still not available as ready-to-use end-to-end packages, making it difficult for researchers to take advantage of their performance. The only alternative for a user interested in applying WSD to downstream tasks is to use currently available end-to-end WSD systems, which, however, still rely on graph-based heuristics or non-neural machine learning algorithms. In this paper, we fill this gap and propose AMuSE-WSD, the first end-to-end system to offer high-quality sense information in 40 languages through a state-of-the-art neural model for WSD. We hope that AMuSE-WSD will provide a stepping stone for the integration of meaning into real-world applications and encourage further studies in lexical semantics. AMuSE-WSD is available online at http://nlp.uniroma1.it/amuse-wsd.

1 Introduction

Word Sense Disambiguation (WSD) is the task of associating a word in context with its most appropriate sense from a predefined sense inventory (Bevilacqua et al., 2021). Learning the meaning of a word in context is often considered to be a fundamental step in enabling machine understanding of text (Navigli, 2018): indeed, a word can be polysemous, that is, it can refer to different meanings depending on the context. Over the past few years, WSD has received growing attention and has been proven to be useful in an increasing range of applications, such as machine translation (Liu et al., 2018; Pu et al., 2018; Raganato et al., 2019), information extraction (Zhong and Ng, 2012; Delli Bovi et al., 2015), information retrieval (Blloshmi et al., 2021), text categorization (Shimura et al., 2019) and question answering (Ramakrishnan et al., 2003).

WSD approaches usually fall into two categories: knowledge-based (Moro et al., 2014; Agirre et al., 2014; Chaplot and Salakhutdinov, 2018), which leverage computational lexicons, and supervised (Bevilacqua and Navigli, 2020; Blevins and Zettlemoyer, 2020; Conia and Navigli, 2021; Barba et al., 2021; ElSheikh et al., 2021), which train machine learning models on sense-annotated data. While early work mainly belongs to the former category (Navigli, 2009), recent studies have shown the superiority in performance of the latter category, especially thanks to complex neural networks (Bevilacqua et al., 2021). However, as WSD systems become more and more reliant on increasingly complex input representations – pretrained language models are becoming the *de facto* representation method in several NLP tasks – and involve neural architectures, the entry requirements for end users have also become higher. Indeed, the implementation of state-of-the-art WSD systems is frequently anything but straightforward. What is more, such systems are often not ready to be used off-the-shelf and require additional preprocessing and postprocessing modules for document splitting, tokenization, lemmatization and part-of-speech tagging. These complications may make the use of recent high-performing WSD systems unattractive, or even out of reach, for researchers who want to take advantage of explicit semantic information in other areas of research, but who are not experts in WSD.

In order to make WSD more accessible, there have been several attempts at providing ready-to-use WSD systems that can be easily integrated into other systems (Navigli and Ponzetto, 2012b; Moro et al., 2014; Agirre et al., 2014; Scozzafava et al., 2020). Nevertheless, current ready-to-use WSD systems are either English-only or based on approaches that now lag behind state-of-the-art mod-

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 298–307
November 7–11, 2021. ©2021 Association for Computational Linguistics

els in terms of performance.

In this paper, we fill the gap and present AMuSE-WSD, an easy-to-use, off-the-shelf WSD package that provides sense annotations in multiple languages through a state-of-the-art neural-based model. The main features of AMuSE-WSD can be summarized as follows:

- We propose the first ready-to-use WSD package to offer a multilingual WSD system built on top of modern pretrained language models;
- AMuSE-WSD offers an easy-to-use REST API that can be queried either online for ease of use, or offline to minimize inference times;
- Our system comes with a Web interface to let users disambiguate short documents on the fly without a single line of code;
- We support 40 languages offline and 10 languages online.

We hope that AMuSE-WSD will facilitate the integration of semantic information into tasks that may benefit from high-quality sense information, enabling the exploitation of WSD in multiple languages.[1]

2 System Description

AMuSE-WSD is the first all-in-one multilingual system for WSD based on a state-of-the-art neural model. Our system encapsulates this model in a pipeline which provides word-level semantic information in an end-to-end fashion so that a user is not required to provide anything more than raw text as input. In this Section, we provide a description of the various components of our model, focusing on its preprocessing pipeline (Section 2.1) and our WSD system (Section 2.2), which is the core of AMuSE-WSD. Furthermore, we provide an overview of its performance on several standard benchmarks for WSD (Section 2.3).

2.1 Preprocessing

Preprocessing is one important aspect that is often overlooked by state-of-the-art models built for research purposes, as they often rely on pre-parsed documents that are already split into sentences, tokenized, lemmatized and PoS-tagged. However, a good preprocessing pipeline is a necessary condition for a high-quality WSD system. Indeed, the most popular sense inventories for WSD, e.g. WordNet (Miller, 1992) and BabelNet (Navigli and Ponzetto, 2012a; Navigli et al., 2021), define the possible meanings of a word with respect to its lemma and PoS tag. Therefore, an accurate preprocessing pipeline is fundamental in order to generate the correct candidate set of possible meanings.

AMuSE-WSD's preprocessing pipeline takes advantage of two popular toolkits, spaCy (Honnibal et al., 2020) and Stanza (Qi et al., 2020), to provide high-quality document splitting, tokenization, lemmatization and PoS tags to the core WSD model. We provide more technical details about the features of our preprocessing pipeline in Section 3.

2.2 Model Architecture

The core of AMuSE-WSD is its WSD model. Since the main objective of our system is to provide the best possible automatic annotations for WSD, our package features a reimplementation of the state-of-the-art WSD model proposed recently by Conia and Navigli (2021). Differently from other ready-to-use WSD packages which are based on graph-based heuristics (Moro et al., 2014; Scozzafava et al., 2020) or non-neural models (Papandrea et al., 2017), this neural architecture is built on top of a Transformer encoder (Vaswani et al., 2017). More specifically, given a word w in context, the WSD model i) builds a contextualized representation $\mathbf{e}_w \in \mathbb{R}^{d_L}$ of the word w as the average of the hidden states of the last four layers of a pretrained Transformer encoder L, ii) applies a non-linear transformation to obtain a sense-specific hidden representation $\mathbf{h}_w \in \mathbb{R}^{d_\text{h}}$, and finally iii) computes the output score distribution $\mathbf{o}_w \in \mathbb{R}^{|S|}$ over all the possible senses of a sense inventory S. More formally:

$$\begin{aligned}
\mathbf{e}_w &= \text{BatchNorm}\left(\frac{1}{4}\sum_{i=1}^{4}\mathbf{l}_w^{-k}\right)\\
\mathbf{h}_w &= \text{Swish}(\mathbf{W}_h\mathbf{e}_w + \mathbf{b}_h)\\
\mathbf{o}_w &= \mathbf{W}_o\mathbf{h}_w + \mathbf{b}_o
\end{aligned}$$

where $\mathbf{l}_w^{-k}$ is the hidden state of the k-th layer of L from its topmost layer, $\text{BatchNorm}(\cdot)$ is the batch normalization operation, and $\text{Swish}(x) = x \cdot \text{sigmoid}(x)$ is the Swish activation function (Ramachandran et al., 2018).

We follow Conia and Navigli (2021) in framing WSD as a multi-label classification problem in

[1] AMuSE-WSD can be downloaded for offline use upon request at http://nlp.uniroma1.it/resources. It is licensed under Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International.

		English datasets						Multilingual datasets		
		SE2	SE3	SE07	SE13	SE15	ALL	SE13	SE15	XL-WSD
WSD Modules	BERT-large	76.3	73.2	66.2	71.7	74.1	73.5	–	–	–
	Conia and Navigli (2020)	77.1	76.4	70.3	76.2	77.2	76.4	–	–	–
	Scarlini et al. (2020)	78.0	77.1	71.0	77.3	83.2	77.9	78.3	70.8	–
	Blevins and Zettlemoyer (2020)	79.4	77.4	74.5	79.7	81.7	79.0	–	–	–
	Bevilacqua and Navigli (2020)	80.8	79.0	75.2	80.7	81.8	80.1	80.3	70.7	–
	Conia and Navigli (2021)	80.4	77.8	76.2	81.8	83.3	80.2	–	–	–
End-to-End Systems	Moro et al. (2014)	67.0	63.5	51.6	66.4	70.3	65.5	65.6	–	52.9
	Papandrea et al. (2017)	73.8	70.8	64.2	67.2	71.5	–	–	–	–
	Scozzafava et al. (2020)	71.6	72.0	59.3	72.2	75.8	71.7	73.2	66.2	57.7
	AMuSE-WSD$_{\text{BERT-large}}$	**80.6**	**78.4**	**76.5**	**81.0**	82.7	**80.2**	–	–	–
	AMuSE-WSD$_{\text{XMLR-large}}$	79.5	77.6	74.1	79.9	**83.4**	79.3	**80.0**	**73.0**	**67.3**
	AMuSE-WSD$_{\text{XMLR-base}}$	77.8	76.0	72.1	77.7	81.5	77.5	76.8	73.0	66.2
	AMuSE-WSD$_{\text{m-MiniLM}}$	76.3	72.4	69.5	76.1	77.8	75.1	74.5	69.6	63.9

Table 1: English WSD results in F_1 scores on Senseval-2 (SE2), Senseval-3 (SE3), SemEval-2007 (SE07), SemEval-2013 (SE13), SemEval-2015 (SE15), and the concatenation of all the datasets (ALL). We also include results on multilingual WSD in SemEval-2013 (DE, ES, FR, IT), SemEval-2015 (IT, ES), and XL-WSD (average over 17 languages, English excluded). We distinguish between *WSD Modules*, that is, research systems that need to be inserted into a pipeline, and *End-to-End WSD Systems*. Best results among end-to-end systems in **bold**.

which the model can learn to assign multiple valid senses to each target word. Indeed, the "boundaries" between different senses of a polysemous word are not always clear cut or well defined (Erk and McCarthy, 2009), often leading to cases in which, given a word in context, more than one meaning is deemed appropriate by human annotators. Framing WSD as a multi-label classification problem allows the model to take advantage of such cases. In particular, this means that the model is trained to predict whether a sense $s \in S_w$ is appropriate for a word w in a given context, independently of the other senses in S_w.

2.3 Evaluation

Datasets. We compare the performance of our model against currently available end-to-end WSD systems on the unified evaluation framework for English all-words WSD proposed by Raganato et al. (2017). This evaluation includes five gold datasets, namely, Senseval-2 (Edmonds and Cotton, 2001), Senseval-3 (Mihalcea et al., 2004), SemEval-2007 (Agirre and Soroa, 2007), SemEval-2013 (Navigli et al., 2013), and SemEval-2015 (Moro and Navigli, 2015). We also evaluate our model in multilingual WSD using the French, German, Italian and Spanish datasets provided as part of SemEval-2013, and the Italian and Spanish datasets of SemEval-2015. Finally, we evaluate AMuSE-WSD on XL-WSD (Pasini et al., 2021), a new multilingual dataset which comprises 17 languages.

Experimental setup. We evaluate how the performance of AMuSE-WSD varies when using four different pretrained language models to represent input sentences: a high-performing English-only version based on BERT-large-cased (Devlin et al., 2019), a high-performing multilingual version based on XLM-RoBERTa-large (Conneau et al., 2020), a multilingual version that relies on the smaller XLM-RoBERTa-base to balance quality and inference time, and a multilingual version based on Multilingual-MiniLM (Wang et al., 2020) that minimizes inference time while still providing good results. In any case, the weights of the underlying language model are left frozen, that is, they are not updated during training. Each model is trained for 25 epochs using Adam (Kingma and Ba, 2015) with a learning rate of 10^{-4}. We train each model configuration on the union of SemCor (Miller et al., 1994) and the WordNet Gloss Corpus. Following standard practice, we perform model selection choosing the checkpoint with highest F1 score on SemEval-2007, the smallest evaluation dataset.

Results. Table 1 shows how AMuSE-WSD performs in comparison to currently available end-to-end WSD systems, that is, systems that only require raw text in input. AMuSE-WSD (XLMR-large) offers a very significant improvement over SyntagRank (Scozzafava et al., 2020), the previous best end-to-end system, both in English WSD (+7.6 in F_1 score on the concatenation of all the English test sets) and multilingual WSD (+6.8, +6.8 and +9.6 in F_1 score on SemEval-2013, SemEval-2015 and XL-WSD, respectively). It is worth noting that even the lightest – and therefore faster – configuration of AMuSE-WSD (m-MiniLM) still shows significant improvements over SyntagRank (+3.4 and +6.2 in F_1 score on ALL and XL-WSD, respectively). For completeness, Table 1 also reports the performance of recently proposed, non end-to-end modules for WSD.

3 AMuSE-WSD API

AMuSE-WSD can easily be used to integrate sense information into downstream tasks. Indeed, AMuSE-WSD is a simple solution for out-of-the-box sense disambiguation as it offers a RESTful API that provides access to a full end-to-end state-of-the-art pretrained model for multilingual WSD. The main advantage of our system is that it is fully self-contained as it takes care of the preprocessing information usually needed by a WSD system, namely, document splitting, tokenization, lemmatization and part-of-speech tagging (see Section 2.1). This means that AMuSE-WSD takes just raw text as input, freeing the user from the burden of having to compose complex preprocessing pipelines. Furthermore, our system is optimized to speed up inference, especially on CPU, which makes AMuSE-WSD accessible to a broader audience with smaller hardware budgets. Finally, the API is available both as a Web API, so as to obtain sense annotations without installing any software, and offline API, so that a user can host AMuSE-WSD locally and minimize inference times. In the following, we provide more details on the main set of functionalities offered by AMuSE-WSD.

Document-level preprocessing. The AMuSE-WSD API lets the user obtain disambiguated text for batches of sentences in a single query to the service, reducing the load on the network and the latency of the responses. On top of this, one common use case for end users is the disambiguation of long texts. In order to assist users in such cases, AMuSE-WSD supports the disambiguation of documents of arbitrary length, performing document splitting to enable faster inference by transparently batching text segments.

Sentence-level preprocessing. The only piece of information the end user needs to provide to AMuSE-WSD is raw text. Nonetheless, the underlying model needs lemmas and part-of-speech tags in order to select the proper sense for a word (see Section 2.1). To this end, AMuSE-WSD integrates a preprocessing pipeline, transparent to the user, that performs tokenization, lemmatization and part-of-speech tagging. In the search for the optimal compromise between latency and accuracy of the preprocessing, AMuSE-WSD employs two different preprocessing pipelines depending on the language of the input document, built around spaCy and Stanza. For each of the 40 supported languages, our system selects one of the preprocessing models available in either spaCy or Stanza, prioritizing the former for its high inference speed and falling back to Stanza for lower-resource languages where the performance of the former is suboptimal.

Usage. The AMuSE-WSD API exposes an endpoint named `/api/model`. The endpoint accepts `POST` requests with a `JSON` body, containing a list of documents. For each document, two parameters must be specified:

- `text`: the text of the document.
- `lang`: the language of the document.

Each request returns a `JSON` response containing a list of objects, one for each input document. Each object in the response provides the tokenization, lemmatization, PoS-tagging and sense information of the corresponding input document. The AMuSE-WSD API returns labels that are easily usable for users interested in WordNet, BabelNet and the NLTK WordNet API, making it simple to switch from one resource/tool to another. For example, the response to a request that contains only one sentence will have the following structure:

```
[{
    "tokens": [
        ...
        {
            "index": 5,
            "text": "river",
            "pos": "NOUN",
            "lemma": "river",
            "bnSynsetId": "bn:00067948n",
            "wnSynsetOffset": "9411430n",
            "nltkSynset": "river.n.01"
        },
        ...
    ]
}]
```

We refer to the online documentation for more details, including the full list of supported languages.[2]

3.1 Offline API

To further promote the integration of WSD into large-scale applications, AMuSE-WSD is also distributed as a Docker[3] image that can be deployed locally on the user's own hardware. We provide several ready-to-use images suitable for different configurations and use cases, depending on whether a user is constrained by hardware requirements or prefers the highest quality over shorter inference times. The images are differentiated based on their size in parameters and on the hardware they are optimized for, that is, CPU or GPU. In the following, we provide more details on the Docker images.

Model configurations. With AMuSE-WSD, an end user can choose between four different types of pre-built Docker images which differ in the pretrained language model used to compute contextualized word representations:

- `amuse-large` uses BERT-large and provides state-of-the-art results in English WSD.
- `amuse-large-multilingual` employs XLM-RoBERTa-large and thus offers the best results in multilingual WSD. However, it is also the most demanding in terms of hardware requirements.
- `amuse-medium-multilingual` adopts XLM-RoBERTa-base which provides outputs that in 98% of cases are the same as those of its larger counterpart, but taking half the time.
- `amuse-small-multilingual` uses the multilingual version of MiniLM, a language model distilled from XLM-RoBERTa-base. It is three times faster and three times smaller, while still achieving remarkable results.

Inference times. In order to further promote its accessibility, AMuSE-WSD is also available in Docker images optimized to run on CPU thanks to ONNX. The ONNX Runtime[4] is an engine built with the aim of significantly improving inference times while, at the same time, reducing system footprint. With ONNX, our system is up to 2.5 times faster on CPU while also being 1.2 times smaller compared to its vanilla PyTorch implementation. We evaluate the inference speed of AMuSE-WSD on two sets of documents: the concatenation of all the sentences of the unified evaluation framework for WSD by Raganato et al. (2017) and a set of 300 documents whose lengths range from 1500 to 3500 characters. Table 2 compares the inference times of our system in several configurations, showing the gains of ONNX on CPU.

AMuSE-WSD			ALL		Long	
Configurations			Δ (ms)	↑	Δ (ms)	↑
English	large	PLAIN	163	1.0×	1610	1.0×
	large	ONNX	93	1.7×	1230	1.3×
	large	CUDA	12	13.6×	170	9.5×
Multilingual	large	PLAIN	190	1.0×	2940	1.0×
	large	ONNX	128	1.5×	2170	1.3×
	large	CUDA	13	14.6×	180	16.3×
	medium	PLAIN	136	1.0×	1050	1.0×
	medium	ONNX	61	2.2×	880	1.2×
	medium	CUDA	11	12.4×	150	7.0×
	small	PLAIN	96	1.0×	600	1.0×
	small	ONNX	39	2.5×	460	1.3×
	small	CUDA	9	10.7×	90	6.7×

Table 2: Inference time for several configurations over two sets of documents: the concatenation of all the test sets of the unified evaluation framework for WSD (**ALL**), and a set of 300 documents whose lengths are between 1500 and 3500 characters (**Long**). **Δ**: latency or median time in milliseconds AMuSE-WSD takes to serve one inference request. ↑: Inference speed-up provided by the ONNX and CUDA implementations over vanilla PyTorch.

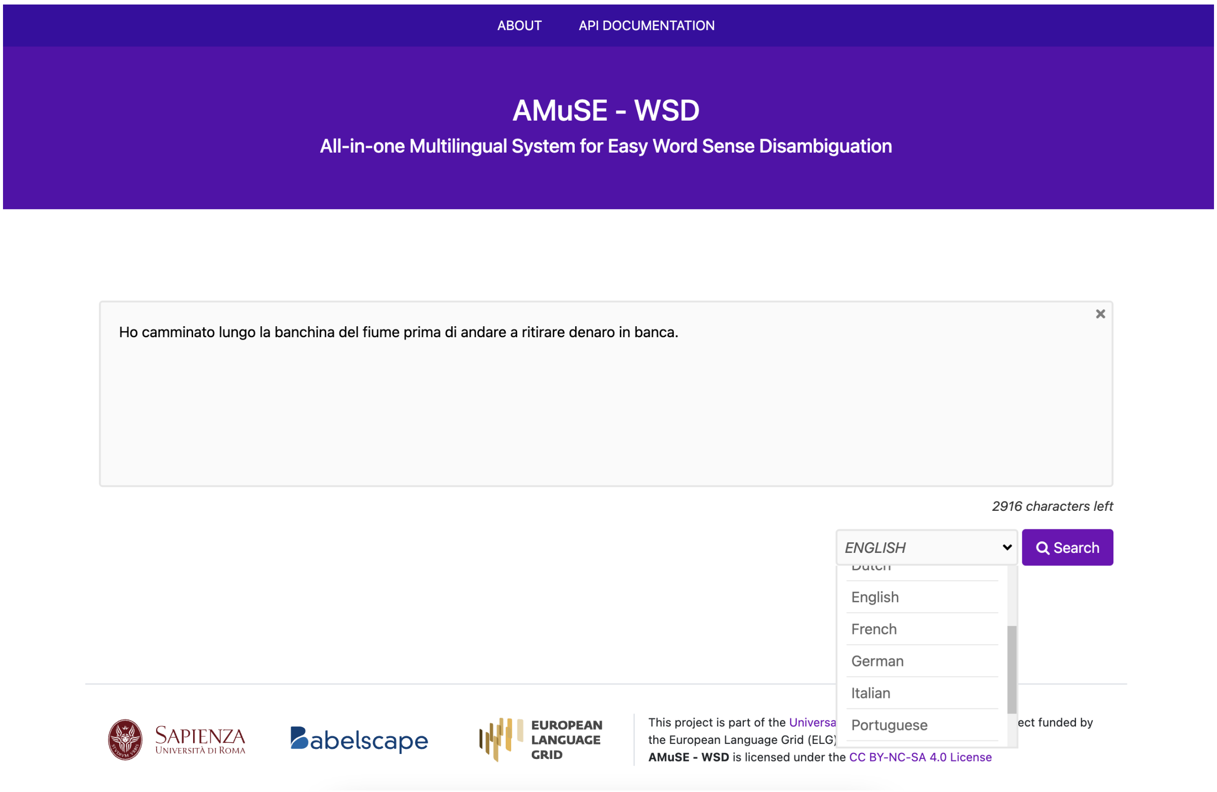
4 Web Interface

AMuSE-WSD comes with a Web interface which allows users to disambiguate text on the fly without having to write a single line of code. Figure 1 shows the home page of AMuSE-WSD in which a user can type a short text (up to 3000 characters) and indicate the language of the inserted text. Figure 2 shows, instead, how the Web interface displays the senses for the input sentence *"the quick brown fox jumps over the lazy dog"*, while Figures 3 and 4 show a correctly disambiguated word, that is, *bank*, which is ambiguous both in English and Italian. For each content word (nouns, adjectives, adverbs and verbs), the interface shows the corresponding sense predicted by the underlying WSD model, including its definition from WordNet and an image, if available. Each meaning is linked

[2] http://nlp.uniroma1.it/amuse-wsd/api-documentation

[3] https://www.docker.com

[4] https://www.onnxruntime.ai



Ho camminato lungo la banchina del fiume prima di andare a ritirare denaro in banca.

2916 characters left

This project is part of the Universa… …ect funded by the European Language Grid (ELG)
AMuSE - WSD is licensed under the CC BY-NC-SA 4.0 License

Figure 1: The home page of the Web interface of AMuSE-WSD. Users can write (or paste) text in the text area. The dropdown menu allows users to select the language of the input text from among the 10 available in the online interface.

to its corresponding Web page in BabelNet 5,[5] a multilingual encyclopedic dictionary that includes WordNet, Open Multilingual WordNet, Wikipedia, Wikidata and Wiktionary, *inter alia*. Users can click on a meaning to obtain further information about it, ranging from other definitions to its translation in other languages, from its hypernyms (generalizations) to its hyponyms (specializations).

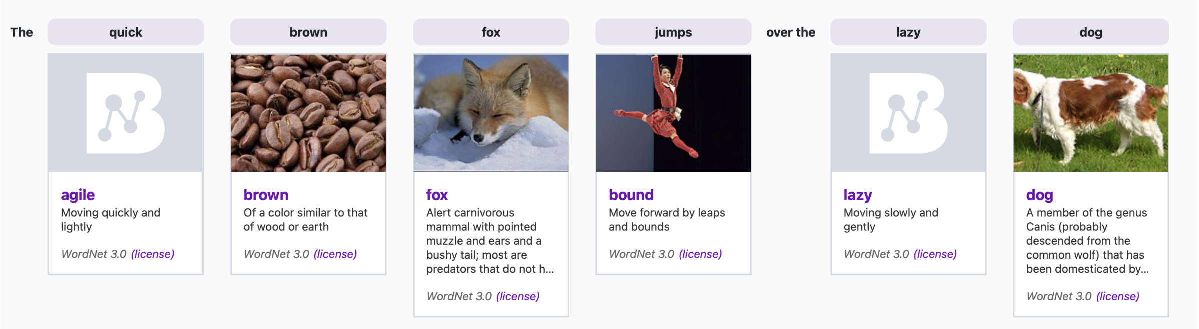
We believe that this interface will be especially useful for researchers, teachers, students and curious people who may be interested in understanding how WSD can be taken advantage of in other fields. Moreover, we also believe that an easy-to-use interface will attract the attention of more researchers on the task of WSD itself, encouraging future developments in this area.

5 Conclusion

Over the past few years, WSD has witnessed an increasing rate of development, especially thanks to the renewed interest in neural networks and the advent of modern pretrained language models. However, research implementations are often far from being ready-to-use as they either take as input pre-parsed documents or require setting up a preprocessing pipeline to take care, at least, of document splitting, tokenization, lemmatization and PoS tagging. Unfortunately, currently available ready-to-use WSD systems now lag behind the state of the art and offer solutions based on non-neural approaches.

In this paper, we addressed this issue and presented AMuSE-WSD, an All-in-one Multilingual System for Easy Word Sense Disambiguation. To the best of our knowledge, AMuSE-WSD is the first system for end-to-end WSD to encapsulate a state-of-the-art neural model for sense disambiguation in multiple languages. Our system makes it easy to obtain and use word-level semantic information about word meanings through a RESTful API. This API is available both online, that is, a user can disambiguate text through HTTP requests to our server, and offline, that is, a user can download prepackaged Docker images and run them locally to minimize inference times on large bulks of documents. We provide different configurations to satisfy different needs, from images that are optimized to run on constrained hardware to high-performing

[5] https://babelnet.org


The quick brown fox jumps over the lazy dog

agile Moving quickly and lightly
WordNet 3.0 (license)

brown Of a color similar to that of wood or earth
WordNet 3.0 (license)

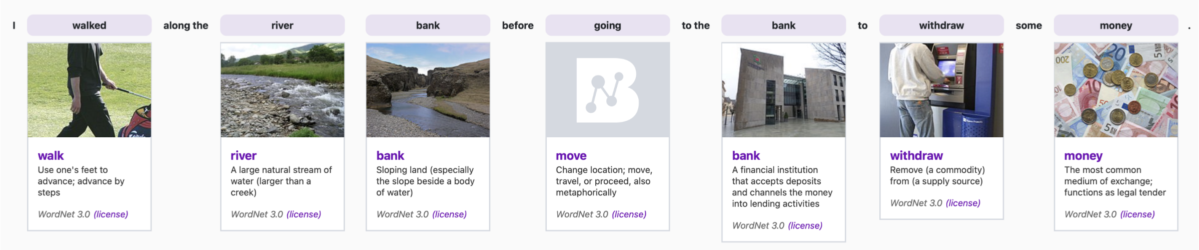
fox Alert carnivorous mammal with pointed muzzle and ears and a bushy tail; most are predators that do not h...
WordNet 3.0 (license)

bound Move forward by leaps and bounds
WordNet 3.0 (license)

lazy Moving slowly and gently
WordNet 3.0 (license)

dog A member of the genus Canis (probably descended from the common wolf) that has been domesticated by...
WordNet 3.0 (license)

Figure 2: Overview of the Web interface of AMuSE-WSD. The WSD model tags each content word with a WordNet sense, its definition and an image, if available. Each sense is linked to BabelNet which provides more information about senses, from their translation into other languages to their related meanings.


I walked along the river bank before going to the bank to withdraw some money .

walk Use one's feet to advance; advance by steps
WordNet 3.0 (license)

river A large natural stream of water (larger than a creek)
WordNet 3.0 (license)

bank Sloping land (especially the slope beside a body of water)
WordNet 3.0 (license)

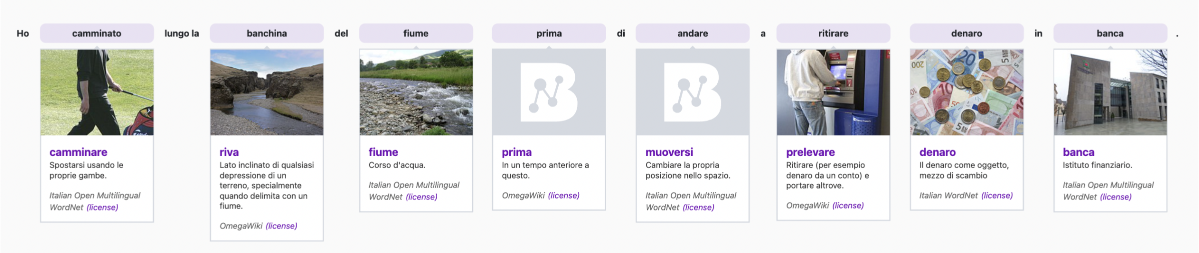
move Change location; move, travel, or proceed, also metaphorically
WordNet 3.0 (license)

bank A financial institution that accepts deposits and channels the money into lending activities
WordNet 3.0 (license)

withdraw Remove (a commodity) from (a supply source)
WordNet 3.0 (license)

money The most common medium of exchange; functions as legal tender
WordNet 3.0 (license)

Figure 3: An example in which AMuSE-WSD correctly distinguishes between two different senses of the word *bank* (its geographical and financial senses).


Ho camminato lungo la banchina del fiume prima di andare a ritirare denaro in banca .

camminare Spostarsi usando le proprie gambe.
Italian Open Multilingual WordNet (license)

riva Lato inclinato di qualsiasi depressione di un terreno, specialmente quando delimita con un fiume.
OmegaWiki (license)

fiume Corso d'acqua.
Italian Open Multilingual WordNet (license)

prima In un tempo anteriore a questo.
OmegaWiki (license)

muoversi Cambiare la propria posizione nello spazio.
Italian Open Multilingual WordNet (license)

prelevare Ritirare (per esempio denaro da un conto) e portare altrove.
OmegaWiki (license)

denaro Il denaro come oggetto, mezzo di scambio
Italian WordNet (license)

banca Istituto finanziario.
Italian Open Multilingual WordNet (license)

Figure 4: The output of AMuSE-WSD for the Italian translation of the sentence shown in Figure 3.

images. Last but not least, AMuSE-WSD comes with an intuitive Web interface that lets users disambiguate short documents on the fly without writing a single line of code. Not only does this interface showcase the capabilities of our system, but we also hope it will attract the interest of new researchers to the field of lexical semantics.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Consolidator Grant MOUSSE No. 726487 and the European Language Grid project No. 825627 (Universal Semantic Annotator, USeA) under the European Union's Horizon 2020 research and innovation programme.

References

Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2014. Random walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics*, 40(1):57–84.

Eneko Agirre and Aitor Soroa. 2007. SemEval-2007 task 02: Evaluating Word Sense Induction and Discrimination Systems. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 7–12, Prague, Czech Republic.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2021. ConSeC: Word Sense Disambiguation as Continuous Sense Comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in Word Sense Disambiguation by incorporating knowledge graph information. In *Pro-*

ceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2854–2864, Online.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent trends in Word Sense Disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4330–4338. Survey Track.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of Word Sense Disambiguation with gloss informed bi-encoders. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1006–1017, Online.

Rexhina Blloshmi, Tommaso Pasini, Niccolò Campolungo, Somnath Banarjee, Roberto Navigli, and Gabriella Pasi. 2021. IR like a SIR: Sense-enhanced Information Retrieval for Multiple Languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic.

Devendra Singh Chaplot and Ruslan Salakhutdinov. 2018. Knowledge-Based Word Sense Disambiguation using topic models. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5062–5069.

Simone Conia and Roberto Navigli. 2020. Conception: Multilingually-enhanced, human-readable concept vector representations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3268–3284, Barcelona, Spain (Online).

Simone Conia and Roberto Navigli. 2021. Framing Word Sense Disambiguation as a multi-label problem for model-agnostic knowledge integration. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3269–3275, Online.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online.

Claudio Delli Bovi, Luca Telesca, and Roberto Navigli. 2015. Large-scale Information Extraction from textual definitions through deep syntactic and semantic analysis. *Transactions of the Association for Computational Linguistics*, 3:529–543.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota.

Philip Edmonds and Scott Cotton. 2001. SENSEVAL-2: Overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5, Toulouse, France.

Ahmed ElSheikh, Michele Bevilacqua, and Roberto Navigli. 2021. Integrating personalized PageRank into neural Word Sense Disambiguation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic.

Katrin Erk and Diana McCarthy. 2009. Graded Word Sense Assignment. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 440–449, Singapore.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Frederick Liu, Han Lu, and Graham Neubig. 2018. Handling homographs in Neural Machine Translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1336–1345, New Orleans, Louisiana.

Rada Mihalcea, Timothy Chklovski, and Adam Kilgarriff. 2004. The senseval-3 English lexical sample task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 25–28, Barcelona, Spain.

George A. Miller. 1992. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a semantic concordance for sense identification. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Andrea Moro and Roberto Navigli. 2015. SemEval-2015 task 13: Multilingual All-Words Sense Disambiguation and Entity Linking. In *Proceedings of the*

9th International Workshop on Semantic Evaluation (SemEval 2015), pages 288–297, Denver, Colorado.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: A unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Roberto Navigli. 2009. Word Sense Disambiguation: A survey. *ACM Comput. Surv.*, 41(2).

Roberto Navigli. 2018. Natural Language Understanding: Instructions for (present and future) use. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 5697–5702.

Roberto Navigli, Michele Bevilacqua, Simone Conia, Dario Montagnini, and Francesco Cecconi. 2021. Ten years of BabelNet: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4559–4567. Survey Track.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. SemEval-2013 task 12: Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, Georgia, USA.

Roberto Navigli and Simone Paolo Ponzetto. 2012a. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.*, 193:217–250.

Roberto Navigli and Simone Paolo Ponzetto. 2012b. Multilingual WSD with just a few lines of code: the BabelNet API. In *Proceedings of the ACL 2012 System Demonstrations*, pages 67–72, Jeju Island, Korea. Association for Computational Linguistics.

Simone Papandrea, Alessandro Raganato, and Claudio Delli Bovi. 2017. SupWSD: A flexible toolkit for supervised Word Sense Disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 103–108, Copenhagen, Denmark.

Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. XL-WSD: An extra-large and cross-lingual evaluation framework for Word Sense Disambiguation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13648–13656.

Xiao Pu, Nikolaos Pappas, James Henderson, and Andrei Popescu-Belis. 2018. Integrating weakly supervised Word Sense Disambiguation into Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 6:635–649.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word Sense Disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain.

Alessandro Raganato, Yves Scherrer, and Jörg Tiedemann. 2019. The MuCoW test suite at WMT 2019: Automatically harvested multilingual contrastive Word Sense Disambiguation test sets for machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 470–480, Florence, Italy.

Prajit Ramachandran, Barret Zoph, and Quoc V. Le. 2018. Searching for activation functions. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*.

Ganesh Ramakrishnan, Apurva Jadhav, Ashutosh Joshi, Soumen Chakrabarti, and Pushpak Bhattacharyya. 2003. Question Answering via Bayesian inference on lexical relations. In *Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering*, pages 1–10, Sapporo, Japan.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. With more contexts comes better performance: Contextualized Sense Embeddings for all-round Word Sense Disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3528–3539, Online.

Federico Scozzafava, Marco Maru, Fabrizio Brignone, Giovanni Torrisi, and Roberto Navigli. 2020. Personalized PageRank with syntagmatic information for multilingual Word Sense Disambiguation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–46, Online.

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2019. Text Categorization by learning predominant sense of words as auxiliary task. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1109–1119, Florence, Italy.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural*

Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pages 5998–6008.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Zhi Zhong and Hwee Tou Ng. 2012. Word Sense Disambiguation improves Information Retrieval. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 273–282, Jeju Island, Korea.