Universitatea Transilvania din Brașov

Universitatea Transilvania din Brașov
FACULTATEA DE INGINERIE ELECTRICĂ ȘI ȘTIINȚA CALCULATOARELOR

ȘCOALA DOCTORALĂ INTERDISCIPLINARĂ
FACULTATEA DE INGINERIE ELECTRICĂ ŞI ŞTIINȚA CALCULATOARELOR

ADRIAN M.P. BRAŞOVEANU, Msc.

Intelligent Systems in Semantic Networks

Sisteme Inteligente în Rețele Semantice

ABSTRACT/REZUMAT

Supervisor / Conducător ştiințific
Prof.dr.mat. RĂZVAN ANDONIE

BRAȘOV, 2021

D-lui(D-nei).....

COMPONENȚA

Comisiei de doctorat

Numită prin ordinul Rectorului Universității Transilvania din Braşov
Nr. ..... din .....

PREŞEDINTE:	- CONF. UNIV. DR. CARMEN GERICAN Universitatea Transilvania din Braşov
CONDUCĂTOR ŞTIINIFIC:	- PROF. UNIV. DR. MAT. RĂZVAN ANDONIE Universitatea Transilvania din Braşov
REFERENȚI:	- PROF. UNIV. DR. DIANA INKPEN Universitatea din Ottawa, Canada - PROF. UNIV. DR. GHEORGHE ȘTEFAN Universitatea Politehnică din Bucureşti - PROF. UNIV. DR. LIVIU P. DINU Universitatea din Bucureşti

Data, ora şi locul susținerii publice a tezei de doctorat: ....., ora ....., sala .....

Eventualele aprecieri sau observații asupra conținutului lucrării vor fi transmise electronic, în timp util, pe adresa brasoveanu.adrian@unitbv.ro.

Totodată, vă invităm să luați parte la şedința publică de susținere a tezei de doctorat.

Vă mulțumim.

CONTENT

CUPRINS (lb. română)

Chapter 1

INTRODUCTION

1.1 Background

Intelligent systems have long been a proxy for Artificial Intelligence (AI). AI is typically defined as the pursuit of human-like or super-human intelligence in machines. The term Machine Learning (ML) encompasses one of the various AI schools that came into prominence in the 1980s and later came to dominate the field. Its main idea is that machines can learn from the data they are fed. Other schools of AI also exists, among them the Semantic Web (SW), also known as Knowledge Graphs (KGs). Knowledge Graphs (KG) are implemented using a triplet format (e.g., subject-predicate- object) which enables the use of expressive query languages for finding facts, as well as automated reasoners to infer new data. Natural Language Processing's (NLP) end goal is to decode human languages in their various formats. In this thesis, we view NLP as the link between the two branches of AIs discussed in this thesis: ML and SW.

1.2 Main Contributions

Semantic AI is a recent approach towards AI that is focused on combining semantics with classic AI methods like classification, clustering or recommendation. By adding semantics, we can increase data quality while simultaneously removing black-box approaches. The core proposition of SAI is that regardless of its original provenance (e.g., text, table, picture), data can be processed and stored into refined formats like those provided by KGs or search engines. These open data clusters can later be used to solve complex problems with hybrid approaches. By combining entities extracted from a KG with sentiment and ML classifiers, it is possible to verify the claims from a sentence, for example. This thesis examines several hybrid methods enabled by SAI to understand how to leverage them to build baselines for research and production. It then asks what we can do to improve these hybrid methods, as it seems that each component may add its errors to the stack and confuse the researchers and developers.

The results presented here can be classified into the following research directions.

1.2.1 SAI for Knowledge Extraction

The contributions in this area are mostly related to the development of SAI systems, as well as to the role small improvements in corpora (e.g., adding entities or relations) can play in achieving better results.

- The first contribution is related to the role of name variance for NEL through i) improving the coverage of NEL tools; and ii) usage of lenses to evaluate different surface forms for the same entities.
- The second contribution targets the creation of domain-specific sentiment engines.
- The third contribution is dedicated to fact verification, as it shows how the earlier techniques can be integrated to detect fake news.

1.2.2 Interpreting and Explaining SAI

Debugging semantic AI systems is extremely difficult for programmers today because it is not always clear which components generate the errors. The contributions from this area address this shortcoming:

- The first contribution from this chapter is a general methodology for developing explainable benchmarking systems for Semantic AI. One of the first ideas in this area was to propose a taxonomy of Named Entity Linking errors.
- The second contribution is the natural extension of the previous one, as the development of the error analysis tool eventually led to a software framework called Orbis which is now used to benchmark and explain results from multiple fields (e.g., Named Entity Linking, Slot Filling, Forum Extraction, etc).
- The last contribution in this area is mostly theoretical, as it is a compressed survey related to the role of visualization in interpreting and explaining SAI.

1.3 Origins

Chapter 1 covers the introduction, and therefore is not dedicated to any publications.

Chapter 2 describes the basic architecture of SAI systems. The presented use case is a system created for displaying tourism indicators which was published in two journal articles in *Semantic Web* (IF=3.524) [BSS+17], and *Journal of Information Technology and Tourism* (IF=2.95) [SOBS16], and a conference article at ENTER 2015 [SBÖ15]. The last section of the chapter

provides a high-level view of the SAI architecture and represents an introduction to the next chapters.

Chapter 3 presents three contributions related to the building of SAI sytems.

Chapter 3.1 discusses contributions to Name Entity Linking, namely lenses and name variants. An entity can have multiple names, an issue we call name variance, and therefore it is important for a NEL system to be able to extract all these names. One possibility is to extract the additional names from multiple KGs. Another possibility is to create algorithms that compute these name variants algorithmically. These ideas were originally published in a conference publication at WIMS 2018 [WKB18] and LDK 2019 [WBKN19]. A method through which the NEL scores can be computed when we consider name variance is also given. The method involves the creation of annotation sets focused on a single property (e.g., mention, type, link) and it shows how this mechanism improves results for some NEL systems. The main findings discussed in this section are based on two conference publication from an ACL conference, CoNLL 2020 [BWN20] and ACM WIMS 2018 [BNW18].

Chapter 3.2 presents a contribution related to the construction of domain-specific affective classifiers. The main idea is to use KGs, embeddings, and pre-trained language models to improve the categorization of emotions. The findings were published in a journal article from *Cognitive Computation* (IF=4.307) [WSB$^+$21].

Chapter 3.3 is dedicated to fact verification. A version of fact checking that is often called fake news is examined. A basic NLP pipeline that includes entities, sentiment and relations is used to assess the degrees of truth associated to a series of corpora based on Politifact. These findings were initially published in a conference article in IWANN 2019 [BA19] and in a journal article from *Neural Processing Letters* (IF=2.891) [BA20a].

Chapter 4 is dedicated to interpretability and explainability. These are perhaps the most important topics today in light of the unexpected success of ML algorithms during the last decade. If we are to develop networks that will diagnose patients or judge legal cases (or even provide help for these tasks), then it is necessary to clearly explain their reasoning.

Chapter 4.1 is focused on explainable benchmarking. Initially a taxonomy was developed to help clarify which NEL components trigger certain types of errors. Later this taxonomy was transformed into a tool that also helps visualize the various errors. The taxonomy was presented at LREC 2018 [BRK$^+$18], Orbis was introduced at SEMANTICS 2018 [OKBW18]. A related article about improving gold standards was published at RANLP 2019 [WBKN19].

Chapter 4.2 discusses the role of visualization in explaining AI systems. A survey is conducted to identify the trends in the field. The section builds upon an IEEE conference publication from IV2020 [BA20b].

Chapter 5 formulates the conclusions and therefore cites some of these publications again.

1.4 Structure

The thesis is structured around the two main contributions.

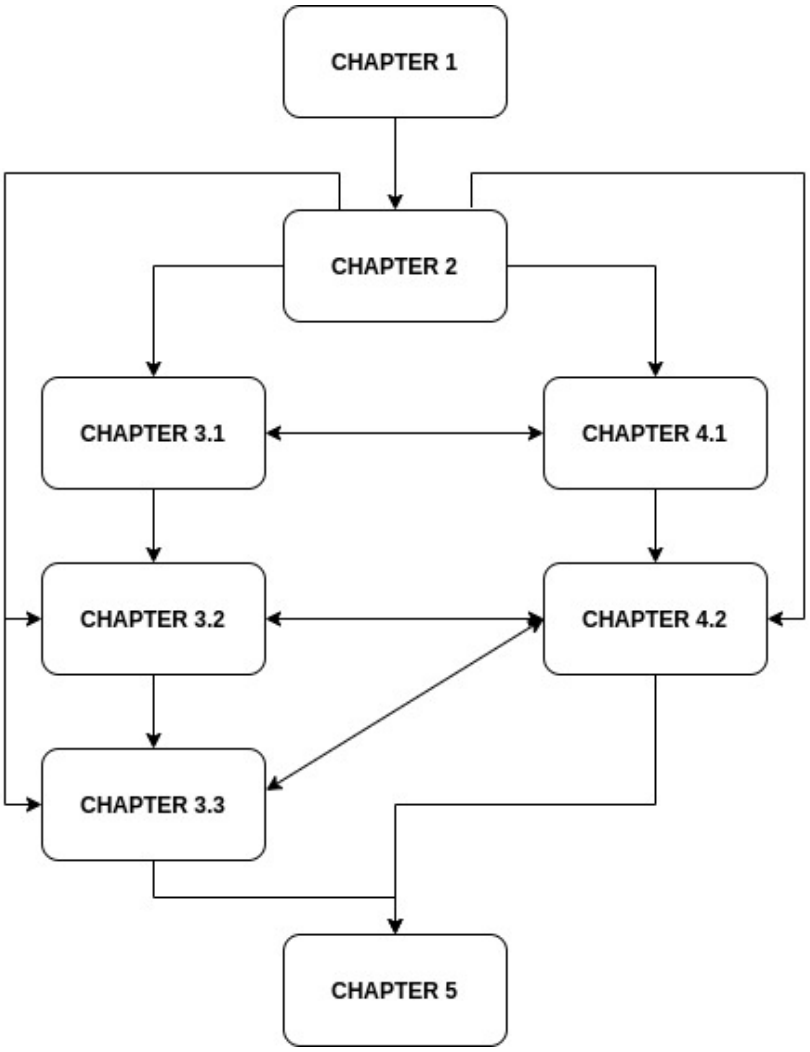

Figure 1.1: The structure of the thesis.

Figure 1.1 showcases the links between the various chapters or sections. As it can be seen, Chapter 2 presents the fundamentals and links to most of the sections. The rest of the chapters build upon the previous chapters or sections. In some cases links between sections from different chapters can also be observed (e.g., chapters 3.1 and 4.1 discuss NEL from two different points of view).

Chapter 2

FUNDAMENTALS OF SEMANTIC AI

2.1 A Brief History of Knowledge Graphs

There are many definitions of a KG. Most of them are related to the Knowledge Representation (KR) aspect of the graph (e.g., property graph, directed graph, etc). Some are also interested in the logical formalism behind it. For our purposes, we prefer a simple definition that is based on Aidan Hogan's definition [HBC+20]:

Definition. *A Knowledge Graph (KG) is a graph that contains a limited representation of the real world and whose nodes and edges represent real-life entities and the relations between them.*

KGs can support query languages, KR constructs (e.g., ontologies or rules), and even multiple serializations (e.g., RDF, JSON).

KG query languages need to be more expressive than SQL, as they have to enable relational operators (e.g., joins, unions), as well as recursive operators that may include path expressions (e.g., expressions that can match paths between nodes). SPARQL is the most widely used KG query language.

Ontologies, a standard formalism in KR, represent entities and relations from a domain. They contain statements that define the domain (TBox), as well as some examples or instances of the defined classes (ABox). Ontologies are defined using logical expressions and they often enable reasoning upon the collected data. The most widely used format for building ontologies today is OWL. Reasoners often use a subset of OWL like OWL DL to express their logical constructs.

Multiple serialization formats were needed because RDF, the original metadata model for the SW written in XML, was difficult to use. Early formalization focused on the triple construct (e.g., N3, N-Triples), whereas more recent serializations also support JSON (e.g., JSON-LD). Some KGs may also be created without such constructs (e.g., based on graph databases), but they will also use a limited formalism.

Having access to KGs that support these constructs is usually the first step towards developing semantic applications.

A classic application is a platform for accessing and visualizing KGs. Such

platforms are generally called Linked Data Platforms (LDP)[1] and tend to respect the Linked Data publishing principles described by Sir Timothy Berners-Lee and several of his collaborators. The main idea behind these set of practices was to publish datasets online using unique identifiers for entities. This allowed these datasets to be queried through languages like SPARQL, but also to be linked to other datasets, therefore expanding the Linked Open Data (LOD) graph. KGs like DBpedia and Wikidata are central hubs in the LOD graph.

More complex applications can also be imagined, and they include elements of Natural Language Processing and Machine Learning. Such systems will generally use a suite of Knowledge Extraction (IE) tools or will themselves be IE tools. One of the main requirements for them will be to extract data (e.g., text, entities, sentiment) from the web or documents (e.g., PDF files) and display it in a format that is easy to interpret by humans.

2.2 Building Semantic Applications without AI

2.2.1 Ontology-Based Data Access

Knowledge Graphs can also be graph representations of classic databases. In some cases, databases can be accessed automatically in a virtual KG format through Ontology-Based Data Access (OBDA)[CCK+17] tools. The virtual KG can be created because the mappings will specify the correspondence between the database entities and the ontological concepts. Some OBDA tools can also materialize the mapped KGs, which means they can create dumps with all the KG's triples. Ontop[CCK+17] is one of the best OBDA tools.

The TourMIS [Wöb03] database was transformed into a KG using the OBDA methodology and Ontop. The construction process is presented in [SOBS16].

2.2.2 Linked Data and Dashboards

One of the most important applications of the ETIQH KG was the construction of a dashboard for showcasing statistical linked indicators. This dashboard was built using the webLyzard Platform[2] which integrates data services and visualizations. The construction of the dashboard is described in [BSS+17].

A screenshot of the resulting dashboard is presented in Figure 2.1. As it can be seen from the screenshot, the line chart allowed us to quickly draw conclusions related to the displayed data. It can easily be seen that there is some kind of seasonality to the data, with clear peaks during Summer and lows during Winter. This was because none of the visualized capitals was a winter destination.

As it can be seen the ETIHQ KG and the associated dashboards included semantic information, however, there is only so much that can be found simply by looking at statistical charts. For example, what caused the peaks? Why do

[1] https://www.w3.org/TR/ldp/

[2] https://www.weblyzard.com/

these peaks differ between the countries or years? The data itself will not offer us enough clues, except if these additional details are surfaced through many searches.

To obtain better answers it is important to introduce new types of data. We might, for example, parse news media articles published during the examined period. This will require more complex architectures like the ones described in the next pages.

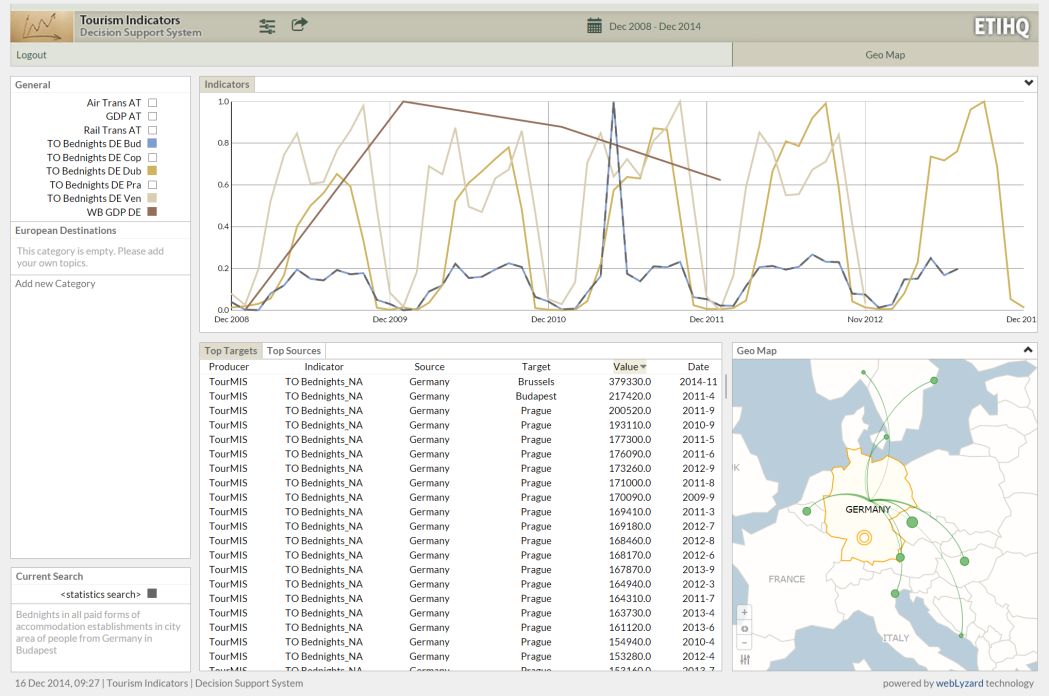

Figure 2.1: The interface of the ETIHQ dashboard. Reproduced from [BSS+17].

2.3 The Architecture of Semantic AI

Figure 2.2 presents the basic elements of a SAI architecture. These include: (i) documents expressed in natural language; (ii) NLP pipelines; (iii) Knowledge Graphs; and (iv) applications.

2.3.1 Natural Language Documents

Regardless of its provenance (e.g., news media articles, social media, forums or audio/video), most of the information available online can be turned into natural text (NL). To access it, we need to clean it and crawl it.

2.3.2 Natural Language Processing

The main task of NLP has always been to extract meaningful information from natural language (NL) documents. Sometimes this meant generating translations, and sometimes adding annotations to identify various entities or events. Dedicated tools exist for NLP tasks, but many of these tools can be accessed via the command line interfaces (CLI) or REST APIs and chained together to form NLP pipelines. This is needed, as often these tasks depend on one another.

2.3.3 Knowledge Graphs

Depending on the system we want to build, there may be a need for one or more KGs. If the system simply needs links to the entities, then it may be enough to simply provide these links. If the tasks that need to be solved are more complicated and also involve the finding of named entities or the filling of missing information in KGs (e.g., Slot Filling), then we may need to provide infrastructure for performing these additional operations.

2.3.4 Applications

Some complex applications are built around sentiment analysis or fact checking. Most of the visualizations or dashboards, as well as LDPs will sit on this layer.

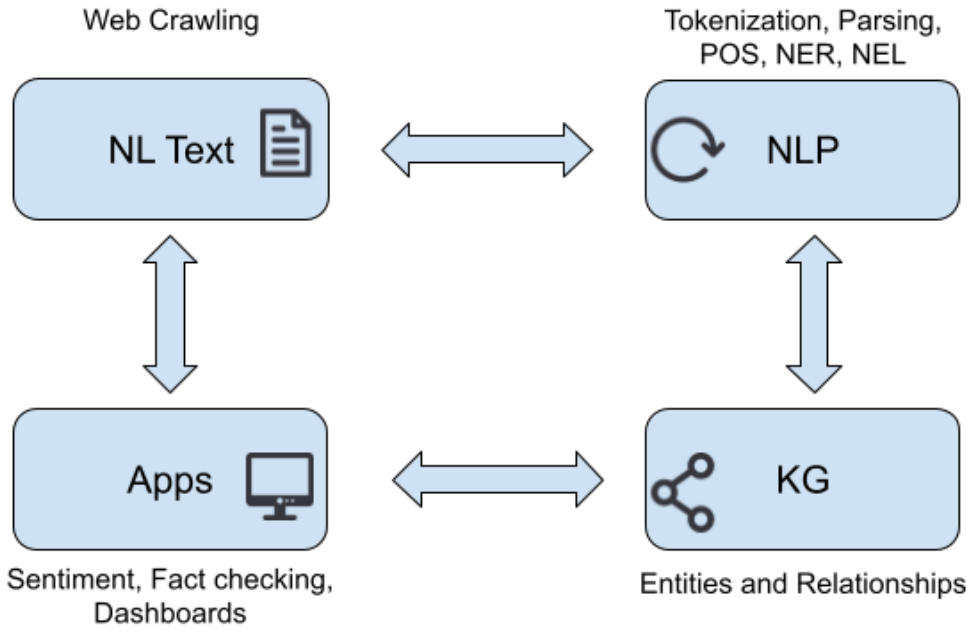

Figure 2.2: The general architecture of the SAI systems.

While this chapter offered a short history of the most important discoveries

for the advancement of KGs, it has not done the same thing for ML. This is simply because some aspects of ML are discussed in detail in the next chapters in the brief background sections for each of the discussed issues. Several other texts can be consulted to add to this information. A survey related to the DL applications for scientific discovery can be found in [RS20]. Closer to the topics approached in this thesis, surveys about the applications of ML and DL for NLP can be found in [YHPC18] and [OMK21].

2.4 Language Models and Semantic AI

The KGs are not the only option for decoding meaning. A serious argument can also be made for the modern language models (LMs) like Transformers. The main idea implemented by the Transformer architecture was that it is enough to simply focus on attention to arrive at good results [VSP+17]. A set of multiple attention heads seems to be all that is needed to better than average performance for NLP tasks. This mechanism provides the model with the possibility to interpret a variety of representation subspaces at various positions. This means that multiple weight matrices can be processed to encode different words or phrases from a document. The outputs are fed into pairs of encoders and decoders, depending on the model. The encoders and decoders may be used for different tasks (e.g., encoders for extracting named entities, and decoders for classifying emotions [RS20]).

Since such LMs can offer good performance for tasks like dependency parsing (DP), extraction of named entities or Question Answering, it is normal to ask if such language models should be considered Semantic AI? The short answer is yes, some of them can be considered SAI. The long answer is a bit more nuanced. If a language model can create its representation of the world, then it will be a SAI. If it cannot do this, then it will not be a SAI. Where should we draw the line? Intelligent systems that can save the learned representations in some kind of accessible format should be considered SAI, from our point of view. This may look like an arbitrary distinction. That is simply because complex representations take time to build, and there is a need to build upon previous layers. There is a need for continuity sometimes. This is why LMs will not fully replace KGs, but rather store their internal KGs or use shared KGs.

Chapter 3

SEMANTIC AI FOR KNOWLEDGE EXTRACTION

This chapter discusses three hybrid applications that use Semantic AI.

3.1 Entities and Knowledge Graphs

3.1.1 Background

The Named Entity Linking (NEL) task introduced in 2006 by Razvan Bunescu and Marius Pasca [BP06] added the requirement to link the entities to a KG like DBpedia [LIJ+15]. Good NEL systems are known to be created through several classes of algorithms: (i) KG disambiguation through community detection algorithms like Louvain [BGLL08] (e.g., Babelfy [MRN14a] and Recognyze [WKB18]); (ii) statistical language models (e.g., DBpedia Spotlight [DJHM13]); or (iii) neural models (e.g., the models proposed by Adel [AS19]).

The rest of the section describes custom algorithms based on the Recognyze architecture, as well as some issues related to their evaluation. Recognyze was jointly developed by researchers from Modul Technology, Swiss University of Applied Sciences of the Grisons and webLyzard. The core team included Albert Weichselbraun, Philipp Kuntschik and Adrian Braşoveanu. More details about this architecture can be found in [WKB18].

3.1.2 Named Entities and Their Variance

There are several possible IE tasks in which entities play a role. They can also be called knowledge extraction (KE) tasks and can be defined based on the treatment of mentions (e.g., the string extracted from the text) or links [WBKN19]:

- NER - the annotator needs to provide the entity's surface form, position and type;

- NEL - besides the fields required by NER, the annotator needs to provide the link to a target KG;
- SF - the annotor needs to provide missing properties for a given entity;
- (O)KE - the annotator needs to extract all entities and relations available in the text.

Figure 3.1 showcases the links between these tasks. A DBpedia text from Edward Thorp's page is presented together with the included entities. All NEL systems will have to collect entity mentions and their links to a target KG entry. If they also provide details about each entity, they may also be able to compete in slot filling challenges. A NEL system is generally called an *automated annotator*, therefore we will often use these two terms (system and annotator) interchangeably. To disambiguate between human and machine (automated) annotators, we will use the type (e.g., human or automated). If no type is assigned, than the annotator is presumed to be a machine.

Definition. *A NEL annotator links a mention* $m_{[s_i]}^{e_i,KG}$ *or* $m_{[x_i,y_i]}^{e_i,KG}$ *of an entity's surface form* s_i *from document* d *to the target entity* e_i *from a KG. The pair* (x_i, y_i) *represents the mention's position within the given document*.

A mention is generally the string that contains the entity's name, often referred as surface form.

Gold standard annotations created by humans need to respect annotation guidelines. Such guidelines often follow rules expressed in previously published guidelines like those from CoNLL 2003 [SM03]. Since some variation is expected across languages or annotators, a judge is required to settle conflicts.

The issue of nested entities (e.g., NY Knicks may be linked to both New York and the NY Knicks team), as well as that of the name variance (e.g., the different names under which an entity may be present in a text) appear quite frequently in NEL. The main question we are concerned for the rest of this section is: What is the impact of name variance on the NEL systems?

Definition. *Name variance signifies the multitude of names, titles or abbreviations assigned to a single entity within a text document.*

It is a key problem in NEL, as to make sure that our systems are competitive enough, it would be important to disambiguate the different mentions that point to the same entity. Improving the handling of name variance is also equivalent with improving coverage and should also lead to improving recall. *Prince Charles*, for example, may be named as both *Prince of Wales* or *Duke of Edinburgh* today, the second title being inherited after his father's passing, but in the future he may as well be named *King Charles* if he inherits the throne.

The problem is compounded because different entity types may have additional variances. Person names can include titles, for example, whereas organization names can include country-specific prefixes or suffixes.

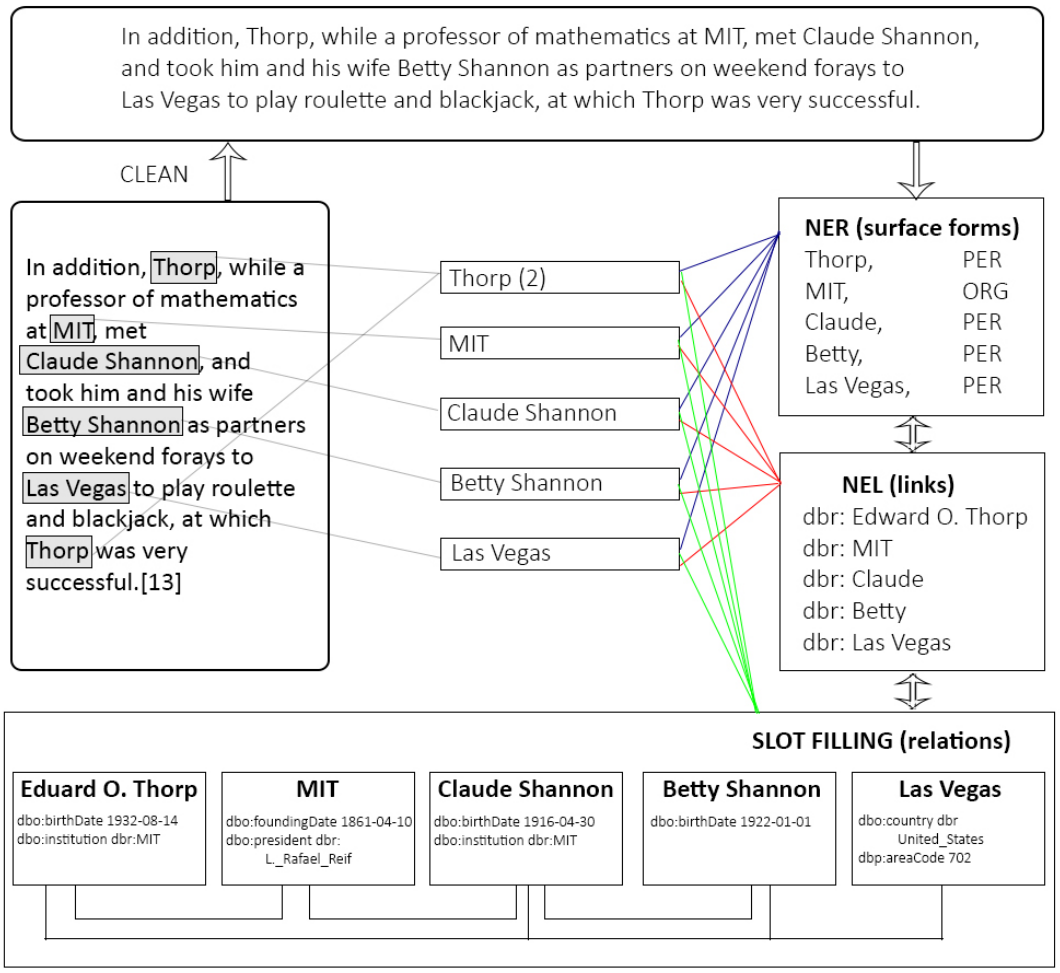

Figure 3.1: Relations between NER, NEL and SF.

3.1.3 Name Variance and NEL Coverage

The following pages are based on [WKB19] and discuss three methods that address this issue: (i) expanding the coverage of names by combining data from multiple sources (e.g., datasets or KGs); and (ii) through heuristics; or (iii) ML algorithms that compute these name variants automatically.

Perhaps the most natural idea was to simply collect name variances from other KGs and add them to the original lexicon. We tested both the impact of this strategy when collecting from a single KG, and from multiple KGs through SPARQL federation. This method is essentially equivalent with a lexicon attack on the name variance issue.

The second idea was to split the candidate names into substrings and then recombine them based on some simple rules. A simple strategy was to provide only the substrings of the original string. A more complex strategy involved:

- (i) replacing the candidate's tokens with synonyms;
- (ii) use heuristics which contained regular expressions to identify possible candidates (e.g., prefixes or suffixes for organization names) and modifying and replacing the corresponding tokens.

We can consider this method as the equivalent of a simple heuristic.

The third idea involves building name analyzers, an expansion of the algorithmic name generation based upon the idea of entropy. In Computer Science entropy represents the randomness collected by a signal or application.

The entropy score evaluates how many valid entity names can be computed from the available tokens. The tokens known to be included in entity names (e.g., suffixes or prefixes for organizations - like *Corp* or *GmbH*) were awarded higher entropy.

The entropy that corresponds to the name variance $\{t_i\}$ of an entity composed of *n* tokens $\{t_1, t_2, ...t_n\}$ can be computed with the formula [WKB19]:

$$H(\{t_i\}) = f_{\text{constr}}(\{t_i\}) \cdot \Big[H_{\text{case}}(\{t_i\}) + H_{\text{classes}}(\{t_i\}) + \sum_{t_j \in \{t_i\}} H_{\text{token}}(t_j) \Big] \quad (3.1)$$

Additionally, H_{case} removes case insensitivity, whereas the factor f_{constr} removes cases that lead to syntactic issues.

An alternative implementation of this method used the Java version of the libSVM[1] library. This method used a diverse set of features, including, but not limited to: (i) morphological (e.g., token case sensitivity, punctuation); (ii) syntactical (e.g., prepositions, pronouns); and (iii) semantic features (e.g., number of words that reference locations, first names or given names, common dictionary terms from multiple languages). The optimal results were obtained after cross-validation and grid-search for the radial basis function kernel (C=8, γ=2^{-5}).

We can consider this method as being equivalent with a brute-force attack, but its implementation can be performed in multiple ways as already explained.

Evaluations were performed on two datasets: N3 Reuters128 [RUH+14], known to be one of the most difficult NEL datasets [AOOV20], and OKE215 [NGP+15]. The best result was then benchmarked against three competing NEL engines: AIDA [HYB+11], Babelfy [MRN14a], and DBpedia Spotlight [DJHM13].

Table 3.1 showcases the results. We have used the classic metrics from IE evaluations: precision, recall and F1 and evaluated on the three entity types, as well as on all the classes.

The baseline (base) included no treatment of name variance. Adding RDF properties (case a) led to no improvements. Similarly, using either one of the KGs alone has not led to the expected improvements, which suggests that adding lots of names simply results in a lot of noise if it is not

[1] https://www.csie.ntu.edu.tw/ cjlin/libsvm/

Table 3.1: Name variance impact on performance. Dataset: Reuters128. Tool: Recognyze Lite. Bold indicates statistically significant scores. NG = name generation. Base = baseline.

Setting	LOC			ORG			PER			All		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
base	63	54	58	72	34	46	57	23	33	66	39	49
a) properties	63	54	58	71	33	45	57	23	33	66	38	49
b1) Wikidata	14	41	20	40	**41**	40	12	**38**	19	21	41	28
b2) Wikipedia	61	54	57	69	33	45	58	25	35	64	39	48
b3) GeoNames	60	54	57	71	33	45	57	23	33	64	38	48
b4) base+(b1,b2,b3)	14	41	21	39	**41**	40	12	**38**	19	21	41	28
c) algorithmic NG	54	**72**	62	35	**53**	42	68	**49**	**57**	43	**58**	50
d1) NG Wikidata	52	54	53	71	38	50	59	26	36	61	42	50
d2) NG Wikipedia	58	52	55	68	35	46	60	29	39	63	39	48
d3) NG GeoNames	48	53	51	70	33	45	57	23	33	58	38	46
d4) base+(d1,d2,d3)	46	53	50	70	38	50	61	30	40	58	42	49
e1) name analyzer (heuristic)	64	52	57	47	**44**	46	60	**56**	**58**	54	**48**	51
e2) name analyzer (SVM)	65	51	57	33	**47**	39	55	**47**	**50**	42	**48**	45
f) base+(a,c,d1,e1)	53	**70**	61	61	**52**	**57**	60	**56**	**58**	58	**58**	**58**

balanced by a method to compute meaningful names (e.g., name analyzers). Algorithmic name generation (case c) has led to improvements in recall, exactly as theorized, but precision dropped. However, applying the same algorithm to the additional KG entries (cases d1-d4) has not led to significant improvements. The name analyzers (cases e1-e2) improve recall and F1 for people (PER) and locations (LOC). Interestingly, improvements for organization names can be observed in the combined version (base+(a,c,d1,e1)), even though precision drops. Interestingly, when compared with the other systems, Recognyze obtained the best results in the respective evaluations, and has always maintained the edge when it comes to recall.

3.1.4 Name Variance and Lenses

Improving upon name variance is only half the story. The benchmarking tools may also not be ready for such improvements, and partially the obtained scores were somewhat lower than expected (though not by much) precisely due to this aspect. To improve this handling within the benchmarking tools, a solution for handling partial matches and nested entities better was needed. The rest of the subsection describes such a method published in [BWN20].

Table 3.2: Automated annotator performance. Datasets: Reuters 128 and OKE2015.

Corpus	System	LOC			ORG			PER			All		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Reuters 128	AIDA	44	64	52	76	29	42	50	49	50	53	43	47
	BabelNet	29	31	30	47	16	24	21	29	24	32	22	26
	Recognyze	**53**	**70**	**61**	61	52	57	60	56	58	58	58	58
	Spotlight	41	**70**	52	64	42	51	47	22	30	50	49	49
OKE 2015	AIDA	25	37	30	69	43	53	66	41	50	50	41	45
	BabelNet	21	35	26	67	40	50	55	14	22	40	26	32
	Recognyze	**62**	**73**	**67**	70	51	59	85	57	68	73	59	65
	Spotlight	50	72	59	81	50	62	56	11	18	61	36	45

The idea of lenses appeared due to a method used by photographers to create different types of pictures: changing lenses. Essentially, we can create annotations based on a certain specification. One lens could greedily select the longest string match for each entity, whereas another could select the shortest one, for example. Yet another one might do something in the middle, like considering entity overlaps. Special types of lenses can also be created by extracting subsets of annotations based on typing (e.g., location sets) or KG links (e.g., DBpedia or Wikidata lenses). This leads us to the definition of lenses:

Definition. *A lens is an annotation in which a single rule is used for the annotation of a specific property like type, length or link across the entire dataset.*

We can think of lenses as being either extreme simplifications of annotation guidelines (e.g., instead of creating ten rules for annotating long strings we use a single rule) or special cases (e.g., we decide to only evaluate types or Wikidata links). This means that by using multiple lenses we can simulate the effect of various design choices on the results.

An early attempt to showcase the concept of corpus with lenses is In Media Res[BWN20]. All the documents were collected from public sources (e.g., Wikipedia, Wikinews, etc.) and made publicly available through GitHub[2]. The corpus explores the naming conventions for entityies that frequently appear in the media like franchises or TV shows.

To simplify the annotation procedure, we created three types of lenses related to the mention's length and overlaps, as it can be easily seen from the following list:

1. *ØMIN* - minimal number of entities - e.g., this returns $m^{dbr:Star\_Trek:\_Picard}_{\text{[Star Trek: Picard]}}$.
2. *ØMAX* - maximum number of entities without overlap - e.g., this lens will match two mentions $m^{dbr:Star\_Trek}_{\text{[Star Trek]}}, m^{dbr:Star\_Trek:\_Picard}_{\text{[Picard]}}$.

[2]https //github.com/modultechnology/in_media_res

Example	ØMIN	ØMAX	OMAX
Sir Patrick Stewart OBE	1: Sir Patrick Stewart OBE	1: Sir 2: Patrick Stewart 3: OBE	1: Sir Patrick Stewart OBE 2: Sir 3: OBE
MLB Advanced Media (MLBAM)	1: MLB Advanced Media (MLBAM)	1:MLB Advanced Media 2: MLBAM	1: MLB Advanced Media (MLBAM) 2: MLBAM
Burbank, California	1: Burbank, California	1: Burbank 2: California	1: Burbank, California 2: California
Seinfeld	1: Seinfeld	1: Seinfeld	1: Seinfeld

Table 3.3: Examples of output for the three lenses. Numbers were added for simplified navigation.

Table 3.4: Comparison of NEL annotator performance on a corpora with multiple lenses (m - micro; M - macro; p - precision; r - recall; $F1$ - F1).

Corpus	*System*	mP	mR	$mF1$	MP	MR	$MF1$
Core set ØMIN	AIDA	0.47	0.48	0.47	0.43	0.48	0.43
	Spotlight	0.53	0.43	0.48	0.35	0.42	0.37
	Recognyze	**0.61**	**0.52**	**0.56**	**0.52**	**0.50**	**0.51**
Core set ØMAX	AIDA	0.49	0.48	0.49	0.45	0.48	0.44
	Spotlight	0.55	0.43	0.48	0.35	0.40	0.36
	Recognyze	**0.62**	**0.54**	**0.58**	**0.55**	**0.52**	**0.53**
Core set OMAX	AIDA	0.49	0.48	0.49	0.45	0.48	0.44
	Spotlight	0.51	0.57	0.54	0.51	**0.58**	0.52
	Recognyze	**0.65**	**0.61**	**0.64**	**0.61**	0.57	**0.59**

3. *OMAX* - maximum number of entities with overlaps - e.g., this lens will match two entities again, but with different mentions than the previous lens: $m^{Star\_Trek:\_Picard}_{\text{[Star Trek: Picard]}}, m^{dbr:Star\_Trek}_{\text{[Star Trek]}}$.

Table 3.3 provides several examples for these annotation rules. We included an example for each of the three classic types (Person, Location, Organization), as well as an example for Works.

An evaluation conducted on the primary partition of the dataset is presented in Table 3.4. Two of the examined annotators (DBpedia Spotlight and Recognyze) seem to benefit more from these lenses in all examined cases. All the annotators showed improvements of up to 4%.

3.1.5 Discussion

Several strategies for implementing name variance in NEL annotators were discussed: (i) extracting name variants from KGs; (ii) algorithmic name generation; and (iii) name analyzers. A combination of these strategies was found to offer good improvements over competing tools (e.g., AIDA, Babelfy), but only a small edge compared to DBpedia Spotlight.

It was discovered that to fully appreciate the impact of name variance on NEL, a number of changes need to be made to the evaluation procedures. The idea of reduced annotation styles that focus on a single property (e.g., mention, type, link) called lenses was introduced. The development of lenses is just one possibility that can be considered to fully evaluate the impact of name variance handling strategies. A corpus was developed to test this hypothesis. The examined tools have indeed performed better in these new evaluation settings that considered multiple possibilities.

3.2 Sentiment and Emotion

3.2.1 Background

Sentiment analysis (SA) is considered an umbrella technology [CLH11]. It is multilayered, as it needs to bridge syntax (e.g., POS tagging), semantics (e.g., NER, word sense disambiguation) and pragmatics (e.g., polarity, aspect, sarcasm).

The core problem of SA is determining if statements are positive, negative, neutral or ambivalent towards a certain object or idea. Emotion analysis (EA) provides a fine-grained emotion classification, as emotions are mapped to emotional categories. SA provides us with an evaluation of emotion, whereas EA tries to provide us with a better picture by linking the emotion to an emotional model.

3.2.2 Domain-Specific Affective Categorization Models

This section is based on [WSB+21].

An affective categorization model can be thought of as the taxonomy upon which an affective classification model is based. It contains the labels that will be predicted by the emotional classifier. Some well-known models include Cambria's *Hourglass of Emotions* and Susanto's revisited version *Hourglass of Emotions* [SLCC20].

Domain-specific affective models are created to compute certain indicators. If an organization, for example, defines its brand using certain phrases or words, it may want to follow the associations between these keywords and its name. In such a case, there may be a need to define a domain-specific model, as a model defined for media will not necessarily work for the publishing industry.

The process starts by annotating an existing model (e.g., old model) with data from KGs ike ConceptNet [SCH17] and Wikidata [Vra13] by mining phrases and synonyms/antonyms related to the seed terms. The annotations are then contextualized through mining for sentences that contain KG concepts. The example sentences showcase the usage of a term in context and basic word sense disambiguation algorithm is applied to compute the corresponding category associations based on the term's senses. The algorithm loops over the various senses and uses the LM to transform them to the corresponding embeddings for the respective senses. A set of refined categories is then computed based on the term's usage. For concepts that are not included in KGs, the algorithm uses the examples from the corpus. The next step is the effective expansion, in which the antonyms and synonyms are also added.

The affective knowledge extraction process then uses the expanded model and transforms the statements into embeddings that are used by LMs like BERT. A feature vector is then computed using the token, sentence and corresponding dependency parse tree (DP). A proximity search is used to compute the value of the affective category within the evaluated context. The score for category is then computed by considering negation and modifiers based on the DP.

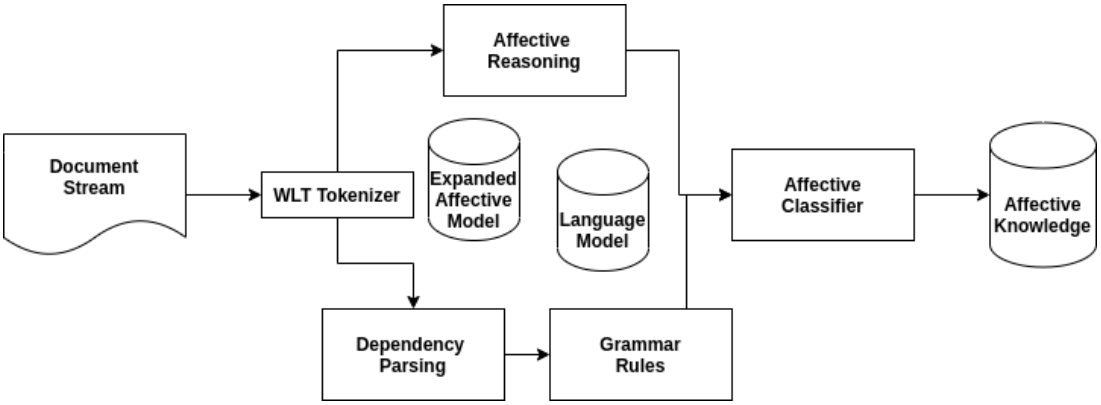

Figure 3.2: Affective knowledge classification architecture.

A special corpus collected from WikiNews was created for this evaluation. The annotation rules were based on previous challenges (e.g., SemEval [CNJA19], or SMM4H [SBF+18]), examples being provided for each of the four classes (*introspection, temper, pleasantness, eagerness*). In addition to the guidelines, annotators were also provided with the tables that explained the updated emotional categories from [SLCC20], and with a list of triggers for the polar opposites for each affective category. Eventually, the list was included in the annotations. The triggers contained lists of words that may provide cues to the emotional categories. Several examples from previous publications (e.g., [SLH+18] and [PHM+19]) were also selected for balancing purposes. Each annotator was asked to annotate 120 sentences and provide the statement's polarity, affective categories, and dominant emotion. The label *Unknown* was used for marking cases where a dominant emotion was absent, whereas

the *None* label was assigned to the emotionless statements. The corpus was created under the supervision of the expert with relevant experience in sentiment analysis and named entities. The annotators had the opportunity to consult the expert during the creation of the annotations for difficult cases. The dominant emotion was selected by averaging the scores for affective categories and polarity.

The updated categories (e.g., *temper*, *introspection*, *attitude* and *sensitivity*) from the Hourglass of Emotion [SLCC20] and associated concepts from the same publication were used to evaluate SenticNet 5, as the sixth version was not yet publicly available at the time [CLX+20]. Top 20 terms except illnesses were selected, whereas new terms were manually added for the newly introduced categories to create a balanced seed set.

Table 3.5: Recall of the *dominant emotion* based on the updated Hourglass of Emotions. Includes BERT/DistilBERT language models, with/without dependency parsing and grammar rules (GR).

category		**BERT**	**DistilBERT**	**BERT+GR**	**DistilBERT+GR**
T+	calmness	0.62	0.46	**0.75**	0.68
T-	anger	0.55	**0.65**	0.45	**0.65**
I+	joy	0.37	**0.46**	0.40	0.43
I-	sadness	0.76	0.80	0.74	**0.83**
A+	pleasantness	0.62	0.64	0.65	**0.67**
A-	disgust	0.68	**0.69**	**0.69**	0.68
S+	eagerness	0.38	0.36	**0.46**	0.36
S-	fear	**0.90**	0.87	0.80	0.70
overall		0.61	0.63	0.62	**0.64**

What is interesting to notice is that many sentences were assigned, as we expected, a neutral value. This is partially because the statements were collected from journalistic sources (Wikinews), so they had to be at least in theory impartial. Another observation is the fact that a single trigger often impacted non-neutral sentences. The scores are further boosted by the addition of dependency parsing (DP) and grammar rules (GR).

Tables 3.5 outlines performance gains achieved by applying Transformer LMs such as BERT and DistilBERT. Several other models were also tested (e.g., RoBERTa, XLNet), but since we discovered that the classic BERT model (*bert-base-uncased*) and the distilled model (*distillbert-base-uncased*) yielded the best early results, we have only considered these in the final evaluations.

3.2.3 Discussion

A negative bias was confirmed in the gold standard, as more negative examples were found. This is confirmed by the literature, as it is known that political articles tend to have negative connotations [LEB12].

The corpus and associated evaluation demonstrate that the proposed method works well with LMs of various sizes, even in environments in which resources are scarce. The evaluation also shows that the method can be used to update previous approaches (e.g., to include negation, if needed).

3.3 Fact Verification

3.3.1 Background

This section is based on [BA19], and its extended version published in [BA20a].

There are multiple instances when fact verification is needed. If a date of birth present in a KG is wrong, we can be verified in several sources. When there is no clear proof of what happened, facts may need to be established. In online environments it corresponds to tracking the provenance of the data in to understand who released it and why. Another possibility is to simply use the texts as they were written together with some context data (if available). Fake news detection would generally be such an instance. Fake news can be seen as a part of several other larger problem classes, including propaganda detection and fact verification [TVCM18]. For the rest of the section we will consider it as being part of the fact verification class.

Propaganda detection is a much larger class which goes beyond fact verification, as it can also include images or videos of deceptive nature, and therefore vision is important when considering it. A survey on automated propaganda identification can be found in [MCB+20]. Fake news is now a large interdisciplinary field that is difficult to fully capture by looking only at NLP articles. Various perspectives can be found in a recent set of surveys [PCLG21].

3.3.2 Fake News

The definition of fact verification suggests that the core of the task is a relation prediction problem:

Definition (based on [PZS+20]). *Given two sequences, a set of statements S={s1...,sn} and set of known facts F={f1,..,fn}, predict the relation between the two sequences.*

The relation is the degree of support and can be modelled as a label that indicates if the statements are supported or refuted by the known facts. The known facts can be the rest of the known attributes from the dataset, or computed features.

Most of the definitions for fake news, since they were defined through political lenses, do not consider the relation prediction as being the core of the problem. The classic definition is based on the 2016 US Election study [AG17]:

Definition. *News items or partial news items can be considered fake if their content is proven false.*

The corresponding ML problem is classification, either binary or multi-class, depending on the dataset. Input contains short statements. It is also possible that additional information is provided (e.g., date, location). The output is a label (e.g., binary or fine-grained).

3.3.3 Semantic Fake News

To exploit the necessary semantic capabilities needed for identification of fake texts, the pipeline used for this task includes:

- *Annotations* - the basic pipeline includes POS tagging, entities, and sentiment;
- *Relations* - computed from text and KG;
- *Neural models* - DL models which include embeddings.

Relations were computed from the text and KG. Two types of relations were extracted from the text i) between entities (e.g., verbs between proper nouns), or ii) between entities and objects (e.g., verb between subject and object). Between entities relations were also extracted from DBpedia where possible.

Liu's **Liar** [Wan17] and Rashkin's **Politifact** [RCJ+17] datasets were published in 2017 and contain similar data that was extracted from a political fact-checking site[3]. The main requirement for both datasets is to annotate short texts with six unbalanced classes which represent their degrees of truth (from *True* to *Pants on fire*). Unlike tweets these statements do not contain the specific abbreviated language of Internet texts (e.g., emojis, retweets), but rather natural language.

The wealth of data available for these datasets allows for several experimental configurations: (i) only the statements (essentially the texts, hence labelled with **T**); (ii) the original attributes or **T+A** (e.g., statements plus the other features present in the datasets); (iii) text plus semantic features labelled **T+R**; and (iv) all the combined features (or **ALL**). It has to be noted that Politifact contains only texts. This suggests that the **T+A** is not needed in reality.

The DL models in particular use the Glove 300 model loaded with the Keras API. The embeddings are placed on the first layer after inputs, as such a placement was considered good for processing small datasets [QSF+18].

Tables 3.6 & 3.7 present the considered models and their scores. They report test set accuracy scores. We have split the tables into classic ML (e.g., statistical or generally pre-DL models) and DL sections. DL models were found to perform better. This is not always a foregone conclusion, as Conditional Random Forest (CRF) ensembles are known to perform well for semantic issues [PRT16].

For statistical ML models [HTF09], the improvements obtained with additional semantic features are small (e.g., 2-3%). Scores are low, and two

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

Model	T	T+R	ALL
Classic ML			
Multinomial Naive Bayes	0.224	0.244	**0.262**
SGDClassifier	0.239	0.235	**0.255**
Logistic Regression (OneVsRest)	0.240	0.260	**0.273**
Random Forest	**0.215**	0.215	0.212
Decision Trees	0.226	0.249	**0.262**
SVM	0.255	0.275	**0.294**
Deep Learning			
CNN	0.241	0.270	**0.289**
BasicLSTM	0.245	0.289	**0.326**
BiLSTM Attention	0.419	0.448	**0.499**
GRU Attention	0.450	0.496	**0.539**
CapsNet	0.565	0.598	**0.649**

Table 3.6: Liar test set accuracy. Best results are displayed with **bold**. **T** signifies text and **R** represents relations.

classifiers show the same results for Politifact (e.g., decision trees and logistic regression). Interestingly, the Random Forest classifier is the only one that doesn't show improvements with additional semantic features, but the model was basic, and not an ensemble as it is typically suggested in the literature. The supremacy of SVM is confirmed for both datasets for this class of algorithms.

The DL models use hot encoding of the class labels. They use TensorFlow [ABC+16], Keras [Cho17], categorical crossentropy loss and the Adam optimizer [KB14]. Preprocessing steps included: document cleanup, removal of stopwords, tokenization, transformation of text into sequences, and padding. Besides the CNN and LSTM, the rest of the models use the Glove300 model. Input vectors were fed directly into the Keras's embeddings layer (the first hidden layer of the network). Pre-trained models used as much as possible. Same learning rate (LR) and batch size is used everywhere and the same stopping condition.

The CNN model was based on [Kim14] and [LLX+17]. It includes an embedding layer with dropout at 0.2, a Convolution1D filter for word groups, GlobalMaxPool, and a dense hidden layer with dropout at 2.0 and ReLU activation. The result is projected on an output layer with a single unit squashed with a softmax.

BasicLSTM is an LSTM with dimension of 300, GlobalMaxPool, spatial dropout at 2.0, and dense hidden layers with softmax activation. Hyperparameters include batch size of 256, 20 pochs and LR=0.001.

BiLSTM [CN16] is based on the CuDNNLSTM with attention implementation, dropout and recurring dropout at 0.25, and a dense hidden layer activated with softmax. Same parameters that were used for BasicLSTM were used.

GRU [ITA+16] used similar settings and hyperparameter like the previous

Model	T	T+R	ALL
Classic ML			
Multinomial Naive Bayes	0.263	0.295	**0.296**
SGDClassifier	0.262	0.294	**0.295**
Logistic Regression (OneVsRest)	0.246	0.269	**0.269**
Random Forest	**0.244**	0.229	0.229
Decision Trees	0.246	0.269	**0.270**
SVM	0.262	0.281	**0.282**
Deep Learning			
CNN	0.203	0.231	**0.244**
BasicLSTM	0.245	0.287	**0.282**
BiLSTM Attention	0.371	0.422	**0.422**
GRU Attention	0.415	0.451	**0.452**
CapsNet	0.473	0.523	**0.524**

Table 3.7: Politifact test set accuracy. Best results are presented with **bold**. **T** represents text and **R** signifies relations.

model.

CapsNet is based on [FK19, KJPC20]. It contains a Capsule layeras a replacement for the GlobalMaxPool layer. BidirectionalGRU with dimension 128 activated by ReLU, and with dropout and recurrent dropouts set at 0.25. The result is sent to a single unit output layer squashed with a sigmoid. Additional parameters include 10 capsules with dimension 16 and 5 routings. Number of training epochs was 5.

3.3.4 Discussion

The statistical models are not well-prepared for this task. The findings suggest that semantics and good preprocessing can be the key to better results, as accuracy increases of up to 4.2% can be obtained simply by adding semantic attributes, or up to 10% if using attention models.

These scores are meant to be interpreted as baselines and not as the top scores for this task. Computing baselines is central for rapid development, especially if the code can be used in different settings (e.g., research or production). It is an important distinction, as it means that, using these techniques, it should be possible to quickly build a good classifier that is based on well-known DL models. This is the main reason why most of the models were restricted to their basic functionality.

Chapter 4

EXPLAINABILITY IN SEMANTIC AI

4.1 Explainable Benchmarking

4.1.1 Introduction to NEL Benchmarking

Early NEL benchmarks simply provided the classic precision, recall and F1 scores without adding any interpretation. Most of the benchmarks were not really standardized, but instead relied on the metrics APIs provided by ML libraries like *Scikit-learn*[1] or similar APIs. Such APIs were designed to deliver black-box results, meaning the users were provided with the final scores, but without too many details on what went wrong. Of course, this was ideal for cases in which results were good, but less so for test runs that were full of errors.

NEL evaluations typically involve several components:

- Dataset (Gold) - is the main dataset holding the reference annotations that will be used for running the experiments [HLAN12].
- KG - the reference KG used for the annotations. KGs are often updated, and therefore links can be added or changed [RUN18].
- Annotator - the automated annotator that generates the output [RUN18].
- NIL Clustering - it is typically a component of the automated annotator that handles the clustering of the various entities (e.g., a person can be identified through multiple names, but all those names should ideally end up in the same cluster) [HNR14].
- Scorer - the scoring script used to produce the final scores. A scorer will return the classic performance metrics (e.g., precision, recall, F1) [HNR14].

Each of these components may generate errors, either due to lack of updates, or effectively due to unforeseen circumstances.

[1] https //scikitlearn.org/stable/modules/classes.html#modulesklearn.metrics

When evaluating NEL results it is common to report the classic measures like precision (P), recall (R) and F1 score ($F1$) as described in the previous section. Sometimes scoring can also be influenced by overlaps. It is generally assumed that entities can find themselves in one of the following situations: i) perfect matches (e.g., if the mention text matches the entire surface form); ii) fully contained in the gold surface forms (e.g., if the returned entity is contained in an example provided in the gold); or (iii) overlapping with the accepted gold solutions (e.g., the entity could extend further than what was recorded in the gold standard).

Some well-known NEL benchmarking suites include BAT framework[2][CFC13], neleval[3][HNR14] and Gerbil[4][RUN18]. Most of these systems are built around the black-box philosophy, and generally offer tables with evaluation results, but fewer explanations or visualizations.

4.1.2 A Taxonomy of Errors in NEL Systems

After analyzing the output of the neleval suite [HNR14] for several annotators, including DBpedia Spotlight [DJHM13], Babelfy [MRN14b], AIDA [HYB+11], and Recognyze [WKB18], we tried to create a classification of the resulting errors. We started by collecting various errors observed in documents from the datasets KORE50 [HSN+12], Reuters128 [RUH+14] and RBB150 [BNWS16]. After a preliminary discussion, we decided upon the categories from the taxonomy. We then proceeded to annotate 50 documents from each of them with neleval and manually tag the errors based on the taxonomy. A summary of the preliminary findings can be found in Table 4.1.

NEL System			**Gold Standard**		**Error**	
Entity Link$_s$	**ET$_s$**	**SurfaceForm**	**Entity Link$_g$**	**ET$_g$**	**Type**	**Cause**
Bruce_Willis	ORG	expiration	-	-	KB	Redirects
(de.)2009	LOC	2009	-	-	KB	Wrong Type
United_States	LOC	U.S.	-	-	DS	Missing Annotation
New_York_City	LOC	New York	New_York	LOC	DS	Wrong Annotation
(de.)Berlin	LOC	Berlin	Berlin	LOC	DS	Different Language
JFK	PER	Kennedy	JPK	PER	AN	Same-Type
Beck	ORG	Beck	Jeff_Beck	PER	AN	Cross-Type
Barack_Obama	PER	Malia Obama	NIL	PER	NIL	Wrong Cluster
NIL	ORG	Knicks	New_York_Knicks	ORG	NIL	Partial Match
Miles_Davis	PER	Davis	Miles_davis	PER	SE	Correct Redirect

Table 4.1: Common errors in NEL. Entity Type (ET) is marked with subscripts *s = system* and *g = gold standard*. The links represent abbreviated English DBpedia links, and the (de.) prefix signifies German version.

Five large error cases were found, even though one of these classes is

[2]https://github.com/marcocor/bat-framework
[3]https://github.com/wikilinks/neleval
[4]http://gerbil.aksw.org/gerbil/

rather less common.

KG errors are generally errors that were though to have originated in the KG. They were probably collected from dumps or old live versions. Some examples include annotations in different language (e.g., German annotation instead of English), wrong surface forms (e.g., words that have nothing to do with the target) or redirects. Sometimes such errors can only be identified if the researchers have access to the KG version used for annotations.

The **DS** errors are probably the most problematic, as they are harder to fix if the corpora is not published with a permissive license. Some of these errors may be similar to KG errors (e.g., wrong surface form, different annotation language), but they were observed in the DS. Such errors can sometimes be fixed by using lenses, as discussed in Chapter 3.1.4.

The **AN** errors are the most frequent errors observed. They can include everything from different typing, to wrong abbreviations or generic terms returned instead of an entity. These errors are caused by the annotator settings, and may be removed if other settings are used.

NIL clustering errors are spread between partial matches, or name sharing between clusters (e.g., when roles or titles are assigned into multiple clusters). Early clustering algorithms rarely used co-reference resolution mechanisms and were more error-prone [Rad15]. Co-references are especially necessary to detect the cases in which a proxy is used instead of the entity names.

The least frequent errors were **SE**. We have only discovered one such error that was caused by redirect (e.g., a Miles Davis link that was a redirect instead of the correct link was classified as the correct link).

Various error counts for these errors were also collected. The code used for computing the errors was later included in Orbis.

4.1.3 Orbis

Leaderboards are used in many challenges as means for promoting competition and motivating research teams to further improve upon their results. Nevertheless, providing developers with feedback which is limited to their rank on a board may not be sufficient toward identifying means for improving their system. The follong pages describe Orbis, a system published in [OKBW18].

Since the current generation of annotation tools rarely publish their best settings and annotation guidelines are – to the best of our knowledge – rarely available in machine-readable formats, the last two steps can be considered research topics onto themselves for now.

Orbis was designed with Open Data principles in mind, and therefore it supports the FAIR data publishing methodology [JdMAJ+20]. The FAIR acronym stands for *Findable, Accesible, Interoperable, and Reusable*. At their core, these principles were created to support the reproducibility of research data. Almost all FAIR principles are implemented in Orbis, except for the qualified references to other metadata and searchable metadata.

Orbis currently supports the following tasks: (i) NER; (ii) NEL; (iii) Slot Filling; and (iv) forum extraction, which is a special case of content extraction

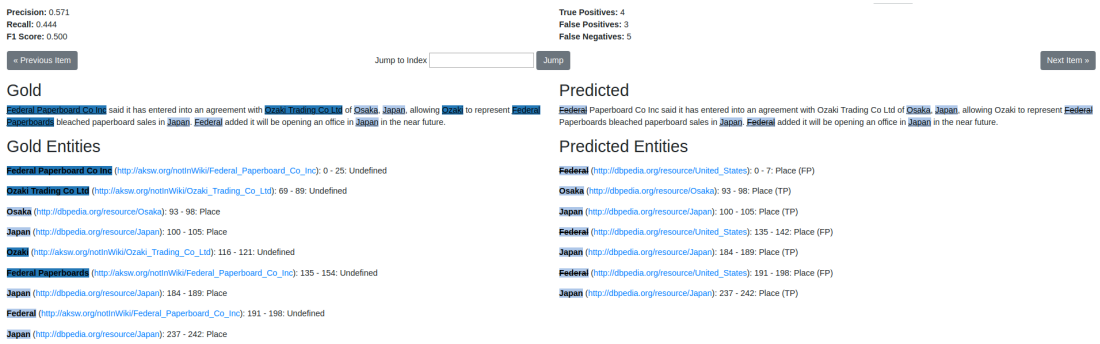

Figure 4.1: The types classification. Corpus: Reuters128. Task: NEL

(CE) [WBWO21]). While forum extraction is not an entity-focused task, it was considered an important NLP task, as the quality of the extracted metadata is only as good as the quality of the extracted text. Orbis was developed through multiple research projects. A large number of datasets and annotators was integrated.

The Orbis pipeline was built around the NIF [HLAN12] format for publishing NLP data. It was designed around a core pipeline described through a YAML configuration file. The core pipeline loads the data (e.g., gold standards, annotator outputs) and sends it to an assessment component that produces a confusion matrix and the desired performance metrics. The results are converted into various output formats and turned into analytics via a set of plugins. The rest of the components are plugins designed to perform a single task like reading gold standards or displaying annotated documents.

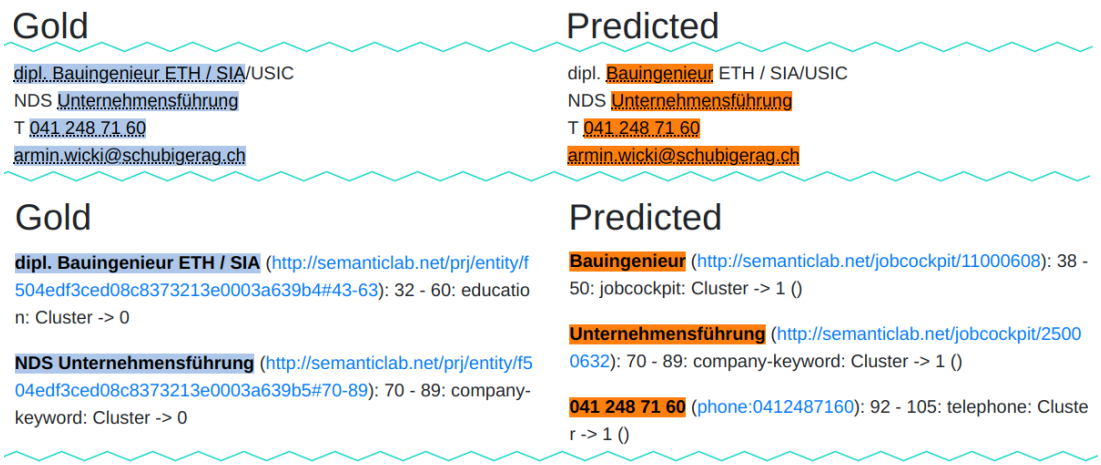

Figure 4.2: A cluster classification screenshot. Zigzag lines represent compressed white spaces. Corpus: JobCockpit. Task: Slot Filling.

The interface is built around a dual display of gold and annotator results. Users can select between multiple classification schemes. For each scheme a different type of coloring was implemented to notify users that a different behavior is expected. The basic classifiers are the following:

- *Entity* - each entity is displayed with a different color to quickly identify it;
- *Type* - colors match entity types;
- *Result* - colors mark predicted results (e.g., TP, FP, TN, FN).

Depending on the task, custom classification and associated color schemes can be added. Current version of Orbis supports the following custom schemes:

- *Cluster* - all slots that belong to a certain entity are displayed with the same color (only for Slot Filling tasks).
- *Paragraph* - instead of entities, colors identify blocks of texts (only for the Forum Extraction task);
- *Error* - errors are highlighted with different colors (only for error classification tasks).

On top of each page, Orbis provides an overview with additional information related to an evaluation. This view describes the evaluation settings (e.g., evaluation type, tool(s), datasets, etc). For debugging purposes, a reduced set of this functionality, e.g., only general settings and results, can be displayed on each page.

4.1.4 Discussion

This section concludes the discussion about explainable benchmarking and revisits several ideas about improving benchmarking processes that were presented in [WBKN19].

As we have seen in the previous sections, debugging NEL processes is a difficult job. Problems may arise on different layers (e.g., KG, DS). While annotators will cause most of the issues, scorers can also introduce errors.

Perhaps the worst issue is the fact that there is no study related to the correctness of the gold standard. While several articles do mention this problem (e.g., [vEMP+16] and [JRN17]), no solutions are provided. Another issue is that annotators can be optimized for certain datasets (e.g., by training excessively on them). This issue can be solved if the best settings are published for each annotator and dataset, but unfortunately papers and published code do not offer these best settings.

Well-known KGs (e.g., DBpedia or Wikidata) are known to new dump versions frequently (e.g., monthly or weekly). This suggests that comparing today's output with a gold created several years ago may not be enough. This led to our advocacy for the importance to keep records about the KG version used for annotating the datasets, as well as those used by annotators.

4.2 The Role of Interpretability and Explainability in AI

4.2.1 Interpretation and Explanation in Model-Agnostic Libraries

In contrast to the section about explainable benchmarking which focused on a single big use case, here we will draw on a survey published in [BA20b] and try to understand the general trends in the field. This will help us move towards the general conclusions of the work.

Programmers with statistics knowledge will defend the term *interpretation* over the term *explanation*. The term explanation on the other hand may be preferred by artists and visualization designers [BA20b]. There are of course some researchers in the middle who may use the terms interchangeably. For NLP researchers, we can argue that explanation should also be used, as we would like our models to offer us clear explanations in natural language.

Providing a clear answer on why a model offered a certain prediction is a difficult task. In terms of engineering we can assume that the result is the sum of the model features and proceed to deconstruct the contribution of each feature. This has been the default method for several decades [GE03].

In the last decade, things have evolved. Modern libraries are built around the idea of model-agnosticism. They claim to be able to interpret the prediction of any model. To a certain degree this is possible. For example, for computing Shapley values, multiple features are aggregated and a score that represents the within-set contribution is computed. Such techniques form the basis of libraries like LIME[RSG16], SHAP [LL17] or ELI5 [FJP+19].

The main criticism against such libraries relies on three ideas: i) they are in reality easy to use, but difficult to interpret; ii) they may end up picking statistical effects rather than compose interpretations; and iii) adversarial attacks against them can be easily propagated.

The last part of the criticism may be valid for any kind of interpretability method. Slack [SHJ+20] demonstrates how to perpetrate simple attacks by deploying biased classifiers. Since most models are in fact ensembles, changing the output of a single classifier may bring down the entire construct. A single classifier would be enough to add bias against a certain category of people or to change a credit score. Robust attacks include: creating fake token sequences that concatenated to strings may turn into universal triggers [WFK+19]; partial removal of training samples [CTW+20]; random spelling attacks [SHY+20]; and many others.

Statistics offer an alternative to XAI: Neural Additive Models (NAMs). They combine Generalized Additive Models (GAM) [AFZ+20] with DL features.

4.2.2 Explaining Recurrent Neural Networks

Providing a good explanation of the neural networks results requires a bit more effort than creating a visualization of the hidden states. The best idea is to follow the flow of information from inputs to outputs, therefore visualize the corpus, embeddings, the attention heads and various hidden layers, the training procedure and the output, for example. This requires a lot of effort, and very few teams have followed this idea. Instead, the more common case involves focusing on visualizing a single topic like embeddings or attention heads.

We discovered three large clusters of papers. Each cluster contained significantly more papers than what is discussed here (at least ten times more papers, based on our estimations). Only papers that introduced new concepts or had numerous citations were considered for inclusion.

Prediction of next topics is a classic topic in NLP. The results are presented through classic charts (e.g., single or parallel charts, matrix views). All of these visualizations are practical, as it can easily be discovered by reading them.

The representation of hidden states is the central topic in explaining RNNs. The visualizations included here are almost like mini dashboards. They all include a control panel for navigating between sentences or documents; a set of word clusters or neural activations; and matrix views which are ideal for highlighting results. Several systems that follow these patterns include: ActiVis [KAKC18], RNNVis [MCZ+17] or LSTMVis [SGPR18]. Designing such interfaces is a collaborative activity, therefore most of the papers include a long list of authors.

Graph Convolution Networks (GCNs) are included here as well. Dedicated libraries (e.g., PyTorch Geometric [FL19]) exist, but are focused on selected models. The papers we mention were published during the last year, but they showcase the fact that graphs and line charts are enough for these kinds of explanation.

4.2.3 Explaining Transformers

The Transformers need only attention and a pair of encoders and decoders to provide better than average results for many tasks. It is only natural to ask how is this possible? It is also natural that most Transformer visualizations are preoccupied with attention. We focus instead on searching for those visualization that attempt to provide us with a clear understanding of the entire model, from corpus, to attention maps, to neural layers, and multilingual outputs.

The central question of the paper that launched the architecture: if attention is itself enough for solving many tasks [VSP+17] is still controversial, despite the high number of citations. Partially this controversy is fuelled by the high number of GPU or TPU pods needed for the training phase. In the end, if we throw all the resources of several small countries to solving a single problem, chances are high that we will succeed, but at what costs for the environment or for the rest of the world [SGM19]? What will happen, for example, to countries

or companies that have less resources? How will they compete? A method to test if attention is enough as explanation involves studying the effects of weight manipulation on outputs [JW19]. If outputs are changed, the explanations probably capture the right information. This is equivalent to studying the entropy flow through the system. A different paper suggests that such a technique should only be applied in limit cases, for example if adversarial training doesn't lead to serious changes in the weight distributions [WP19]. The stochastic parrot criticism [BGMS21] revolves around the fact that biases cannot be easily removed from LMs due to the high costs associated with their retraining; but this criticism is refuted by many researchers who view it as political activism [Lis21]. In our view, biases can be identified and removed in time.

Visualizing attention provides some insights, so it can be considered a form of explanation, even if not always granular. Transformer visualization are also dedicated to embeddings and dependency parsing [RYW+19], attention weights during pre-training or training (e.g., [Vig19] or [SZC+20]), encoded linguistic phenomena like prepositions or co-references [CKLM19] or structural probing [HM19]. As it was the case with RNNs, visualization of hidden states also occupies a significant amount of literature.

We can discuss two categories of Transformer visualizations: (i) focused (or single topic), and (ii) holistic (e.g., dedicated to the entire workflow or model).

Some trending topics in focused visualization include attention (e.g., [AZ20], [VTM+19], or [Vig19]), probing [VST19], effects of information interaction ([HDWX20], and [VST19]), or multilingual models [TDP19].

Probing is considered a special kind of explanation that showcases the linguistic information encoded in vectors [EER16]. Structural probes [HM19] solve a limited version of this problem: testing if syntax tree are embedded in a neural network's word representation space. If such evidence is found, then it can be assumed that the LM's vector geometry embeds the respective syntax trees. Critics argue that the method works well for cases in which word distances are known, but not when huge differences appear for various classifier accuracies. Also, the provenance of the LMs hardly matters, as even BERT-based models can developed novel linguistic representations, despite their shared origin. Voita [VT20] suggested that probes should transmit some data (e.g., a description or label) which can be evaluated based on its length. The mechanism is stable when implemented on top of structural probes.

Very few holistic visualizations include the training corpus (e.g., [SES+20], and [HSG19]) or the associated dictionaries [YCOL21], although these errors can propagate to the downstream tasks [BRK+18]. Transformer errors are examined in [CKLM19]. Only a small subset of the systems include views for all the important components, including the corpus, embeddings, attention and layers (ExBERT [HSG19] and AttViz [SES+20]).

None of the examined systems manages to capture the entire complexity of a Transformer system. One of the main reason is the excessive focus on the role of attention. The lack of details about encoders and decoders is another one. This is a complex design issue. Combining both the *form* (e.g.,

the architecture with its encoders and decoders) and *function* (e.g., the neural pathways of the information) in a single interface is difficult. It is typical to focus only on one of them. The form is emphasized when the design is focused on circuits and logics, whereas the function is emphasized when design is focused on the process. For NLP, the focus on function is enough. To understand why these networks work so well for different classes of problems, including vision, it is best to find a compromise between both, or alternatively to create two separate visualizations.

4.2.4 Language and Vision

This is essentially the first step towards the merging of the various branches of AI like vision, NLP, speech, SW or robotics. During the last couple of years this topic was unavoidable at ML conferences. More details about these models can be found in the Visual Question Answering (VQA) survey [GCL+20].

4.2.5 Discussion

The last pages have showcased the fact that even if it may not be clear if attention is enough to explain the reasoning of NLP systems, visualizing attention may be a route towards clear explanations.

The modern design of DL visualizations [SGPR18] was established around the time when the Transformer architecture was published [VSP+17]. The basic idea was to split the architecture according to function and focus on several components like the inputs, hidden states and outputs. This offered a high-level view of the information workflow. Due to the timing of the respective publication and the massive adoption enjoyed by Transformers, more visualizations were based on it for the Transformer architecture than for the other architectures. It can almost be argued that Transformers provided a solution for most NLP problems, if not for the controversies related to the cost of training and bias.

What is clear is that most visualizations are now model-oriented, whereas a universal visualization framework (or at least model-agnostic one) for neural models does not exist yet. Such an achievement would open the door to universal explanations. It is not sure that it will convince the skeptics, but it is worth building towards it. We do not want an approximate understanding of AI. We need to get a clear understanding of it.

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Impact

These chapters are based on a set of conference and journal publications. Impact factor (IF) presented in the tables is generally for 2019 (published in 2020). For conferences, we considered the CORE rankings from 2021, or from the last available year for the respective conference.

Chapter 2 described the main benefits and showcased some applications of KGs: a tourism KG published in a conference article [SBÖ15] and a journal article [SOBS16]; as well as a dashboard built around it [BSS+17].

Some details about these publications are included in Table 5.1.

Chapter 3 reviews various contributions related to the development of the SAI systems.

The first set of contributions develop the idea of name variance in NEL systems, first through algorithms [WKB19], then through lenses [BWN20] that can help when evaluating results. These contributions were built on top of a NEL system called Recognyze, which is briefly presented [WKB18], while a contribution related to a slicing tool is brieft mentioned [MSS+17]. The second set of contributions develops the idea of expanding affective models and building quick baselines on top of them. A publication about it was accepted at Cognitive Computation, a prestigious Springer journal [WSB+21]. The last set of contributions is dedicated to fact verification, and is dedicated to building quick baselines using well-known LMs and semantic attributes. This contribu-

Table 5.1: Impact for Chapter 2.

Publication	Venue	Type	Rank/IF
[SBÖ15]	ENTER 2015	Journal	C(2021)
[SOBS16]	Journal of IT & Tourism	Journal	IF=2.95
[BSS+17]	Semantic Web	Journal	IF=3.524

Table 5.2: Impact for Chapter 3.

Publication	Venue	Type	Rank/IF
a) NEL			
[BNWS16]	LREC 2016	Conference	C(2021)
[MSS+17]	IEEE ICSC 2017	Conference	N/A
[WKB18]	ACM WIMS 2018	Conference	N/A
[BNW18]	ACM WIMS 2018	Conference	N/A
[WKB19]	LDK 2019	Conference	NEW(2019)
[BWN20]	ACL CoNLL 2020	Conference	A(2021)
b) Sentiment Analysis			
[WSB+21]	Cognitive Computation	Journal	IF=4.307
c) Fact Checking			
[BA19]	IWANN 2019	Conference	B(2018)
[BA20a]	Neural Processing Letters	Journal	IF=2.891

Table 5.3: Impact for Chapter 4.

Publication	Venue	Type	Rank/IF
a) Benchmarking			
[BRK+18]	LREC 2018	Conference	C(2021)
[OKBW18]	SEMANTICS 2018	Conference	N/A(2021)
[WKB19]	ACL RANLP 2019	Conference	C(2021)
[WBWO21]	IEEE/WIC/ACM WI 2020	Conference	B(2021)
b) Visualization			
[BA20b]	IEEE IV2020	Conference	B(2021)

tion also resulted in conference [BA19] and journal [BA20a] publications.

The publications summarized in Chapter 3 are included in Table 5.2.

Chapter 4 is built around the idea of explainability. The first three sections summarize contributions to the NEL benchmarking, including a taxonomy for error analysis [BRK+18], a NEL benchmarking system [OKBW18], as well as some ideas about how to improve the benchmarking process [WBKN19]. A contribution related to an adjacent topic (forum extraction [WBWO21]) is also mentioned. The last part of the chapter is dedicated to a survey of the LM explainability and helps contextualize previous sections, while also suggesting new research directions [BA20b].

The contributions discussed in Chapter 4 are included in Table 5.3.

The last chapter reviews the contributions.

5.2 Conclusion

This section contextualizes and expands upon the discussion sections that followed each section of the work.

The early observations from Chapter 2 about the limitations of classic semantic systems served as the launchpad for the rest of the contributions. It is clear that KGs are useful, and that they need to complement KGs to create good representations.

Chapter 3 discussed three apparently unrelated NLP applications: NEL, sentiment and fact verification. In reality, they build upon each other, and this is why the sequence was arranged in this order. Entities are needed everywhere. They can also be used during the sentiment computations and fact verification. Similarly, the particular instance of fact verification discussed here (fake news detection), needs entities and sentiment.

It is clear that the issue of name variance (Chapter 3.1) should be important for the design of both NEL systems, and their benchmarking systems. Both contributions lead to small improvements in the treatment of name variance for NEL systems (up to 2% for algorithmic implementations; up to 4-10% for lenses). The issue of name variance is treated generally through the implementation of partial matches in systems like neleval [HNR14]. A single lens can be considered to be equivalent with a partial match. A series of lenses (e.g., like those presented in Section 3.1.4, on the other hand, can fully cover all the cases of name variance. This is the main reason why lenses were developed in the first place - to cover as many cases of variance as possible.

Chapter 3.2 covers a method to expand affective models in conditions of resource scarcity. The extensive evaluations showed that the method works. The goal was to use the method both in research and production environments. The thesis covered the research use cases. The code was later adapted and included in production environments. What is important to remember is that LMs were used as a part of a larger ensemble that also included KGs, word sense disambiguation algorithms, and sentiment lexicons.

Chapter 3.3 showcases how classic or DL systems can be used to detect fake news. The entire section shows that by adding several semantic attributes like entities, sentiment and relation, it is possible to obtain good results. The method is effective for creating fast baselines.

The methods discussed in this section have several attributes in common: i) they all use KG and ML; ii) they treat issues related to some kind of variance (e.g., name variance in NEL, domain adaptation for sentiment, degree of truthfulness for fact verification); and iii) they all lead to good baselines. To improve upon these results, more sophisticated architectures can be imagined.

Chapter 4 is focused on explainability. Multiple contributions are discussed.

Chapter 4.1 presents three contributions related to explainable benchmarking: i) a taxonomy of errors; ii) a tool built for visualizing benchmarking; and iii) a proposal to improve the publication of corpora by including additional attributes in their metadata. All these contributions lead towards a clear idea: more things need to be done to improve NEL benchmarking. The first steps in this direction were taken. However, this can only be achieved if the entire community agrees to participate.

Chapter 4 provides an overview of the current status of the visualization

methods that help explain LMs. The results are surprising. A lot of progress was made during the last 3-4 years. However, most visualizations are focused on the functionality of the LMs. Achieving some kind of balance between visualizing architectures and their function may be needed. The architecture essentially place some restrictions on what can be implemented. The additional information about architecture (e.g., what operations are supported? how are these operations visualized?) can lead to some interesting insight. This avenue is not explored yet.

The general criticism toward ML, and by extension towards SAI, focuses on the core issues of dependency (e.g., data or domain-dependency), consistency (e.g., knowledge transfer, fine-tuning) and transparency (e.g., reproducibility) [CPGT17]. This thesis has provided some ideas on how to approach some of these issues. It has described how to tackle the issue of domain adaptivity by using KGs and LMs. It has approached the issue of tweaking existing models repeatedly through the discussions built around the performed evaluations. The topic of transparency was discussed in the context of benchmarking, as well as in the context of explainability. These are only some possible solutions. They worked for the respective use cases. They may not work for other use cases. If SAI is to conquer the world, these issues will eventually be solved. If at least some ideas presented here are examined by other people, the work has served its purpose.

5.3 Future Work

There is a renewed interest in designing visualization methods for explaining the results of neural networks. The main challenge will be to capture both the architecture and the function of the visualized networks.

Another interesting research area could be applying NLP to time-series. This may include the development of new sentiment indicators and their visualizations.

Perhaps the most important future research direction is to understand how can SAI systems survive without an external representation of the world (e.g., KGs, maps, For now, our view, is that Semantic AI systems will always need interfaces through which to save their representations.

Bibliography

[ABC+16] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek Gordon Murray, Benoit Steiner, Paul A. Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zhang. Tensorflow: A System for Large-Scale Machine Learning. *CoRR*, abs/1605.08695, 2016.

[AFZ+20] Rishabh Agarwal, Nicholas Frosst, Xuezhou Zhang, Rich Caruana, and Geoffrey E. Hinton. Neural additive models: Interpretable machine learning with neural nets. *CoRR*, abs/2004.13912, 2020.

[AG17] Hunt Allcott and Matthew Gentzkow. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2):211–36, 2017.

[AOOV20] Tareq Al-Moslmi, Marc Gallofré Ocaña, Andreas L. Opdahl, and Csaba Veres. Named entity extraction for knowledge graphs: A literature overview. *IEEE Access*, 8:32862–32881, 2020.

[AS19] Heike Adel and Hinrich Schütze. Type-aware convolutional neural networks for slot filling. *Journal of Artificial Intelligence Research*, 66:297–339, 2019.

[AZ20] Samira Abnar and Willem H. Zuidema. Quantifying attention flow in transformers. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4190–4197. Association for Computational Linguistics, 2020.

[BA19] Adrian M.P. Brasoveanu and Răzvan Andonie. Semantic fake news detection: A machine learning perspective. In Ignacio Rojas, Gonzalo Joya, and Andreu Català, editors, *Advances in Computational Intelligence - 15th International Work-Conference on Artificial Neural Networks, IWANN 2019, Gran Canaria, Spain, June 12-14, 2019, Proceedings, Part I*, volume 11506 of *Lecture Notes in Computer Science*, pages 656–667. Springer, 2019.

[BA20a] Adrian M.P. Brasoveanu and Răzvan Andonie. Integrating machine learning techniques in semantic fake news detection. *Neural Processing Letters*, pages 1–18, 2020.

[BA20b] Adrian M.P. Brasoveanu and Răzvan Andonie. Visualizing Transformers for NLP: A brief survey. In Ebad Banissi, Farzad Khosrowshahi, Anna Ursyn, Mark W. McK. Bannatyne, João Moura Pires, Nuno Datia, Kawa Nazemi, Boris Kovalerchuk, John Counsell, Andrew Agapiou, Zora Vrcelj, Hing-Wah Chau, Mengbi Li, Gehan Nagy, Richard Laing, Rita Francese, Muhammad Sarfraz, Fatma Bouali, Gilles Venturini, Marjan Trutschl, Urska Cvek, Heimo Müller, Minoru Nakayama, Marco Temperini, Tania Di Mascio, Filippo Sciarrone, Veronica Rossano, Ralf Dörner, Loredana Caruccio, Autilia Vitiello, Weidong Huang, Michele Risi, Ugo Erra, Răzvan Andonie, Muhammad Aurangzeb Ahmad, Ana Figueiras, and Mabule Samuel Mabakane, editors, *24th International Conference on Information Visualisation, IV 2020, Melbourne, Australia, September 7-11, 2020*, pages 270–279. IEEE, 2020.

[BDS19] Jill Burstein, Christy Doran, and Thamar Solorio, editors. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019.

[BGLL08] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[BGMS21] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In Madeleine Clare Elish, William Isaac, and Richard S. Zemel, editors, *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 610–623. ACM, 2021.

[BNW18] Adrian M.P. Brasoveanu, Lyndon J.B. Nixon, and Albert Weichselbraun. Storylens: A multiple views corpus for location and event detection. In *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics (WIMS 2018)*, Novi Sad, Serbia, 2018. ACM.

[BNWS16] Adrian M.P. Braşoveanu, Lyndon J. B. Nixon, Albert Weichselbraun, and Arno Scharl. A regional news corpora for contextualized entity discovery and linking. In *Proceedings of the Tenth*

International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016., pages 3333–3338, 2016.

[BP06] Razvan C. Bunescu and Marius Pasca. Using encyclopedic knowledge for named entity disambiguation. In Diana McCarthy and Shuly Wintner, editors, *EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, April 3-7, 2006, Trento, Italy*, pages 9–16. The Association for Computer Linguistics, 2006.

[BRK+18] Adrian M.P. Brasoveanu, Giuseppe Rizzo, Philipp Kuntschick, Albert Weichselbraun, and Lyndon J.B. Nixon. Framing named entity linking error types. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 266–271, Paris, France, may 2018. European Language Resources Association (ELRA).

[BSS+17] Adrian M.P. Brasoveanu, Marta Sabou, Arno Scharl, Alexander Hubmann-Haidvogel, and Daniel Fischl. Visualizing statistical linked knowledge for decision support. *Semantic Web*, 8(1):113–137, 2017.

[BWN20] Adrian M.P. Brasoveanu, Albert Weichselbraun, and Lyndon J. B. Nixon. In media res: A corpus for evaluating named entity linking with creative works. In Raquel Fernández and Tal Linzen, editors, *Proceedings of the 24th Conference on Computational Natural Language Learning, CoNLL 2020, Online, November 19-20, 2020*, pages 355–364. Association for Computational Linguistics, 2020.

[CCK+17] Diego Calvanese, Benjamin Cogrel, Sarah Komla-Ebri, Roman Kontchakov, Davide Lanti, Martin Rezk, Mariano Rodriguez-Muro, and Guohui Xiao. Ontop: Answering SPARQL queries over relational databases. *Semantic Web*, 8(3):471–487, 2017.

[CFC13] Marco Cornolti, Paolo Ferragina, and Massimiliano Ciaramita. A framework for benchmarking entity-annotation systems. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 249–260. International World Wide Web Conferences Steering Committee / ACM, 2013.

[Cho17] Francois Chollet. *Deep Learning with Python*. Manning Publications Co., 2017.

[CKLM19] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of bert's attention. *CoRR*, abs/1906.04341, 2019.

[CLH11] Erik Cambria, Andrew G. Livingstone, and Amir Hussain. The hourglass of emotions. In Anna Esposito, Antonietta Maria Esposito, Alessandro Vinciarelli, Rüdiger Hoffmann, and Vincent C. Müller, editors, *Cognitive Behavioural Systems - COST 2102 International Training School, Dresden, Germany, February 21-26, 2011, Revised Selected Papers*, volume 7403 of *Lecture Notes in Computer Science*, pages 144–157. Springer, 2011.

[CLX+20] Erik Cambria, Yang Li, Frank Z. Xing, Soujanya Poria, and Kenneth Kwok. SenticNet 6: Ensemble Application of Symbolic and Subsymbolic AI for Sentiment Analysis. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 105–114, 2020.

[CN16] Jason P. C. Chiu and Eric Nichols. Named Entity Recognition with Bidirectional LSTM-CNNs. *TACL*, 4:357–370, 2016.

[CNJA19] Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. SemEval-2019 Task 3: EmoContext Contextual Emotion Detection in Text. In *Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, June 6-7, 2019*, pages 39–48, 2019.

[CPGT17] Erik Cambria, Soujanya Poria, Alexander F. Gelbukh, and Mike Thelwall. Sentiment analysis is a big suitcase. *IEEE Intell. Syst.*, 32(6):74–80, 2017.

[CTW+20] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. *CoRR*, abs/2012.07805, 2020.

[DJHM13] Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. Improving efficiency and accuracy in multilingual entity extraction. In *I-SEMANTICS 2013 - 9th International Conference on Semantic Systems, ISEM '13, Graz, Austria, September 4-6, 2013*, pages 121–124. ACM, 2013.

[EER16] Allyson Ettinger, Ahmed Elgohary, and Philip Resnik. Probing for semantic evidence of composition by means of simple classification tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, RepEval@ACL 2016,*

Berlin, Germany, August 2016, pages 134–139. Association for Computational Linguistics, 2016.

[FJP+19] Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. ELI5: long form question answering. In Korhonen et al. [KTM19], pages 3558–3567.

[FK19] Haftu Wedajo Fentaw and Tae-Hyong Kim. Design and investigation of capsule networks for sentence classification. *Applied Sciences*, 9(11):2200, 2019.

[FL19] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *CoRR*, abs/1903.02428, 2019.

[GCL+20] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[GE03] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.

[GvLB+17] Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors. *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017.

[HBC+20] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutiérrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan F. Sequeda, Steffen Staab, and Antoine Zimmermann. Knowledge graphs. *CoRR*, abs/2003.02320, 2020.

[HDWX20] Yaru Hao, Li Dong, Furu Wei, and Ke Xu. Self-attention attribution: Interpreting information interactions inside transformer. *CoRR*, abs/2004.11207, 2020.

[HLAN12] Sebastian Hellmann, Jens Lehmann, Sören Auer, and Marcus Nitzschke. NIF combinator: Combining NLP tool output. In *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, volume 7603 of *Lecture Notes in Computer Science*, pages 446–449. Springer, 2012.

[HM19] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In Burstein et al. [BDS19], pages 4129–4138.

[HNR14] Ben Hachey, Joel Nothman, and Will Radford. Cheap and easy entity evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 464–469. The Association for Computer Linguistics, 2014.

[HSG19] Benjamin Hoover, Hendrik Strobelt, and Sebastian Gehrmann. exbert: A visual analysis tool to explore learned representations in transformers models. *CoRR*, abs/1910.05276, 2019.

[HSN+12] Johannes Hoffart, Stephan Seufert, Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. KORE: keyphrase overlap relatedness for entity disambiguation. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 545–554. ACM, 2012.

[HTF09] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer Series in Statistics. Springer, 2009.

[HYB+11] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 782–792, 2011.

[IJNW19] Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics, 2019.

[ITA+16] Kazuki Irie, Zoltán Tüske, Tamer Alkhouli, Ralf Schlüter, and Hermann Ney. LSTM, GRU, Highway and a Bit of Attention: An Empirical Overview for Language Modeling in Speech Recognition. In Nelson Morgan, editor, *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 3519–3523. ISCA, 2016.

[JdMAJ+20] Annika Jacobsen, Ricardo de Miranda Azevedo, Nick Juty, Dominique Batista, Simon Coles, Ronald Cornet, Mélanie Courtot, Mercè Crosas, Michel Dumontier, Chris T Evelo, et al. Fair principles: interpretations and implementation considerations, 2020.

[JRN17] Kunal Jha, Michael Röder, and Axel-Cyrille Ngonga Ngomo. All that glitters is not gold - rule-based curation of reference datasets for named entity recognition and entity linking. In Eva Blomqvist, Diana Maynard, Aldo Gangemi, Rinke Hoekstra, Pascal Hitzler, and Olaf Hartig, editors, *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 305–320, 2017.

[JW19] Sarthak Jain and Byron C. Wallace. Attention is not explanation. In Burstein et al. [BDS19], pages 3543–3556.

[KAKC18] Minsuk Kahng, Pierre Y. Andrews, Aditya Kalro, and Duen Horng (Polo) Chau. Activis: Visual exploration of industry-scale deep neural network models. *IEEE Trans. Vis. Comput. Graph.*, 24(1):88–97, 2018.

[KB14] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2014.

[Kim14] Yoon Kim. Convolutional neural networks for sentence classification. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751. ACL, 2014.

[KJPC20] Jaeyoung Kim, Sion Jang, Eunjeong L. Park, and Sungchul Choi. Text classification using capsules. *Neurocomputing*, 376:214–221, 2020.

[KTM19] Anna Korhonen, David R. Traum, and Lluís Màrquez, editors. *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Association for Computational Linguistics, 2019.

[LEB12] Günther Lengauer, Frank Esser, and Rosa Berganza. Negativity in Political News: A Review of Concepts, Operationalizations and Key Findings. *Journalism*, 13(2):179–202, 2012.

[LIJ+15] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed

Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[Lis21] Michael Lissack. The slodderwetenschap (sloppy science) of stochastic parrots - A plea for science to NOT take the route advocated by gebru and bender. *CoRR*, abs/2101.10098, 2021.

[LL17] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Guyon et al. [GvLB+17], pages 4765–4774.

[LLX+17] Yunfei Long, Qin Lu, Rong Xiang, Minglei Li, and Chu-Ren Huang. Fake News Detection Through Multi-Perspective Speaker Profiles. In Greg Kondrak and Taro Watanabe, editors, *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017, Volume 2: Short Papers*, pages 252–256. Asian Federation of Natural Language Processing, 2017.

[MCB+20] Giovanni Da San Martino, Stefano Cresci, Alberto Barrón-Cedeño, Seunghak Yu, Roberto Di Pietro, and Preslav Nakov. A survey on computational propaganda detection. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4826–4832. ijcai.org, 2020. Scheduled for July 2020, Yokohama, Japan, postponed due to the Corona pandemic.

[MCZ+17] Yao Ming, Shaozu Cao, Ruixiang Zhang, Zhen Li, Yuanzhe Chen, Yangqiu Song, and Huamin Qu. Understanding hidden memories of recurrent neural networks. In Brian D. Fisher, Shixia Liu, and Tobias Schreck, editors, *12th IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2017, Phoenix, AZ, USA, October 3-6, 2017*, pages 13–24. IEEE Computer Society, 2017.

[MRN14a] Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation: a unified approach. *TACL*, 2:231–244, 2014.

[MRN14b] Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244, 2014.

[MSS+17] Edgard Marx, Saeedeh Shekarpour, Tommaso Soru, Adrian M.P. Brasoveanu, Muhammad Saleem, Ciro Baron, Albert Weichselbraun, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, and Sören

Auer. Torpedo: Improving the state-of-the-art RDF dataset slicing. In *11th IEEE International Conference on Semantic Computing, ICSC 2017, San Diego, CA, USA, January 30 - February 1, 2017*, pages 149–156, San Diego, CA, USA, 2017. IEEE Computer Society.

[NGP+15] Andrea Giovanni Nuzzolese, Anna Lisa Gentile, Valentina Presutti, Aldo Gangemi, Darío Garigliotti, and Roberto Navigli. Open knowledge extraction challenge. In *Semantic Web Evaluation Challenges - Second SemWebEval Challenge at ESWC 2015, Portorož, Slovenia, May 31 - June 4, 2015, Revised Selected Papers*, volume 548 of *Communications in Computer and Information Science*, pages 3–15. Springer, 2015.

[OKBW18] Fabian Odoni, Philipp Kuntschik, Adrian M.P. Brasoveanu, and Albert Weichselbraun. On the importance of drill-down analysis for assessing gold standards and named entity linking performance. In Anna Fensel, Victor de Boer, Tassilo Pellegrini, Elmar Kiesling, Bernhard Haslhofer, Laura Hollink, and Alexander Schindler, editors, *Proceedings of the 14th International Conference on Semantic Systems, SEMANTICS 2018, Vienna, Austria, September 10-13, 2018*, volume 137 of *Procedia Computer Science*, pages 33–42. Elsevier, 2018.

[OMK21] Daniel W. Otter, Julian R. Medina, and Jugal K. Kalita. A survey of the usages of deep learning for natural language processing. *IEEE Trans. Neural Networks Learn. Syst.*, 32(2):604–624, 2021.

[PCLG21] Deepak P, Tanmoy Chakraborty, Cheng Long, and Santhosh Kumar G. *Data Science for Fake News - Surveys and Perspectives*, volume 42 of *The Information Retrieval Series*. Springer, 2021.

[PHM+19] Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 527–536, 2019.

[PRT16] Julien Plu, Giuseppe Rizzo, and Raphaël Troncy. Enhancing entity linking by combining NER models. In Harald Sack, Stefan Dietze, Anna Tordai, and Christoph Lange, editors, *Semantic Web Challenges - Third SemWebEval Challenge at ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Revised Selected Papers*, volume 641 of *Communications in Computer and Information Science*, pages 17–32. Springer, 2016.

[PZS+20] Beatrice Portelli, Jason Zhao, Tal Schuster, Giuseppe Serra, and Enrico Santus. Distilling the evidence to augment fact verification models. In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 47–51, 2020.

[QSF+18] Ye Qi, Devendra Singh Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. When and why are pre-trained word embeddings useful for neural machine translation? In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 529–535. Association for Computational Linguistics, 2018.

[Rad15] Will Radford. *Linking Named Entities to Wikipedia*. PhD thesis, School of Information Technologies, Faculty of Engineering and IT, The University of Sydney, 2015.

[RCJ+17] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. Truth of varying shades: Analyzing language in fake news and political fact-checking. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2931–2937. Association for Computational Linguistics, 2017.

[RS20] Maithra Raghu and Eric Schmidt. A survey of deep learning for scientific discovery. *CoRR*, abs/2003.11755, 2020.

[RSG16] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi, editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144. ACM, 2016.

[RUH+14] Michael Röder, Ricardo Usbeck, Sebastian Hellmann, Daniel Gerber, and Andreas Both. N^3 - A collection of datasets for named entity recognition and disambiguation in the NLP interchange format. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*,

pages 3529–3533. European Language Resources Association (ELRA), 2014.

[RUN18] Michael Röder, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. GERBIL - benchmarking named entity recognition and linking consistently. *Semantic Web*, 9(5):605–625, 2018.

[RYW+19] Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B. Viégas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and measuring the geometry of BERT. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 8592–8600, 2019.

[SBF+18] Abeed Sarker, Maksim Belousov, Jasper Friedrichs, Kai Hakala, Svetlana Kiritchenko, Farrokh Mehryary, Sifei Han, Tung Tran, Anthony Rios, Ramakanth Kavuluru, Berry de Bruijn, Filip Ginter, Debanjan Mahata, Saif M. Mohammad, Goran Nenadic, and Graciela Gonzalez-Hernandez. Data and systems for medication-related text classification and concept normalization from twitter: insights from the social media mining for health (SMM4H)-2017 shared task. *Journal of American Medical Informatics Association*, 25(10):1274–1283, 2018.

[SBÖ15] Marta Sabou, Adrian M.P. Brasoveanu, and Irem Önder. Linked data for cross-domain decision-making in tourism. In Iis Tussyadiah and Alessandro Inversini, editors, *Information and Communication Technologies in Tourism 2015, ENTER 2015, Proceedings of the International Conference in Lugano, Switzerland, February 3 - 6, 2015*, pages 197–210. Springer, 2015.

[SCH17] Robyn Speer, Joshua Chin, and Catherine Havasi. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451, 2017.

[SES+20] Blaz Skrlj, Nika Erzen, Shane Sheehan, Saturnino Luz, Marko Robnik-Sikonja, and Senja Pollak. Attviz: Online exploration of self-attention for transparent neural language modeling. *CoRR*, abs/2005.05716, 2020.

[SGM19] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational*

Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, pages 3645–3650. Association for Computational Linguistics, 2019.

[SGPR18] Hendrik Strobelt, Sebastian Gehrmann, Hanspeter Pfister, and Alexander M. Rush. Lstmvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks. *IEEE Trans. Vis. Comput. Graph.*, 24(1):667–676, 2018.

[SHJ+20] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling LIME and SHAP: adversarial attacks on post hoc explanation methods. In Annette N. Markham, Julia Powles, Toby Walsh, and Anne L. Washington, editors, *AIES '20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020*, pages 180–186. ACM, 2020.

[SHY+20] Lichao Sun, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip S. Yu, and Caiming Xiong. Adv-bert: BERT is not robust on misspellings! generating nature adversarial samples on BERT. *CoRR*, abs/2003.04985, 2020.

[SLCC20] Yosephine Susanto, Andrew G. Livingstone, Ng Bee Chin, and Erik Cambria. The hourglass model revisited. *IEEE Intell. Syst.*, 35(5):96–102, 2020.

[SLH+18] Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. CARER: Contextualized Affect Representations for Emotion Recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3687–3697, 2018.

[SM03] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL, 2003.

[SOBS16] Marta Sabou, Irem Onder, Adrian M.P. Brasoveanu, and Arno Scharl. Towards cross-domain data analytics in tourism: a linked data based approach. *J. Inf. Technol. Tour.*, 16(1):71–101, 2016.

[SZC+20] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. VL-BERT: pre-training of generic visual-linguistic representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[TDP19] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In Korhonen et al. [KTM19], pages 4593–4601.

[TVCM18] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and verification. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 809–819. Association for Computational Linguistics, 2018.

[vEMP+16] Marieke van Erp, Pablo N. Mendes, Heiko Paulheim, Filip Ilievski, Julien Plu, Giuseppe Rizzo, and Jörg Waitelonis. Evaluating entity linking: An analysis of current benchmark datasets and a roadmap for doing a better job. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.*, pages 4373–4379, 2016.

[Vig19] Jesse Vig. A multiscale visualization of attention in the transformer model. In Marta R. Costa-jussà and Enrique Alfonseca, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations*, pages 37–42. Association for Computational Linguistics, 2019.

[Vra13] Denny Vrandecic. The rise of wikidata. *IEEE Intelligent Systems*, 28(4):90–95, 2013.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Guyon et al. [GvLB+17], pages 5998–6008.

[VST19] Elena Voita, Rico Sennrich, and Ivan Titov. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. In Inui et al. [IJNW19], pages 4395–4405.

[VT20] Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. *CoRR*, abs/2003.12298, 2020.

[VTM+19] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In Korhonen et al. [KTM19], pages 5797–5808.

[Wan17] William Yang Wang. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. *CoRR*, abs/1705.00648, 2017.

[WBKN19] Albert Weichselbraun, Adrian M.P. Braşoveanu, Philipp Kuntschik, and Lyndon J.B. Nixon. Improving named entity linking corpora quality. *RANLP 2019*, pages 1328–1337, 2019.

[WBWO21] Albert Weichselbraun, Adrian M.P. Brasoveanu, Roger Waldvogel, and Fabian Odoni. Harvest - an open source toolkit for extracting posts and post metadata from web forums. *CoRR*, abs/2102.02240, 2021.

[WFK+19] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. Universal adversarial triggers for attacking and analyzing NLP. In Inui et al. [IJNW19], pages 2153–2162.

[WKB18] Albert Weichselbraun, Philipp Kuntschik, and Adrian M.P. Brasoveanu. Mining and leveraging background knowledge for improving named entity linking. In *Proceedins of the 8th International Conference on Web Intelligence, Mining and Semantics (WIMS 2018)*, pages 27:1–27:11, Novi Sad, Serbia, 2018. ACM.

[WKB19] Albert Weichselbraun, Philipp Kuntschik, and Adrian M.P. Brasoveanu. Name variants for improving entity discovery and linking. In Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski, editors, *2nd Conference on Language, Data and Knowledge, LDK 2019, May 20-23, 2019, Leipzig, Germany.*, volume 70 of *OASICS*, pages 14:1–14:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2019.

[Wöb03] Karl W Wöber. Information supply in tourism management by marketing decision support systems. *Tourism Management*, 24(3):241–255, 2003.

[WP19] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In Inui et al. [IJNW19], pages 11–20.

[WSB+21] Albert Weichselbraun, Jakob Steixner, Adrian MP Braşoveanu, Arno Scharl, Max Göbel, and Lyndon JB Nixon. Automatic expansion of domain-specific affective models for web intelligence applications. *Cognitive Computation*, pages 1–18, 2021.

[YCOL21] Zeyu Yun, Yubei Chen, Bruno A. Olshausen, and Yann LeCun. Transformer visualization via dictionary learning: contextualized embedding as a linear superposition of transformer factors. *CoRR*, abs/2103.15949, 2021.

[YHPC18] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing [review article]. *IEEE Comp. Int. Mag.*, 13(3):55–75, 2018.

List of Publications

Journal Articles

1. Weichselbraun, A., Steixner, J., **Braşoveanu, A.M.P.**, Scharl, A., Gobel, M., Nixon, L.B. (2021). Automatic Expansion of Domain-Specific Affective Models for Web Intelligence Applications. Cognitive Computation 13. DOI: 10.1007/s12559-021-09839-4. (**IF=4.307**)

2. **Braşoveanu, A.M.P.**, Andonie, R. (2020). Integrating Machine Learning Techniques in Semantic Fake News Detection. Neural Processing Letters, 1-18. DOI: 10.1007/s11063-020-10365-x. (**IF=2.891**)

3. **Braşoveanu, A.M.P.**, M. Sabou, Scharl, A. Hubmann-Haidvogel, A., Fischl, D. (2017). Visualizing Statistical Linked Knowledge for Decision Support. Semantic Web, 8.1, pp. 113137. DOI: 10.3233/SW-160225. (**IF=3.524**)

4. Sabou, M., Onder, I., **Braşoveanu, A.M.P.**, Scharl, A. (2016). Towards Cross-Domain Data Analytics in Tourism: A Linked Data Based Approach. J. of IT & Tourism 16.1, pp. 71-101. DOI: 10.1007/s40558-015-0049-5. (**IF=2.95**)

Articles in Proceedings

5. **Braşoveanu, A.M.P.**, Weichselbraun, A., Nixon, L. (2020). In Media Res: A Corpus for Evaluating Named Entity Linking with Creative Works. In Proceedings of the 24th Conference on Computational Natural Language Learning, ACL, pp. 355-364. DOI: 10.18653/v1/2020.conll-1.28.

6. Weichselbraun, A., **Braşoveanu, A.M.P.**, Waldvogel, R., Odoni, F. (2021). Harvest - An Open Source Toolkit for Extracting Posts and Post Metadata from Web Forums. IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, Melbourne, Australia, IEEE/ACM, pp. 430-436. DOI: https://arxiv.org/abs/2102.02240 (arXiv only).

7. **Braşoveanu, A.M.P.**, Andonie, R. (2020) Visualizing Transformers for NLP: A Brief Survey. 24th International Conference on Information Visualisation, IV 2020, Melbourne, Australia, IEEE, pp. 270-279. DOI: 10.1109/IV51561.2020.00051.

8. Weichselbraun, A., **Braşoveanu, A.M.P.**, Kuntschik, P., Nixon, L.J.B. (2019). Improving Named Entity Linking Corpora Quality. RANLP 2019, Varna, Bulgaria, Incoma, pp. 1329-1338. DOI: 10.26615/978-954-452-056-4_152.

9. Weichselbraun, A., Kuntschik, P., **Braşoveanu, A.M.P.** (2019). Name Variants for Improving Entity Discovery and Linking. LDK 2019, Leipzig, Germany. OASICS 70, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik 2019, pp. 14:1-14:15. DOI: 10.4230/OASIcs.LDK.2019.14.

10. **Braşoveanu, A.M.P.**, Andonie, R. (2019). Semantic Fake News Detection: A Machine Learning Perspective. IWANN 2019 Part I, Springer, pp. 656-667. DOI: 10.1007/978-3-030-20521-8_54.

11. Odoni, F., **Braşoveanu, A.M.P.**, Kuntschik, P., Weichselbraun, A. (2019). Introducing orbis: An extendable evaluation pipeline for named entity linking performance drilldown analyses. Proceedings of the Association for Information Science and Technology, 56(1), 468-471. DOI: 10.1002/pra2.49.

12. Odoni, F., Kuntschik, P., **Braşoveanu, A.M.P.**, Weichselbraun, A. (2018). On the Importance of Drill-Down Analysis for Assessing Gold Standards and Named Entity Linking Performance. Semantics 2018, In Procedia Computer Science, 137, 33-42. Elsevier. DOI: 10.1016/j.procs.2018.09.004.

13. **Braşoveanu, A.M.P.**, Rizzo, G., Kuntschik, P., Weichselbraun, A., Nixon, L. (2018). Framing Named Entity Linking Error Types. LREC 2018, Miyzaki, Japan, pp. 266-271. ELRA, Paris, France. DOI: www.lrec-conf.org/proceedings/lrec2018/pdf/612.pdf.

14. Weichselbraun, A., Kuntschik, P., **Braşoveanu, A.M.P.** (2018). Mining and Leveraging Background Knowledge for Improving Named Entity Linking. WIMS 2018, Novi Sad, Serbia, ACM, pp. 27:1-27:11. DOI: 10.1145/3227609.3227670.

15. **Braşoveanu, A.M.P.**, Nixon, L. J., Weichselbraun, A. (2018). StoryLens: A multiple views corpus for location and event detection. WIMS 2018, Novi Sad, Serbia, ACM, pp. 30:1-30:4. DOI: 10.1145/3227609.3227674.

16. Marx, E., Sherkarpour, S., Soru, T., **Braşoveanu, A.M.P.**, Saleem, M., Baron, C., Weichselbraun, A., Lehmann, J., Ngonga Ngomo, A.-C., Auer, S. (2017). Torpedo: Improving the State-of-the-Art RDF Dataset

Slicing. ICSC 2017, San Diego, California, IEEE, pp. 149-156. DOI: 10.1109/ICSC.2017.79.

17. **Braşoveanu, A.M.P.**, L.J.B. Nixon, A. Weichselbraun, A. Scharl. A Regional News Corpora for Contextualized Entity Discovery and Linking. LREC 2016, Portoroz, Slovenia, ELRA, pp. 3333-3338. DOI: www.lrec-conf.org/proceedings/lrec2016/summaries/835.html.

18. Sabou, M., **Braşoveanu, A.M.P.**, Onder, I. (2015). Linked Data for Cross-Domain Decision-Making in Tourism. ENTER 2015, Springer, pp. 197-210. DOI: 10.1007/978-3-319-14343-9_15.

Short Abstract - Scurt Rezumat

Abstract. Semantic AI is a recent approach towards AI that is focused on combining semantics with classic AI methods like classification or clustering. By adding semantics, we can increase data quality while removing black-box approaches. Its core proposition is that regardless of its original provenance, data can be processed and stored into refined formats like those provided by knowledge graphs or search engines. These open data clusters can later be used to solve complex problems with hybrid approaches. By combining entities extracted from a KG with sentiment and ML classifiers, it is possible to verify the claims from a sentence, for example. This thesis examines several hybrid methods enabled by SAI to understand how to leverage them to build baselines for research and production. Once these methods are examined, it emerges that each component may add its errors to the stack and confuse the researchers and developers. It then argues that to move forward, it is important to build some practical solutions like a taxonomy of errors or a tool for visualizing benchmarking results, to help researchers navigate this complexity.

Rezumat. IA semantică este o abordare recentă de IA prin care se combină semantica şi metodele clasice de IA, cum ar fi clasificarea sau clusterizarea. Adăugând semantică, putem creşte calitatea datelor, eliminând în acelaşi timp abordările de tip cutie neagră. Propunerea ei de bază este că, indiferent de provenienţa originală, datele pot fi procesate şi stocate în formate rafinate, precum cele furnizate de reţele semantice sau motoare de căutare. Aceste clustere de date pot fi utilizate ulterior pentru a rezolva probleme complexe cu abordări hibride. Combinând entităţi extrase dintr-o reţea semantică cu analiza sentimentului şi clasificatori bazaţi pe învăţare automată, este posibil să se verifice afirmaţiile dintr-o propoziţie, de exemplu. Această teză examinează mai multe metode hibride propuse de IA semantică pentru a înţelege cum să le folosească pentru a construi linii de bază pentru cercetare şi producţie. Odată examinate aceste metode, rezultă că fiecare componentă îşi poate adăuga erorile în stivă şi poate deruta cercetătorii şi dezvoltatorii. Apoi argumentează că pentru a merge mai departe, este important să construim câteva soluţii practice, cum ar fi o taxonomie a erorilor sau un instrument pentru vizualizarea rezultatelor evaluărilor, pentru a ajuta cercetătorii să navigheze această complexitate.

Adrian M.P. Braşoveanu, M.Sc. - CV

Main Areas of Research	Natural Language Processing (NLP) Semantic Web (SW) and Knowledge Graphs (KG) Machine Learning (ML) Information Visualization (IV)

Education

2014-Present	**PhD Student**, Transylvania University, Braşov, România. Domain: Computer Science. Thesis topic: *Intelligent Systems in Semantic Networks*.
2008	**MSc in Distributed and Parallel Processing Systems (Dipl.-Ing.),** Lucian Blaga University of Sibiu, Sibiu, România
2007	**MSc in Computer Science and Automatic Control (Dipl.-Ing.),** Lucian Blaga University of Sibiu, Sibiu, România

Work Experience

2020.11-Present	**Researcher**, MODUL University Vienna, Austria
2018.11- Present	**Researcher**, MODUL Technology GmbH, Vienna, Austria
2017.11-2018.10	**Invited Researcher**, University of Applied Sciences of the Grisons (UASG), Chur, Switzerland
2016.06-2017.10	**Researcher**, MODUL Technology GmbH, Vienna, Austria.
2011.11- 2016.05	**Researcher,** MODUL University Vienna, Austria.
2011.03- 2011.10	**Intern**, MODUL University, Austria
2007.09- 2011.02	**Java Programmer**, em2Soft, Sibiu, Romania.

Key International Cooperation Partners

LINKS	LINKS Foundations, Torino (Italy), Senior Researcher Giuseppe Rizzo
AKSW	AKSW Leipzig (Germany), Senior Researcher Milan Dojchinovski
DICE	DICE Lab at Paderborn University (Germany), Prof. Axel-Cyrille Ngonga Ngomo

Most Important Research Projects

ReTV	European Horizon 2020 Programme (Researcher)
InVID	European Horizon 2020 Programme (Researcher)
ASAP	EU 7th Framework Program (Researcher)
LinkedTV	EU 7th Framework Program (Researcher)
ETIHQ	A PlanetData WP - EU 7th Framework Program (Researcher)

Awards

ISWC 2017	**Best Reviewer Award** at the **International Semantic Web Conference 2017 (ISWC 2017)**
IEEE ICSC 2017	**Honorable Mention Award**, Awarded for E. Marx, S. Sherkarpour, T. Soru, **A.M.P. Braşoveanu**, M. Saleem, C. Baron, A. Weichselbraun, J. Lehmann, A.-C. Ngonga Ngomo, S. Auer. Torpedo: Improving the State-of-the-Art RDF Dataset Slicing.

List of Publications

Journal Articles

- Weichselbraun, A., Steixner, J., **Braşoveanu, A.M.P.**, Scharl, A., Gobel, M., Nixon, L.B.J. (2021). Automatic Expansion of Domain-Specific Affective Models for Web Intelligence Applications. *Cognitive Computation* 13. 10.1007/s12559-021-09839-4.
- **Braşoveanu, A.M.P.**, Andonie, R. (2020). Integrating Machine Learning Techniques in Semantic Fake News Detection. Neural Processing Letters, 1-18. DOI: 10.1007/s11063-020-10365-x.
- **Braşoveanu, A.M.P.**, Sabou, M., Scharl, A., Hubmann-Haidvogel, A., Fischl, D. (2017). Visualizing Statistical Linked Knowledge for Decision Support. Semantic Web 8(1), pp. 113–137. DOI: 10.3233/SW-160225.
- Sabou, M., Onder, I., **Braşoveanu, A.M.P.**, Scharl, A (2016). Towards Cross-Domain Data Analytics in Tourism: A Linked Data Based Approach. J. of IT & Tourism 16(1), pp. 71-101. DOI: 10.1007/s40558-015-0049-5.
- Sabou, M., Arsal, I., **Braşoveanu, A.M.P.** (2013). TourMISLOD: A tourism linked data set. Semantic Web 4(3), pp. 271-276. DOI: 10.3233/SW-2012-0087.
- **Braşoveanu, A.M.P**, Dzitac, I (2012). The Role of Visual Rhetoric in Semantic Multimedia: Strategies for Decision Making in Times of Crisis. Int. J. Comput. Commun. Control, 7(4), pp. 606-616. DOI: 10.15837/ijccc.2012.4.1361.
- **Braşoveanu, A.M.P.**, Nagy, M, Mateut-Petrisor, O., Urziceanu, R. (2010). The Avatar in the Context of Intelligent Social Semantic Web. Int. J. Comput. Commun. Control, 5(4), pp. 477-482. DOI: 10.15837/ijccc.2010.4.2497.
- **Braşoveanu, A.M.P.**, Manolescu, A., Spinu, M.N. (2010). Generic Multimodal Ontologies for Human-Agent Interaction. Int. J. Comput. Commun. Control, 5(4), pp. 625-633. DOI: 10.15837/ijccc.2010.5.2218.

Conference Articles

- **Braşoveanu, A.M.P.**, Weichselbraun, A., Nixon, L.J.B. (2020). In Media Res: A Corpus for Evaluating Named Entity Linking with Creative Works. In Proceedings of the 24th Conference on Computational Natural Language Learning (pp. 355-364). DOI: 10.18653/v1/2020.conll-1.28
- Weichselbraun, A., **Braşoveanu, A.M.P.**, Waldvogel, R., Odoni, F. (2020). Harvest - An Open Source Toolkit for Extracting Posts and Post Metadata from Web Forums. IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology: 430-436.
- **Braşoveanu, A.M.P.**, Andonie, R. (2020) Visualizing Transformers for NLP: A Brief Survey. IV 2020: 270-279. DOI: 10.1109/IV51561.2020.00051.
- Weichselbraun, A., **Braşoveanu, A.M.P.**, Kuntschik, P., Nixon, L.J.B. (2019). Improving Named Entity Linking Corpora Quality. RANLP 2019, Varna, Bulgaria. Incoma. Published by ACL. pp. 1329-1338. DOI: 10.26615/978-954-452-056-4_152.
- Weichselbraun, A., Kuntschik, P., **Braşoveanu, A.M.P.** (2019). Name Variants for Improving Entity Discovery and Linking. LDK 2019. OASICS 70, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik 2019, pp. 14:1-14:15. DOI: 10.4230/OASIcs.LDK.2019.14.
- **Braşoveanu, A.M.P.,** Andonie, R. (2019). Semantic Fake News Detection: A Machine Learning Perspective. IWANN 2019 Part I, Springer, pp. 656-667. DOI: 10.1007/978-3-030-20521-8_54.
- Odoni, F., **Braşoveanu, A.M.P.**, Kuntschik, P., Weichselbraun, A. (2019). Introducing orbis: An extendable evaluation pipeline for named entity linking performance drill-down analyses. Proceedings of the Association for Information Science and Technology, 56(1), 468-471. DOI: 10.1002/pra2.49.
- Odoni, F., Kuntschik, P., **Braşoveanu, A.M.P.**, Weichselbraun, A. (2018). On the Importance of Drill-Down Analysis for Assessing Gold Standards and Named Entity Linking Performance. Semantics 2018, Procedia Computer Science, 137, 33-42. Elsevier. DOI: 10.1016/j.procs.2018.09.004.
- **Brasoveanu, A.M.P.**, Rizzo, G., Kuntschik, P., Weichselbraun, A., Nixon, L.J.B. (2018). Framing Named Entity Linking Error Types. LREC 2018, pp. 266-271. ELRA, Paris, France. DOI: www.lrec-conf.org/proceedings/lrec2018/pdf/612.pdf
- Weichselbraun, A., Kuntschik, P., **Braşoveanu, A.M.P.** (2018). Mining and Leveraging Background Knowledge for Improving Named Entity Linking. WIMS 2018, pp. 27:1-27:11. ACM. DOI: 10.1145/3227609.3227670.

- **Braşoveanu, A.M.P.**, Nixon, L.J.B., Weichselbraun, A. (2018). Storylens: A multiple views corpus for location and event detection. WIMS 2018, pp. 30:1-30:4. ACM. DOI: 10.1145/3227609.3227674.
- Marx, E., Sherkarpour, S., Soru, T., **Braşoveanu, A.M.P.**, Saleem, M., Baron, C., Weichselbraun, A., Lehmann, J.,. Ngonga Ngomo, A.-C., Auer, S. (2017). Torpedo: Improving the State-of-the-Art RDF Dataset Slicing. ICSC 2017, pp. 149-156. IEEE.DOI: 10.1109/ICSC.2017.79.
- **Braşoveanu, A.M.P.**, Nixon, L.J.B. Weichselbraun, A., & Scharl, A. (2017). A Regional News Corpora for Contextualized Entity Discovery and Linking. LREC 2016, pp. 3333-3338. ELRA, Paris, France. DOI: www.lrec-conf.org/proceedings/lrec2016/summaries/835.html.
- Sabou, M., **Braşoveanu, A.M.P.**, Onder, I. (2015). Linked Data for Cross-Domain Decision-Making in Tourism. ENTER 2015, pp. 197-210. DOI: 10.1007/978-3-319-14343-9_15.
- **Braşoveanu, A.M.P.**, Hubmann-Haidvogel, A., Scharl, A. (2012). Interactive visualization of emerging topics in multiple social media streams. AVI 2012, pp. 530-533. DOI: 10.1145/2254556.2254655.
- Sabou, M., **Braşoveanu, A.M.P.**, Arsal, I. (2012) Supporting tourism decision-making with linked data. I-SEMANTICS 2012, pp. 201-204. DOI: 10.1145/2362499.2362533.
- Hubmann-Haidvogel, A., **Braşoveanu, A.M.P.**, Scharl, A., Sabou, M., Gindl, S. (2012). Visualizing Contextual and Dynamic Features of Micropost Streams. MSM 2012, pp. 34-40. DOI: http://ceur-ws.org/Vol-838/paper_05.pdf.
- Oprean, C., Kifor, C., Barbat, B.E., **Braşoveanu, A.M.P.**, Fabian, R.D. (2010). Bounded Rationality in Computer Science Curricula. FECS 2010, pp. 135-140.