Funded by the European Union

UK Research and Innovation

Deliverable number: D1.1

hybrids

Technical report on the state of the art on hybrid methods in NLP

Charting the Evolution of NLP: A Focus on Hybrid Methods

Version 1.

Project Details

Project Acronym:	HYBRIDS
Project Title:	Hybrid Intelligence to monitor, promote and analyse transformations in good democracy practices
Grant Number:	101073351
Call	HORIZON-MSCA-2021-DN-01
Topic:	HORIZON-MSCA-2021-DN-01-01
Type of Action:	HORIZON-TMA-MSCA-DN
Project website:	https://hybridsproject.eu/
Coordinator	Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS)-Universidade de Santiago de Compostela (USC)
Main scientific representative:	Prof. Pablo Gamallo Otero, pablo.gamallo@usc.es
E-mail:	citius.kmt@usc.es, info@hybridsproject.eu
Phone:	+34 881 816 414

Deliverables Details

Number:	D1.1
Title:	Technical report on the state of the art on hybrid methods in NLP
Work Package	WP1: Hybrid Intelligence
Lead beneficiary:	UEVORA
Deliverable nature:	R- Document, report
Dissemination level:	PU-Public
Due Date (month):	31/12/2023 (M12)
Submission Date (month):	30/12/2023 (M12)
Keywords:	Natural Language Processing; Hybrid Methods; Machine Learning; Symbolic Approaches; Societal Impact; Misinformation

Abstract

D1.1. Technical Report on the State of the Art on Hybrid Methods in NLP offers a comprehensive analysis of the dynamic landscape of Natural Language Processing (NLP) within the context of Artificial Intelligence (AI). It examines the transition from rule-based to deep learning approaches and advocates for hybrid methods. These hybrid approaches blend human and machine intelligibility, enhancing NLP system capabilities. The report emphasizes the societal impact of NLP, including misinformation and online toxicity. It provides crucial insights for researchers and practitioners, presenting a roadmap for navigating the dynamic world of NLP with a focus on hybrid methodologies.

Deliverable Contributors

	Name	Institution	E-mail
Deliverable leader	Renata Vieira	UEVORA	renatav@uevora.pt
Contributing Authors	Michele Joshua Maggini	CiTIUS-USC	michele.cafagna@um.edu.mt
	Paloma Piot Pérez-Abadín	UDC	paloma.piot@udc.es
	Rabiraj Bandyopadhyay	GESIS	rabiraj.Bandyopadhyay@gesis.org
	Rafael Martins Frade	NEWTRAL	rafael.martins@newtral.es
	Rrubaa Panchendrarajan	QMUL	r.panchendrarajan@qmul.ac.uk
	Søren Fomsgaard	UNICAEN	soren.fomsgaard@unicaen.fr
Reviewers	Arkaitz Zubiaga	QMUL	a.zubiaga@qmul.ac.uk
	Pablo Gamallo Otero	CiTIUS, USC	pablo.gamallo@usc.es
	Martín Pereira Fariña	USC	martin.pereira@usc.gal
	Nelleke Oostdijk	RU	nelleke.oostdijk@ru.nl
	Patricia Martín Rodilla	UDC	patricia.martin.rodilla@udc.es
	Javier Parapar López	UDC	javier.parapar@udc.es
	Claudia Wagner	GESIS	Claudia.Wagner@gesis.org
	Rubén Miguez	NEWTRAL	ruben.miguez@newtral.es
	Gaël Dias	UNICAEN	gael.dias@unicaen.fr
	Sara Tonelli	FBK	satonelli@fbk.eu

History of Changes

Version	Date	Changes to previous version	Status
0.1	06/12/2023	First Draft	Draft
0.2	12/12/2023	Consortium Internal review	Review
1.0	30/12/2023	Approved version to be submitted	Final

List of Acronyms

AEDA	An Easier Data Augmentation
AL	Active Learning
AI	Artificial Intelligence
AMR	Abstract Meaning Representation
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
DRS	Discourse Representation Structures
DL	Deep Learning
EDA	Easy Data Augmentation
ERNIE	Enhanced Language RepresentatioN with Informative Entities
GNN	Graph Convolution Networks
KG	Knowledge Graph
LDA	Latent Dirichlet Allocation
LLM	Large Language Model
LSTM	Long Short-Term Memory Network
ML	Machine Learning
NLG	Natural Language Generation
NLP	Natural Language Processing
NLR	Natural Language Reasoning
NLU	Natural Language Understanding
NMT	Neural Machine Translation
OpenIE	Open Information Extraction
QA	Question Answering
RNN	Recurrent Neural Network
ROC	Receiver operating characteristic
RST	Rhetorical Structure Theory
SVM	Support Vector Machine
TF-IDF	Term Frequency - Inverse Document Frequency

Contents

1 Introduction

Recent years have seen a rapid and sweeping transformation of the research and application of Artificial Intelligence. In accordance with this, the research and application of Natural Language Processing, Understanding, Generation, and Reasoning has also evolved. The focus of this report is on detailing the developments and challenges mainly within the field of NLP, although advances and techniques in the remaining fields just mentioned will also be touched upon. This development has occurred both in terms of the data available, and the level of modeling and compute capacities necessary for the research and development of systems that analyze or produce natural language and which are driven by artificial intelligence models. In terms of application, this transformation has found applications both in academic research and in the industry - in the private as well as the public sector.

There is now more data, compute power and off-the-shelf toolkits for machine learning and NLP readily available for more actors (both in research and the industry) than ever before in the past. However, just as techniques and methodologies have evolved over time since the infancy of the fields of Artificial Intelligence and Computational Linguistics in the middle of the 20th century, so have the practical applications, their societal implications, changed. These two aspects of change in the field of NLP are what motivates this report. On the one hand, there is a need to investigate and improve upon current methodologies in NLP. On the other hand, there is a need to address the societal harms and benefits that these technologies have introduced and will likely introduce in the future. In natural language processing, the dominant approaches from the infancy of AI in the 1950s to the 1980s were largely rule-based and heuristically-oriented and are typically referred to as 'symbolic' (after the symbolic AI paradigm). While still popular in certain areas of NLP to this day, symbolic approaches gradually gave way to statistical machine learning and more recently deep-learning approaches in terms of popularity and performance in many NLP tasks. Because of some of the limitations of these paradigms, there is a growing need to improve upon methods in NLP that are hybrid (more on this in a moment). From a societal perspective, and in a European context, for example, the general contemporary issues of misinformation, toxic or hateful online language, complex political discourse and argumentation all translate to specific pressing issues of the spread of misinformation about immigration, climate change and EU skepsis, and online harassment and persecution of particular marginalized groups. The impact that these issues have on political discussion and argumentation requires tools that can verify claims, detect bots and hyperpartisan news in political online discourse, as well as trace and untangle harassment in social spaces online.

The subsequent sections will in turn provide an overview of the various approaches to NLP, starting with machine learning (section 2) and symbolic approaches (section 3). In due turn, the challenges and shortcomings of these approaches will be detailed. Because of the challenges associated with these approaches, section 4 will examine and make the case for adopting a hybrid approach to NLP.

Section 5 will detail approaches specifically related to text classification tasks, since our project has many challenges related to detection of certain classes of texts involving toxicity, harassment, xenophobia, or specific types of news such as hyper-partisan political news.

The concept of 'hybrid' in the context of NLP methodologies is sometimes used to refer to different aspects of an NLP system. The first is in terms of *modeling*, where a system can be said to be hybrid in the sense that it incorporates different types of model architectures (for example, rule-based and deep learning-based components). The second is in terms of *representation*, whereby a system can be said to be hybrid in the way that it represents its input in ways that are intelligible both to humans and to machines (e.g., by combining feature extraction with embeddings of different kinds). The third and final sense is in terms of *learning and/or inference*, whereby a system can be said to be hybrid in that there are humans “in the loop” or “on the loop” at some stage of the design of the NLP system, as opposed to being “out of the loop” entirely. By ‘hybrid approach’ we mean any approach to NLP systems and tasks, in the latter two senses just mentioned. We therefore consider a given approach to be hybrid if it integrates information from both human and machine-intelligible modalities, and which may include humans during learning and decision-making.

2 Machine Learning Approaches to Natural Language Processing

In recent times Machine Learning as a sub-field of Artificial Intelligence has enjoyed widespread adoption in solving real world challenges. Machine Learning, unlike rule based systems, learn relationships between input data and its corresponding labels (a task known as supervised learning) or learns to group together unlabeled data points (a task known as unsupervised learning). Recently, Deep Learning [Schmidhuber, 2014] which is based on Neural Networks has been adopted to solve a variety of supervised and unsupervised learning problems, especially in working with unstructured data: mainly images and text. Deep Learning gained widespread adoption in image tasks with the launch of ResNet [He et al., 2015] which scored the highest on the ImageNet dataset [Deng et al., 2009] and handwritten digit recognition [LeCun et al., 1989] which introduced the concept of Convolutional Neural Networks.

Deep Learning has achieved remarkable success in Natural language Processing as well, especially in text classification, language generation, machine translation tasks. Deep learning was first used for language modeling by [Bengio et al., 2003], which introduced the concept of distributed representations of words (thought of as a vector representation of words, a concept which was further improved upon by [Mikolov et al., 2013a, Mikolov et al., 2013b]). Recurrent Neural Networks (RNNs) [Rumelhart et al., 1986a] became popular with language modeling tasks as well. These networks use a gradient descent algorithm to train its parameters and the gradients are calculated using Backpropagation Through Time (BPTT) algorithm introduced in [Mozer, 1995]. Variants of Recurrent Neural Networks were also introduced namely the Long Short Term Memory(LSTM) [Hochreiter and Schmidhuber, 1997] and Gated Recurrent Unit (GRU) [Cho et al., 2014]. These architectures have also been adopted for language modeling tasks to solve the inherent problem of Vanishing Gradients associated with RNNs. For a long time RNNs and its variants were considered standard models in NLP, but it was the introduction of the attention mechanism [Bahdanau et al., 2016] and the development of Transformer architecture [Vaswani et al., 2017a] and subsequent innovations that optimized calculation of self-attention (self-attention can be thought of as numerically quantizing the inter-token relationship in text), notable among them being Sparse attention [Child et al., 2019] and Flash Attention [Dao et al., 2022] led to model scaling and enabled the training of billion-parameter scale models on a lot of data. Transformers have now become the de-facto architecture that is used to build language models.

2.1 Text Classification

Classification of text data is one of the most popular tasks in both industry and academia, it aids in building systems that can help in product categorization [Krishnan and Amarthaluri, 2019], email classification, identifying fraudulent transactions, and also preventing harmful content in social media.

A text classification task usually consists of the following steps:

1. Text Encoding / Feature Extraction
2. Classification

3. Evaluation

2.1.1 Text Encoding

Text Encoding is the first step in any text classification pipeline, this step consists of transforming text into representations that can be used by the Machine Learning model to classify text to its appropriate labels. Before the advent of Deep Learning architectures text was usually translated to features using **Bag of Words**, which used counts of words in a text data to represent it. This was improved upon by introduction of **Term Frequency-Inverse Document Frequency** which introduced the concept of Inverse Document Frequency, in assigning weight to a word in a single unit of text data. The concept of word embedding was introduced in [Bengio et al., 2003]. This paper calculated the distributed vectorial representation of word w_t by optimizing the language modeling objective $p(w_{1:n}) = \prod_{i=1}^{n} p(w_i|w_{1:i-1})$ and which has since then become ubiquitous in pre-training large scale generative models. (A concept which will be covered in **section 2.2**). Word Embeddings further became popular with the introduction of Word2Vec [Mikolov et al., 2013a] which introduced 2 methods which were used to calculate word embeddings using 1 hidden layer neural networks, namely Continuous Bag of Words and SkipGram model. Further improvements on the skip gram objective was proposed in the paper [Mikolov et al., 2013b] which introduced the concept of negative sampling. The semantic similarities captured by word embeddings obtained by using word2vec have been explored in detail using cosine-similarity and Pearson-correlation metrics in [Jatnika et al., 2019]. Another model similar to Word2Vec was also introduced named GLoVE (Global Vectors for Word Representation) by [Pennington et al., 2014], which improved upon Word2Vec by training only on the non-zero co-occurences rather than on the entire sparse matrix like Word2Vec. The introduction of Transformer architecture by [Vaswani et al., 2017a] and its pre-trained variants namely BERT [Devlin et al., 2019] and RoBERTa [Liu et al., 2019], have made it even easier for people to extract embeddings for training models on their data.

2.1.2 Classical Machine Learning Approaches

After the text encoding has been applied, classification algorithms are used to classify the text to its appropriate label(s) in a supervised learning setup. In this subsection some popular Machine Learning algorithms which are used in classifying data will be covered in brief.

Random Forest Random Forest is an ensemble method of learning from data first introduced in [Breiman,]. As mentioned before, Random Forest works by creating an ensemble of decision trees which are trained using the bagging method (bootstrap aggregation). It works by randomly selecting examples/observations from the dataset and then training a decision tree on it. The decision trees are kept shallow so as to prevent overfitting. After this step the decision trees are combined and a classification label is assigned to a data point by averaging out.

XGBoost XGBoost is another popular ensemble model used for classification tasks and is an acronym for extreme gradient boosting, it was introduced in [Chen and Guestrin, 2016] and is a hardware optimized implementation of gradient boosted trees. Unlike Random Forests which uses bagging for training the ensemble of decision trees, boosting is an ensemble technique where new models are added to correct the errors made by existing models. Models are added sequentially until no further improvements can be made. Gradient boosting is an approach where new models are created that predict the residuals or errors of prior models and then added together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

Logistic Regression Logistic Regression is a popular algorithm for classification of data points and is often used in conjunction with Deep Learning models like BERT and RoBERTa when fine-tuning on binary or multi-class classification problems. Logistic Regression can be used with the input word vectors for classifying text in both binary and multi-class settings. It uses a sigmoid activation (for binary-classification) or softmax activation (for multi-class classification) after a linear transformation to map the input to probability distribution and the parameters are trained using Stochastic Gradient Descent Algorithm. Logistic Regression was first applied to Language Classification tasks in [Schütze et al., 1995].

2.1.3 Deep Learning Approaches

In this section a brief overview will be given of deep learning approaches which incorporates fine-tuning pretrained models like BERT. In deep learning approaches, as will be seen here, the text encoding step is automated, and with the proliferation of large pre-trained models, it has become easier to fine-tune a model on a dataset.

Recurrent Neural Networks Recurrent Neural Networks [Rumelhart et al., 1986b] have been quite popular in modeling sequential data and thus have been applied to language modeling as well. RNNs work on sequences of arbitrary length by recursively capturing the hidden state of the previous step and applying an activation function. RNNs were applied for large scale classification tasks in [Liu et al., 2016]. Due to the vanishing gradient problem in RNNs, variants of RNNs namely LSTM (Long Short Term Memory) [Hochreiter and Schmidhuber, 1997] and GRU (Gated Recurrent Unit) [Cho et al., 2014] were proposed in order to stabilize the training on long sequences.

Fine-Tuning or Domain Adaptation Recently, the availability of pre-trained models like BERT [Devlin et al., 2019] and RoBERTa [Liu et al., 2019] have made it trivial to use them on classification and regression tasks without explicitly crafting features because these models automate the feature-extraction step. A classification head can be applied directly on top of the features to map into probability space and the parameters of the underlying model can either be frozen or trained using the Stochastic Gradient Descent Algorithm. Variants of stochastic gradient descent have been proposed which helps in stabilizing training such large models on data. Some of them are Adam [Kingma and Ba, 2017], RMSProp and AdamW (Weighted Adam) [Loshchilov and Hutter, 2019]. Lately, AdamW has become very popular in fine-tuning models because of its stability.

2.1.4 Evaluation Metrics

After training a model on a classification objective, it needs to be evaluated on a held out test data so as to analyze how well it performs. This is applicable for both binary and multi-class classification settings. This section will briefly illustrate some of the metrics that are used to evaluate a classification model in a supervised setting.

Accuracy This metric is one of the popular metrics that is used to evaluate the performance of a trained model. It is defined as the ratio of the number of correctly classified data-points to the total number of data-points. **(Number of correct predictions)** / **(Total number of datapoints)**.

Even though accuracy is popular it does not adequately capture the performance of a classifier. Let's take an example of Sentiment Analysis task, if there are 400 data-points which map to Positive and 1000 data-points which map to Negative and the model classifies 1000 of the Negative data points to its correct label but classifies 200 data points of Positive labeled data points as Positive, the accuracy comes out to be 85.7% but on closer inspection we will see that the classifier is misclassifying 50% of the True labels. This is bad if the classifier is deployed in real world applications where correct labeling is of paramount importance, like medicine or email classification. Hence, other metrics have been developed that help us in understanding the classification performance of the model better.

Confusion Matrix: It is especially helpful in understanding the model performance on each label by dividing the classified data-points into a 2×2 matrix with each element of the resultant matrix representing the following:

- **True Positive (TP)**: the number of data points that are correctly classified the Positive sentiment.
- **False Positive (FP)**: the number of data points that have been incorrectly assigned the Positive sentiment.
- **True Negative (TN)**: the number of data points that have been correctly assigned the Negative sentiment.
- **False negative (FN)**: the number of data points that have been incorrectly assigned the Negative sentiment.

Based on the confusion matrix, there are 3 scores that are typically used to help us quantify the performance of the model, they are as follows:

- Precision: This is the metric that calculates the ratio of the number of True Positives (TP) to the sum of number of True Positives and False Positives (FP). Given by $Precision = \frac{n(TP)}{n(TP)+n(FP)}$
- Recall: This is the metric that calculates the ratio of the True Positives to the sum of True Postives and False Negatives. Given by $Recall = \frac{n(TP)}{n(TP)+n(FN)}$
- F1 Score: F1 Score takes into consideration both Precision and Recall, and is given by $F1 = \frac{2*(Precision*Recall}{Precision+Recall}$

For the multi-class setting, the confusion matrix changes to accommodate multiple classes. For example, if we are building a grading system with the following grades $[A, B, C]$, then we will get number of data points assigned a label from the set of grades. In that case Precision, Recall and F1 Score will be calculated as before but for each of the classes.

2.2 Text Generation

Text generation is another exciting part of NLP that has put the field in the general spotlight. Text generation has found widespread popularity with the release of OpenAI's ChatGPT and competing models namely Llama-2 [Thoppilan et al., 2022] and Mistral [Jiang et al., 2023]. However, text generation has been an area of research in Natural Language Processing for a long time and has been used in a variety of tasks such as text summarization and question answering (QA) or dialog systems. The first known text generation system was ELIZA [Weizenbaum, 1966a] which was a rule based dialog system . With the advent of machine learning, Hidden Markov Models were initially used for text generation [Szymański and Ciota, 2002]. The concept of language modeling using Deep Learning was explored in [Bengio et al., 2003] and laid the foundation for more research in this area. As Recurrent Neural Networks became popular in Natural Language Modeling, it wasn't long before they were used for generating text as detailed in [Sutskever et al., 2011], who introduced the concept of Multiplicative RNNs. The introduction of the Transformer architecture in [Vaswani et al., 2017a], led to the development of the Generative Pre-trained Transformer [Radford et al., 2019] which used unsupervised pre-training on a large corpus of text data to train a decoder only model. The authors further fine-tuned the model on text summarization, QA and machine translation tasks. Later OpenAI released GPT-3 [Brown et al., 2020], which was scaled up to 13B parameters and was pre-trained on trillions of tokens of the CommonCrawl text corpus. At inference the model was tested on various Few-shot, One-shot, and Zero-shot learning tasks. Instruct-GPT [Ouyang et al., 2022] which is a variant of GPT-3, was made available to the public and developers as a part of OpenAI API. This model is considered a pre-cursor to Chat-GPT because it used human feedback to rank text while fine-tuning, using a method called Reinforcement Learning with Human Feedback [Christiano et al., 2023, Stiennon et al., 2020]. Other competing models were made during this time, notable amongst them are BLOOM [Workshop et al., 2023] from Big Science, and OPT [Zhang et al., 2022a] from Meta AI. These large language models have been met with significant enthusiasm as well as skepticism from various NLP researchers and the general public.

2.3 Conclusion

This section dealt with ways in which Machine Learning has aided in building systems that can make use of language to solve various problems related to text classification and text generation. Machine learning approaches for text classification can be divided into 2 broad categories. The first one deals with applying statistical machine learning models like Logistic regression or Random Forests to features extracted using statistical algorithms like Term-Frequency Inverse-Document-Frequency(TF-IDF) or shallow deep-learning methods like Word2Vec; the second one deals with deep learning approaches that automate feature extraction, using models like RNNs, LSTMs, and pre-trained Transformer-based models like BERT and its variants like RoBERTa. Text generation has also evolved from frequency-based methods like n-grams, automated methods like Hidden Markov Models to deep learning based methods like GPT-4 which has garnered a lot of public attention. However, despite the breakthroughs that have been made in the realm of applying Deep Learning to NLP, there are problems that persist as well. Researchers have pointed out gender, inter-sectoral and socio-demographic biases [Bender et al., 2021, Bhaskaran and Bhallamudi, 2019, Blodgett et al., 2020, Barth et al., 2022]that these models are prone to have. Also there has been concern about the datasets that are used to train Large-Language Models like GPT-4, as well as concerns that the models exploit spurious correlations that might exist in the data. Care needs to be taken when curating data for the models before training them and releasing into the world and also including human-encoded information while training such large models. The following sections will dive deeper into how human knowledge has been used to train models and what the future scope is.

3 Symbolic approaches to natural language processing

Symbolic AI, as a prevalent approach, covered the period from the 1950s until the late 1980s. The pioneer who introduced first-order logic applications to abstract reasoning in AI systems was John McCarthy [McCarthy, 1960]. By leveraging propositional representation, from a linguistic point of view, the sentence gains meaning through the ordered combination of its parts and their respective meanings. This is called the principle of compositionality. The sub-parts of the sentence bind relations among them that, in each step of reasoning, are calculated, inferring a subset of relations. This progression adheres to a formally defined set of rules [Garnelo and Shanahan, 2019]. Indeed, in such a system, the relationships between distinct objects are represented by symbols. Thus, the employment of a compositional system of representation has the capability to grant the agent the ability to create abstractions and generalize well beyond its own experiences. This is possible as representations of familiar objects and relations can combine in new and novel ways.

Symbolic AI uses knowledge representation structured by humans; they are human-knowledge-driven, and mostly, they are transparent, meaning that their explainability is significant, allowing humans to understand the steps of the algorithms. Nowadays, state-of-the-art research is studying the possibilities to integrate a Symbolic AI approach with Deep Learning, and this brand new field is called Neuro-Symbolic AI. According to the various implementations of these two branches, we have diverse definitions of Neuro-Symbolic [Garcez and Lamb, 2020].

Symbolic AI has been applied to the Natural Language Processing domain since the 1950s. One of the first experiments was the Georgetown machine translation task. In the 1960s, restricted vocabularies were introduced for SHRDLU [Winograd, 1971], developed for an NLP subtask, Natural Language Understanding. Within the program, the user engages in a dialogue with the computer, manipulating objects, assigning names to collections, and inquiring about the status of a simplified "blocks world," essentially, a virtual container populated with various blocks. Another experiment in this period was ELIZA [Weizenbaum, 1966b], a simulation of a Rogerian psychotherapist. In the 1970s, numerous programmers started creating "conceptual ontologies," organizing real-world information into data comprehensible to computers. From the 1980s, rule-based parsers emerged. Along with the mentioned techniques, others were introduced like Knowledge Graph and Linked Data, namely the foundation of the Semantic Web.

In this section, we will focus on the most representative knowledge structures modeled in symbolic AI for NLP tasks. Due to the fast development in the AI field, some of them do not represent the current state-of-the-art anymore. We will cover: Controlled vocabularies, Knowledge Graphs, Linked Data, Ontologies, Abstract Meaning Representation, and Rhetorical Structure Theory.

3.1 Controlled Vocabularies

The creation and maintenance of Controlled vocabularies is a field of knowledge management that prioritizes the effective storing and organization of the information for subsequent retrieval. It involves the utilisation of systematic tools to handle the ambiguity of natural language with limited vocabularies like thesauri, taxonomies, and ontologies [Pedraza-Jimenez and Codina, 2009, Vallez et al., 2015]. These systems of organizing information descriptions are often used as synonymous with controlled vocabularies. Despite this ambiguous naming convention, ontologies overcome the limitation of a controlled vocabulary due to the representation of the relations amongst the entities [Smith, 2021]. In this section we will focus on Thesauri. [Smith, 2021].

3.1.1 Thesauri

In the realm of information and documentation, the term Thesauri holds specific significance, referring to a form of "systematic vocabulary" or subjects. This "compilation" consists of a defined set of terms, or more precisely 'descriptors,' interconnected through mutual references or the application of specific codes. These connections are established based on synonymous affinities or hierarchical dependencies, with the ultimate goal of facilitating the classification and automatic retrieval of documents.

Historically, the construction of controlled vocabularies has been a manual undertaking. Domain experts collaboratively set common objectives, collect and organize terms, and subsequently express them in the SKOS format. The distinction between manually created thesauri (e.g., Roget, WordNet) and those generated automatically from text corpora lies in their nature. The former can be considered semantic, relying on meaning as a clustering factor, while the latter are distributional, with computers categorizing terms based on distribution patterns [Kilgarriff, 2003].

Well-crafted thesauri have demonstrated their efficacy in enhancing the efficiency of various NLP tasks such as parsing and word sense disambiguation. They are also valuable for representing ontologies.

In recent years, the volume of available data has consistently grown, leading to increased demand for scalable approaches. For instance, [Riedl and Biemann, 2013] showcased the scalability of their method using the extensive Google Books dataset, derived from dependencies in 17.6 billion sentences. However, it remains unclear whether this approach is applicable to other large corpora beyond Google Books Syntactic N-grams.

A novel approach to computing Distributional Thesauri (DT) differentiates itself by employing pruning techniques and a distributed framework. This innovation makes computations for vast corpora feasible on comparatively modest computational resources, yielding higher-quality DTs and outperforming previous methods in terms of both speed and quality.

Further enhancements involve combining distributional thesaurus embeddings with state-of-the-art word representations, resulting in significant performance improvements for various natural language processing tasks. This approach does not necessitate handcrafted lexical resources and is applicable to languages other than English. It contributes to ongoing research in learning word representations, a fundamental step in understanding text and improving natural language understanding [Jana and Goyal, 2018].

In contemporary NLP trends, there is a growing emphasis on elevating the importance of minority languages. This shift acknowledges that a majority of tasks are accomplished using resource-rich languages such as English. Performance disparities are notable for low-resource languages due to the scarcity of available data, making them under-represented in scientific research. Researchers are now actively preserving the linguistic diversity spectrum.

The application of Distributional Thesaurus (DT) extends to this sector. Enhancing contextualized representations for low-resource languages using DT has become a methodology to improve the state-of-the-art performance for various NLP tasks. A recent approach involves implementing a Graph Neural Network on top of pre-trained Transformer models. The DT, a semantic count-based similarity graph constructed based on the distributional hypothesis, captures semantic relationships between words [Harris, 1954]. In [Venkatesh et al., 2022], the authors fuse information from a DT with the transformer architecture to enhance contextualized representations for low-resource languages. By incorporating this information, the authors aim to improve the model's ability to capture semantic relationships and enhance performance across various NLP tasks. They validate the effectiveness of their approach across low and mid resource languages: English, German, Hindi, Bengali, and Amharic.

3.2 Knowledge Graphs

Many of the symbolic approaches to artificial intelligence consist of concepts and methods usually associated with what has been called the Semantic Web. The Semantic Web can be seen as a set of theories, standards and techniques concerned with establishing forms of representing knowledge in a way that it can be easily shared, reused and understood by computational agents [Hitzler, 2021].

An important step for the development of the semantic web was the use of graph structured knowledge commonly referred as knowledge graphs. It has been highly influenced by the launch of the Google Knowledge Graph. The difference with respect to linked data, such as DBpedia, is the emphasis on internal consistency [Hitzler, 2021]. An example of such a universal standard is schema.org, a set of schemas whose vocabulary is widely used in industry [Guha et al., 2016]. Knowledge graphs have been used for question-answering, search engines and recommendation systems [Chen et al., 2020].

Part of the current research on knowledge graphs has been on automated reasoning. If traditional reasoners use first-order logic to make inferences, the research on first-order logic reasoners in the past decade has focused on making these algorithms robust in large-scale graphs. Algorithms that deal with fuzzy description logic based on random walks have also been developed. Another branch of research is based on projecting graph entities into vector spaces. Once the vector space is defined, numerical techniques can be used to make inferences [Chen et al., 2020].

Chen et al. [Chen et al., 2020] also mention as relevant techniques dimensionality reduction, vector distance evaluation and semantic matching energy functions. Another relevant branch of research on knowledge graph reasoning is based on neural networks. For example, [Shi and Weninger, 2018] develop a noisy robust convolutional neural network for graph completion. Neural networks based models have raised many NLP benchmarks in the recent years, but they have some shortcomings: they can also be sensitive to noise, they are black boxes that can be hard to interpret and they also need a significant amount of data to perform well [Wang et al., 2023].

Since knowledge graph (KG) reasoning can infer new from existing information, it can be applied to graph completion and graph entity classification. Outside the knowledge graph domain itself, knowledge graph reasoning has been applied to several other domains, Chen et al. [Chen et al., 2020] cite as meaningful examples: medical diagnostics, finance, question answering and recommendation systems. In the medical domain, for instance, it has been used for decision support systems in intensive care [Yuan et al., 2018]. In the financial field, KG reasoning has been applied to fraud identification [Kapetanakis et al., 2012] and stock volatility prediction [Ding et al., 2016].

3.3 Linked Data

Another important standard of the Semantic Web is the Resource description framework (RDF). RDF is a "syntax for expressing directed, labeled and typed graphs" [Hitzler, 2021]. RDF documents are based on object-attribute-value triples. The advantage of this notation is that objects and values can be interchanged. Since RDF statements themselves can be "the object or value of a triple", RDF allows for graph nesting and chaining [Decker et al., 2000]. Representing knowledge in RDFs also allows for the use of SPARQL, a graph matching query language [1].

What is usually referred as Linked Data or Linked Open Data is usually a set of large and interconnected RDFs [Hitzler, 2021]. The main example is DBpedia. The DBpedia project is a conversion of Wikipedia into structured graph knowledge [Auer et al., 2007]. It allows for use of complex queries. Linked Data organized in RDFs have been considered to make integration, reusability and diffusion of knowledge easier than plain ontologies [Hitzler, 2021].

One of main results of the Semantic Web standards is that resources become easily shareable. Hitzler [Hitzler, 2021] observes that, around 2010, the community started to become aware of the limitations of linked data. According to him, it became clear that integrating different sources of data needed "more effort than initially thought".

3.4 Ontologies

Ontologies are usually understood as a framework to represent information. An ontology can make explicit the common understanding of general or domain knowledge and make this common understanding reusable, shareable and analysable by humans and computers [Noy et al., 2001]. In practice, ontologies consist of classes representing concepts which are organized in a taxonomic hierarchy and instances of these classes with their properties and relations [Noy et al., 2001]. The concept of Ontology is domain-agnostic, for instance, there are Ontologies for music production [Raimond and Sandler, 2012], geographic information systems [Fonseca and Egenhofer, 1999] or bioinformatics [Stevens et al., 2000].

The most common language used to define an ontology is OWL [2]. Its syntax allows the ontology to be defined in terms of logical formalism, like axioms of a logical theory [Handschuh, 2007]. An ontology expressed in OWL can be used by automated reasoners to validate and infer knowledge from the ontology.

Ontologies can be used to make domain knowledge explicit, providing domain specific vocabulary, integrate information, facilitate information retrieval and allow for question answering in expert systems [Handschuh, 2007]. One of the main advantages of expressing knowledge as an

[1] www.w3.org/TR/rdf-sparql-query
[2] www.w3.org/OWL

Ontology is the use of reasoners. A reasoner is a software with the ability to infer knowledge not explicitly stated on the ontology semantics following descriptive logic rules [Abburu, 2012].

Ontologies, linked data, knowledge graphs and technologies associated with them constitute the core of the Semantic Web, but there are additional standards worth mentioning. The first is topic maps, a standard for the representation and interchange of knowledge, with an emphasis on the findability of information. Topic maps can represent any type of semantic networks, where the nodes are represented by topics, the association of topics are modeled by the links between nodes, and semantic meaning can be added to the model by using topic types, association types and association role types [Steiner et al., 2001]. According to the ISO definition [Steiner et al., 2001], reasons to model information in a topic map might be the following: organizing and visualizing unstructured information; structuring information in a way that it can be navigated using tools like indexes, citation systems or glossaries; and filtering customizable views of information.

3.5 Abstract Meaning Representation

Abstract Meaning Representations were originally introduced by Langkilde and Knight in 1998 [Langkilde and Knight, 1998] as a derivation of the Penman Sentence Plan Language [Kasper, 1989].

In the field of Natural Language Processing (NLP), Abstract Meaning Representation (AMR) involves representing the meaning of a sentence using directed, acyclic, rooted labeled graphs as a means of schematizing its internal semantic structure in a human-readable way. This method is capable of generalizing across diverse sentences with similar meanings [Banarescu et al.,]. Through this approach, both the content and intent of the text can be preserved [Kumar and Solanki, 2023]. These graphs are valuable for various tasks in NLP that require a deep understanding of semantic content in text.

In this section, we will present relevant works that cover the most important aspects of the field of AMR starting from 2013, the year in which this field regained interest among scientists.

By introducing a novel approach to parse sentences into the AMR, namely a two-part algorithm called JAMRA capable of identifying both concepts and relations, [Flanigan et al., 2014] provided a strong baseline for future improvement. The method is based on a unique algorithm for finding a maximum spanning, connected sub-graph, embedded within a Lagrangian relaxation of an optimization problem that imposes linguistically inspired constraints. The proposed algorithm for parsing sentences into AMR consists of two parts. The first part identifies concepts using a semi-Markov model, which maps spans of words in the input sentence to concept graph fragments. The second part identifies the relations between these concepts by searching for the maximum spanning connected sub-graph (MSCG) from an edge-labeled, directed graph representing all possible relations between the identified concepts. To solve the latter problem, the authors introduce a novel algorithm that is similar to the maximum spanning tree (MST) algorithms that are widely used for dependency parsing.

In 2016, a task in the SemEVal-2016 challenge was dedicated to Meaning Representation. The evaluation of the submitted systems revealed that AMR parsing is a difficult and competitive task, with a large number of systems using released code to lower the bar to entry. However, this may have led to a narrowing of diversity in approaches. The best-performing systems tended to perform well on instance matching but struggled with relation matching. Low-level irregularities such as creative tokenization and misspellings also posed challenges for the systems. Future research in this area may focus on improving the performance of systems on relation matching and addressing these low-level irregularities. The following year, the SemEval-2017 also included AMR parsing task in the biomedical domain [**?**].

Subsequently, [Damonte and Monti, 2021] was able to introduce a state-of-the-art parser for AMR. It also proposes a test-suite that assesses specific sub-tasks that are helpful in comparing AMR parsers. The parser is competitive with the state of the art on the LDC2015E86 dataset and outperforms state-of-the-art parsers for recovering named entities and handling polarity. The transition-based parser for AMR presented in this paper works by processing sentences left-to-right, in linear time, using a transition system that builds AMR graphs. The system is characterized by a set of configurations and transitions between them, with the basic components of a configuration being a stack of partially processed words and a buffer of unseen input words. Starting from an initial configuration, the system applies transitions until a terminal configuration is reached. The sentence is scanned left to right, with linear time complexity for dependency parsing. This is made possible by the use of a greedy classifier that chooses the transition to be applied at each step. The parser is shown to be competitive with the state of the art on the LDC2015E86 [Knight et al., 2021] dataset and outperforms state-of-the-art parsers for recovering named entities and handling polarity. The main advantage of this parser over other parsers is that it demonstrates that it is possible to perform AMR parsing using techniques inspired by dependency parsing, which are computationally less expensive than traditional graph-based techniques.

The authors [Hardy and Vlachos, 2018] present a novel approach to abstractive summarization that incorporates explicit semantic modeling of the source document and its summary. By guiding the neural language generation stage with the source document, they achieve significant improvements in summarization results. According to the experiments conducted in this paper, the guided neural language generation approach outperforms the unguided baseline model and achieves results that are comparable to state-of-the-art AMR-to-text approaches. Specifically, the guidance from the source document improves the result of AMR-to-Text by 7.4 and 10.5 points in ROUGE-2 using gold standard AMR parses and parses obtained from an off-the-shelf parser respectively. The summarization performance using the latter is 2 ROUGE-2 points higher than that of a well-established neural encoder-decoder approach trained on a larger dataset.

The reinforcement learning approach improves the parser's performance by using the Smatch score of the predicted graph for a given sentence as a reward [Naseem et al., 2019]. This allows for training with exploration of the action space and alleviates the strong dependency on hard alignment. The self-critical policy gradient training algorithm is used, which is a special case of the REINFORCE algorithm with a baseline. Training with exploration via reinforcement learning gives further gains of about 2 points and achieves one of the best results ever reported on the task and state of the art in some of the metrics. The AMR-to-text alignments are used to align the AMR graph with the corresponding sentence, which is a crucial step in AMR parsing. The parser uses several AMR-to-text alignments with an attention mechanism to improve its performance. Contextualized embeddings, on the other hand, are used to provide more than just a point on top of the best model with traditional word embeddings. The paper reports that BERT contextualized vectors provide better results than traditional word embeddings, and combining BERT with a model that only sees words achieves the best results surpassed only by models that also use contextualized vectors and reinforcement learning objective. Therefore, the parser is supplemented with pre-processed contextualized embeddings to improve its performance.

Bevilacqua et al. [Bevilacqua et al., 2021] show that by using a symmetric transduction task and a carefully designed graph linearization, it is possible to achieve state-of-the-art performances in both tasks using the same seq2seq approach. This means that complex pipelines and heuristics are no longer necessary, and the need for graph recategorization is actually harmful outside the standard benchmark. First, the authors show that a symmetric seq2seq approach can be used to perform both Text-to-AMR parsing and AMR-to-Text generation, which simplifies the overall process and makes it more efficient. Second, it demonstrates that a carefully designed graph linearization algorithm can improve the accuracy of both tasks and eliminate the need for graph recategorization. Third, it suggests that the use of heuristics and complex pipelines may not always be necessary for achieving state-of-the-art performances in semantic parsing and generation. Finally, this research opens up new directions for integrating parsing and generation and for developing more robust automatic AMR approaches. Overall, this work contributes to the development of more effective and efficient natural language processing systems that can be applied to a wide range of tasks and domains.

Thanks to the data augmentation (DA) that uses Abstract Meaning Representation (AMR) to enhance NLP/NLU tasks (AMR-DA), it is possible to convert a sentence to an AMR graph, modify it with various policies, and generate new augmentations. The results are impressive, as AMR-DA outperforms other techniques and leads to more robust improvements. AMR-DA outperforms other data augmentation techniques such as EDA [Wei and Zou, 2019] and AEDA [Karimi et al., 2021] in NLP/NLU tasks. In particular, experiments show that AMR-DA boosts the performance of state-of-the-art models on several STS benchmarks and leads to more robust improvements in text classification [Shou et al., 2022].

Lastly, since transformer models have become increasingly popular, [Kumar and Solanki, 2023] proposes a Transformer Technique with Self-Attention Mechanism (T2SAM) to improve the performance of text summarization. Their model outperforms existing state-of-the-art baseline models and is trained on a combination of the Inshorts News dataset [3] and the DUC-2004 shared tasks dataset [Kumar and Solanki, 2023].

3.6 Rhetorical Structure Theory

In a text, clauses, sentences, and paragraphs are intricately interconnected to create a unified discourse. The aim of discourse parsing is to reveal this underlying structure of coherence, a procedure that has demonstrated its benefits for a wide array of applications in Natural Language Processing (NLP). Rhetorical Structure Theory (RST) was the core idea behind parsers, and it became an important part of the branch of parser engineering [Neumann, 2021]. This approach [Mann and Thompson, 1988] was theorised to study the logical and rhetorical hierarchical structure within a general document, namely the Discourse Tree (DT). Within the realm of NLP, Rhetorical Structure Theory (RST) deals with the structure and analysis of discourse in

[3] Available at kaggle.com/datasets/shashichander009/inshorts-news-data

a document. Specifically, it adopts a constituent tree representation where fundamental units, called Elementary Discourse Units (EDUs), form the terminal nodes. These EDUs, which can be segments of sentences or clausal units, are connected through rhetorical relations, such as ”Elaboration” or ”Attribution,” to form larger text spans until encompassing the entire document. These segments are then classified as nuclei (central and fundamental) or Satellites (subordinate and supportive) based on their relative importance in rhetorical relations. Consequently, document-level discourse parsing through RST involves four essential subtasks:

1. EDU segmentation,
2. tree structure construction,
3. determination of nuclearity,
4. classification of relations

Such an approach provides a detailed organizational structure of discourse, enabling an in-depth understanding of relationships between text components. Indeed, the application of RST in document parsing is crucial for unveiling the hierarchical structure of discourse and meaningful connections.

Its applications vary from summarization [Marcu, 1998],[Hirao et al., 2013], [Gerani et al., 2014], question-answering [Jansen et al., 2014], sentiment analysis [Bhatia et al., 2015], conversational machine reading [Gao et al., 2020], machine translation evaluation [Lin et al., 2019] and text categorization [Ji and Smith, 2017].

The following paragraphs provide a comprehensive diachronic analysis of key contributions and critical aspects in the realm of Rhetorical Structure Theory (RST) discourse parsing, presenting a nuanced understanding of the field’s evolution from 2014 to 2022, by assessing the approaches that have followed in recent years up to the current state of the art.

Feng and Hirst’s seminal work [Feng and Hirst, 2014] marked a pivotal moment by introducing an adversarial learning approach to address the scarcity of training data, proposing a top-down neural architecture for text-level Discourse Representation Structures (DRS) parsing that sidestepped the use of handcrafted features. Despite its success, the paper acknowledges the ongoing challenge of achieving perfection in performance. Li’s [Li et al., 2016] attention-based hierarchical neural network model advanced document-level discourse parsing, emphasizing comparable performance to existing systems while minimizing manual feature engineering. This work recognized the importance of document-level parsing, identifying limitations in current algorithms and calling for more efficient models.

Morey et al.’s (2017) replication study [Morey et al., 2017] critically assessed progress in RST discourse parsing, revealing that reported performance increases were often artifacts of differences in evaluation procedures. This paper’s emphasis on the need for standardized evaluation metrics is pivotal, challenging the reliability of reported gains and urging the community towards a unified evaluation framework.

Wang’s (2017) [Wang et al., 2017] two-stage parsing method represented a leap forward in text-level discourse analysis, introducing a neural network for relation labeling and a graph-based algorithm for structure building. This novel approach challenged prior methods reliant on hand-crafted features, achieving state-of-the-art performance on the RST Discourse Treebank. The paper underscored the challenges posed by the scarcity of annotated data and the intricate nature of rhetorical structures, pushing the boundaries of discourse parsing methodologies. Kobayashi's (2020) proposal of a top-down RST parsing method at multiple granularity levels – paragraphs, sentences, and Elementary Discourse Units (EDUs) – demonstrated a nuanced approach, efficiently building RST trees for the entire document. The results showcased state-of-the-art scores, emphasizing the potential downstream benefits for tasks like sentiment analysis and summarization. This paper identified the significance of parsing granularity levels and set a new benchmark for accuracy and efficiency.

Kobayashi's innovative use of silver data (cleaned data) for pre-training a neural RST parser addressed the persistent challenge of limited annotated training data, leading to substantial performance improvements [Kobayashi et al., 2021]. This marked a shift in the field's methodology, allowing models to learn from vast amounts of automatically annotated data without the need for costly manual efforts. The study showcased remarkable gains in relation labeling, a critical aspect where large annotated training data was often lacking. The joint framework for document-level multilingual RST discourse segmentation and parsing, proposed in [Liu et al., 2021b], was a groundbreaking contribution, addressing key issues in prior models. By integrating EDU segmentation into tree parsing, supporting multilingual parsing, and achieving state-of-the-art performance, this framework emerged as a game-changer in the realm of natural language processing. This approach tackled the limitations of existing parsers, enabling their application to new data and diverse languages.

Neumann's RST-workbench software package simplified the usage of numerous end-to-end RST parsers, offering a web-based interface for concurrent text analysis [Neumann, 2021]. This tool emerged as a valuable resource for the community, enabling users to visually compare RST trees, manually post-edit them, and store analyses for future reference.

Nguyen's [Nguyen et al., 2021] proposal of a top-down end-to-end formulation for document-level discourse parsing in the RST framework represented a significant leap forward . The method, treating discourse parsing as a sequence of splitting decisions at token boundaries, outperformed existing methods without handcrafted features, offering speed and adaptability to new languages and domains. This approach held the potential to enhance various NLP applications, such as text classification, summarization, sentiment analysis, machine translation evaluation, and conversational machine reading.

In 2022, Kobayashi's exploration of a "Simple and Strong Baseline for End-to-End Neural RST-style Discourse Parsing" continued the trajectory of advancement in the field. By combining transformer-based pre-trained language models with both top-down and bottom-up parsing strategies, the study delved into parsing performance dependencies on pre-trained language models rather than parsing strategies. The bottom-up parser, particularly when employing DeBERTa, showcased substantial gains, challenging the prevailing notion that parsing strategies were the primary determinants of performance. The study also shed light on the impact of language models with a span-masking scheme, particularly in intra- and multi-sentential parsing and nuclearity prediction. By providing a strong baseline for RST parsing based on open-source parsers and transformer-based models, the paper contributed significantly to the ongoing discourse on the role of pre-trained language models in parsing performance [Kobayashi et al., 2022].

3.7 Conclusion

This section discussed the evolution of Symbolic NLP and some of its most relevant concepts and theories. We have seen different ways of organizing information (Ontologies, Linked Data, Knowledge Graphs, Topic Maps and Controlled Vocabularies), some of their applications, recent research and limitations. It also discussed two important theoretical frameworks, Abstract Meaning Representation and Rhetorical Structure Theory. Recent research on AMR has focused on improving parsing, which has also benefited from the new advances in deep learning.

Regarding Rhetorical Structure Theory, the survey of previous work in RST discourse parsing underscores the evolution of RST discourse parsing, from foundational adversarial learning approaches to sophisticated top-down and bottom-up parsing strategies. The contributions discussed in each paper have significantly shaped the field, addressing critical challenges such as limited training data, the need for standardized evaluation metrics, and the complexity of rhetorical structures. The diachronic variable emphasizes not only the temporal evolution of techniques but also the interconnectedness of these advancements in addressing persistent challenges in RST discourse parsing. The field has witnessed a transition from traditional methodologies reliant on handcrafted features to innovative approaches leveraging neural architectures, attention mechanisms, and pre-trained language models. While each paper represents a valuable contribution to the field, the collective impact of these works lies in their collective ability to push the boundaries of what is achievable in the complex and dynamic landscape of RST discourse parsing.

4 Hybrid Approaches to Natural Language Processing

Machine learning has made remarkable progress over the past few decades achieving human-level performance in various Natural Language Processing (NLP) tasks. Large language models, in particular, have garnered global attention in the last couple of years, captivating not only the NLP community but also integrating into the daily routines of numerous professionals. Meanwhile, the symbolic approaches have seen significant advancements in representing human cognitive capabilities and knowledge-rich data enabling it to be more interpretable and comprehensible. However, these two main pillars of computer science research possess their own set of advantages and disadvantages.

Machine learning approaches are stronger in learning patterns and relationships in data via optimization strategies. However, they often fall short in capturing and interpreting the factual knowledge required for most downstream NLP tasks, instead attempting to mimic the facts and knowledge present in the training data [Pan et al., 2023]. This hinders both traditional machine learning approaches and deep learning methods from achieving high performance in knowledge-intensive tasks. Further, the latest studies [Petroni et al., 2019] have demonstrated that even the robust large language models trained using vast amounts of training data and parameters suffer from hallucinations generating non-factual responses. This raises significant concerns regarding the trustworthiness of such expensive large language models. Similarly, the symbolic approaches are stronger in resembling human cognitive abilities by explicitly capturing the knowledge required for a task. On the other hand, the symbolic approaches have shown poor learning skills compared to machine learning approaches, especially in handling dynamic data and generalizing the knowledge beyond training data [Pan et al., 2023].

Recent focus has been turned into bridging the gap between machine learning approaches and symbolic approaches to overcome the limitations of both when applied independently while retaining their strengths [Zhu et al., 2023]. This hybrid approach enables the statistical methods to utilize knowledge-enriched input data to improve the inference with external knowledge and to produce interpretable results. At the same time, the symbolic approaches are empowered with statistical learning to incorporate semantic knowledge and generate generalized knowledge representations. In particular, the combination of deep learning and symbolic approaches has paved the way for a new research area called Neuro-symbolic approaches [Hamilton et al., 2022] aiming to develop trustworthy and interpretable NLP solutions. Furthermore, a recent survey [Sarker et al., 2021] has shown, that neuro-symbolic solutions gain benefits in four key research aspects including interpretability, generalization to handle both small training data and out of distributions and error recovery due to aggregated advantages of both deep learning and symbolic approaches. This has drawn the attention of the computer science research community to develop effective hybrid solutions to solve various real-world problems.

This section of the report presents a brief review of existing hybrid approaches that combine machine learning or deep learning techniques and symbolic approaches for NLP. The following subsections introduce hybrid techniques used for three key fields of NLP, Natural Language Understanding, Natural Language Generation, and Natural Language Reasoning, and present a concise overview of the recent development in the adoption of hybrid approaches for various NLP tasks within each field.

4.1 Hybrid Approaches to Natural Language Understanding

Natural Language Understanding (NLU) is a branch of Natural Language Processing (NLP) that focuses on understanding the meaning and semantics of text. This includes various downstream NLU tasks such as text classification, sequence labeling, and question answering. This section briefly introduces the latest hybrid approaches used to produce state-of-the-art performance in popular NLU applications.

4.1.1 Text Classification

As introduced earlier in 2.1, text classification is the task of automatically assigning a label from a predefined set of labels for an input text. While there are enormous amounts of text classification tasks explored across various domains, sentiment analysis, stance detection, and language detection are some of the prominent tasks that attracted considerable focus of the NLU research community. With the introduction of large language models and their exceptional performance across various NLP tasks, text classification tasks often exploit fine-tuning language models on limited domain-specific training data [Pittaras et al., 2023].

A straightforward strategy for adopting hybrid approaches for text classification is to incorporate external knowledge obtained via symbolic approaches as additional input to the classification models. Skrlj et al. [Škrlj et al., 2021] use word taxonomies from Wordnet [Fellbaum, 2010] to generate new semantic features for text classification. The authors transform the input document into a semantic feature representation by extracting hypernyms of words from the documents and obtaining its double normalized TF-IDF scores, followed by feature selection. The authors observed a significant improvement in the performance across six short text classification tasks when these external features were used with neural classifiers. Similarly, Liu et al. [Liu et al., 2022] extract concept words using the Probase knowledge base [Wang et al., 2010] and obtain a concept word embedding by aggregating word embeddings of all the concept words present in a text. This knowledge-enriched vector representation is provided as additional input to a neural model for text classification. Instead of constructing a knowledge-rich vector representation as additional input, Liu et al. [Liu et al., 2020a] generate a sentence tree as input to the BERT model by injecting knowledge graph triplets to corresponding places in the sentence. Injected knowledge is controlled via techniques such as soft-positioning and visible matrix. The Authors demonstrated promising results for the knowledge-injected BERT named K-BERT in 12 NLP tasks including text classification, sequence labeling, and question answering.

Hybrid architectures are an alternative to injecting knowledge as inputs to classifiers, which can combine symbolic representation learning models with traditional classifiers to improve inference. One of the pioneering works in this direction for text classification was experimented by Yao et al. [Yao et al., 2019]. The authors propose TextGCN, a graph neural network architecture that models documents and words as nodes in a graph to generate a heterogeneous graph, and then jointly learn embedding representations for words and documents or nodes in the graph as supervised learning for performing text classification. This enables the authors to transform the text classification problem into a node classification problem in a heterogeneous graph and TextGCN is shown to outperform standalone neural models in several text classification tasks. Section 5 further discusses the hybrid techniques used for text classification.

4.1.2 Sequence Labeling

Sequence labeling is the task of assigning a label to text at the word level instead of the sentence or phrase level from a predefined set of labels. Named entity recognition, part-of-speech tagging, and language detection are some of the well-known sequence labeling tasks. Information predicted at a word level in sequence labeling tasks generally serves as input features for various other downstream NLP tasks.

Similar to injecting external knowledge extracted from a knowledge graph into an input text, several attempts have been made to adopt hybrid techniques for sequence labeling tasks. In particular, the entities present in a text can be linked with external information for knowledge-augmented learning. Motivated in that direction, Enhanced Language RepresentatioN with Informative Entities (ERNIE) was proposed by [Zhang et al., 2019] to learn language representation with infused entity knowledge. The authors identify entities in the text first, align them with a knowledge graph, and obtain their entity embedding using graph embedding techniques. Following that, the authors train an auto-encoder architecture with random entities masked in the text corpus to enforce the representation learning to incorporate entity knowledge. This model was shown to be outperforming BERT-based architectures in various NLP tasks including sequence labeling with limited finetuning [Wang et al., 2020b]. The idea was extended to BERT-base architectures later for sequence labeling [Hu et al., 2022].

Linking Wikipedia data for improving sequence labeling, especially for recognizing named entities has been explored in recent research works [Tedeschi et al., 2021, Wang et al., 2022b, Boros et al., 2022]. While Tedeschi et al. [Tedeschi et al., 2021] exploit Wikipedia data for creating automatically annotated training data, Wang et al. [Wang et al., 2022b] and [Boros et al., 2022] extract external information from Wikipedia for knowledge-enriched training. The authors use query-based approaches to retrieve relevant information from knowledge graphs constructed using Wikipedia. Apart from these techniques, graph neural architectures are also explored for sequence labeling tasks by converting the word-level classification problem into a node classification problem in a graph [Gui et al., 2019], enabling the authors to capture non-sequential dependencies via graph structure.

4.1.3 Question Answering

Question Answering (QA) is the task of generating or choosing relevant answers for a question in natural language. It is one of the notable NLU tasks that often requires common sense and external knowledge and demands early adoption of hybrid solutions. While training large language models in Wikipedia data and books enabled them to show exceptional performance in QA tasks [Mitra et al., 2019], numerous other hybrid solutions have been explored in the literature in recent times.

Similar to other NLU applications, an evident hybrid technique is to provide external knowledge required to perform the QA task as input to the inference model. [Mitra et al., 2019] perform an elastic search using question and answers as a query in external sources and retrieve the top 50 related sentences. Authors rerank the retrieved sentences using sentence similarity and provide the top 10 sentences as additional information to a BERT model. For effective context retrieval, Karpukhin et al. [Karpukhin et al., 2020] propose obtaining dense vector representation of passages using both the TF-IDF approach and a BERT model. During the inference, the dot-product between the dense representation of the question and the passage is used to determine the relevant context to be used as external information.

The integration of knowledge graphs into QA tasks is an expected development in the progression of hybrid approaches. Lv et al. [Lv et al., 2020] extract knowledge from ConceptNet [Speer et al., 2017] and Wikipedia and generate a knowledge graph for further inference. Graph convolution networks (GNN) are used to extract node representation from two knowledge graphs and the obtained contextual representation of questions and answers are fed into another graph-based attention model to choose the right answer. In addition to the enriched information available in each node in the knowledge graph, Feng et al. [Feng et al., 2020] propose techniques to utilize relational paths using a Multi-hop graph relation network, leading to more interpretable models. The proposed approach chooses a sub-graph related to the input query and both node embedding and path embeddings are obtained using a GNN architecture. Finally, the correct answer is chosen using a text encoder model given the embedding representations of questions and answers. Yasunanga et al. [Yasunaga et al., 2021] extend this work to obtain a subgraph related to the QA context and then score each node in the subgraph using a language model. Later the authors jointly learn the representation of QA context and nodes in the graph using a graph neural network. The proposed model was shown to outperform other knowledge graph augmented language models in QA tasks in commonsense and biomedical domains. Following this approach, similar ideas of mutually exchanging information between language models and knowledge graphs were explored in literature for QA task [Zhang et al., 2022b].

4.2 Hybrid Approaches to Natural Language Generation

Natural Language Generation (NLG) is a fundamental task in Natural Language Processing, aiming to produce meaningful text in Natural Language. This requires semantic and syntactic understanding of the language to generate text which makes it challenging while also ensuring its applicability across a broad spectrum of NLG tasks such as dialogue systems, summarization, and machine translation. NLG solutions generally follow encoder-decoder architecture [Vaswani et al., 2017a], where the encoder understands the input text and generates hidden states interpreted by the decoder to generate meaningful text [Yu et al., 2022b]. However, the text generated by those powerful models often fails to match human responses due to the limited knowledge available in the training data and the lack of generalization capabilities. This demands rapid embracement of hybrid techniques to generate knowledge-enhanced text. This section presents state-of-the-art hybrid approaches used for various NLG tasks.

4.2.1 Language Modeling

Language modeling is the task of learning a universal representation of the language from an unlabelled text corpus. This task is often modeled as a next word or token prediction task, where the language model is trained to predict the next word given its previous or surrounding words in a piece of text. Substantial effort has been made in the direction of integrating external knowledge into language models.

Integration of knowledge sources into the input representation of the language models is a prominent way of infusing external knowledge prior to inference. Similar to K-BERT [Liu et al., 2020a] and ERNIE [Zhang et al., 2019] introduced earlier (section 4.1.1 and section 4.1.2), Peters et al. [Peters et al., 2019] explicitly links entities available in the text by injecting their embedding representation obtained from knowledge graph into the input. The authors observed an increase in the ability to recall facts in the resulting model called KnowBERT. Instead of linking entities, Ji et al. [Ji et al., 2020] extract the relevant subgraph from a knowledge graph using the Multi-hop technique and input their embedding representation by aggregating the node embedding obtained via graph neural network. Liu et al. [Liu et al., 2021a] extend this idea by modifying the encoder-decoder architecture with a dedicated encoder and decoder augmented with embedding representation obtained from the knowledge graph. Following the utilization of node embedding as input for language representation learning, jointly optimizing the node embedding representation as well as the language representation is observed as a promising direction of improvement [Wang et al., 2021, Yu et al., 2022a].

Different from these approaches, Xiong et al. [Xiong et al., 2019], propose to modify the training objective to force the language model to learn about real-world entities. The authors identify and link entities mentioned in the text to Wikipedia and generate negative statements of the corresponding text by randomly replacing the entity occurrences with the names of the same entity types. During the training, the model learns to identify the correct entity. Similarly, Zhou et al. [Zhou et al., 2020] modify the training objective to enforce the model to generate concept-aware text. The first objective imposes the model to predict the original sentence given some unordered keywords of the sentence, whereas the next objective is aimed at recovering the order of concepts in a sentence given a shuffled list of concepts. Here, the authors define verbs, nouns, and proper nouns present in a sentence as concepts. Instead of modifying the training objective, Guan et al. [Guan et al., 2020] post-train the language models on sentences reconstructed from a knowledge graph. The authors convert commonsense triples from ConceptNet [Speer et al., 2017] and ATOMIC [Sap et al., 2019] to readable sentences using a template-based method.

Generating text with complex ideas may require capturing the knowledge from structured or unstructured knowledge from external sources. Recently this problem has been studied as a Knowledge graph to text generation problem, where the information available from external sources is converted to a knowledge graph first and the output text is generated based on the knowledge graph. In this direction, Koncel-Kedziorski et al. [Koncel-Kedziorski et al., 2019] propose GraphWriter, an extension of the transformer model for knowledge-graph-to-text generation. Following this study, various extensions of it were proposed [Cai and Lam, 2020] for encoding structural information in the knowledge graph.

4.2.2 Dialogue Systems

Dialogue systems are designed to converse coherently with humans in natural language. This requires understanding the language, recalling the conversation history, and producing accurate responses. Undoubtedly, the ability to generate precise responses hinges on the understanding of the external world and utilizing common sense.

One of the pioneering works in the hybrid application for dialogue systems was experimented by [Ghazvininejad et al., 2018]. The authors introduce two encoders dedicated to encoding the conversation history as well as the external facts, enabling the responses to be conditioned on both factors. The authors extract focus phrases containing entities from the input query and collect raw text related to the focus phrases from external sources such as Wikipedia. An RNN encoder is used as a fact-encoder to convert the raw text with related facts into a hidden state in the proposed encoder-decoder model. Instead of encoding all the related facts retrieved, [Dinan et al., 2018] use an attention-based component to carefully choose the relevant information gathered from external sources, and use a shared encoder to encode the knowledge and the dialogue context. The authors propose a generative transformer memory network, capable of retrieving relevant information from large memory and generating response conditions on both relevant information and dialogue history.

Integration of knowledge graphs into encoder-decoder models is an anticipated research trajectory. As evidence of this, Zhou et al. [Zhou et al., 2018] propose an encoder-decoder model coupled with graph attention mechanisms. The authors retrieve one knowledge graph per word present in the input query and convert it to a vector representation using a static graph attention mechanism. This vector representation is concatenated with the vector presentation of the corresponding word provided as input to the encoder-decoder model. The decoder model is also combined with a dynamic graph attention mechanism to attend to all the relevant knowledge graphs retrieved to generate the output. Instead of utilizing the existing graphs and looking for related information within them, Zhang et al. [Zhang et al., 2020] generate a concept graph by starting with grounded concepts present in the input and expanding it to more meaningful conversations by traversing through the related concepts. Following the other knowledge graph integration approaches, the author encodes the concept into a vector representation using a graph neural network and input to the encoder-decoder model along with the input query.

4.2.3 Text Summarization

Text summarization is one of the core challenges in NLG which aims to generate summaries based on sources ranging from a single document to a collection of documents. There are two types of underlying approaches for text summarization namely, extractive summarization and abstractive summarization. The first approach strives to choose key sentences or phrases from the source, and it is often solved as a ranking or scoring task of existing sentences in the source. On the other hand, abstractive summarization aims at producing the summary by constructing sentences or phrases using words available in the source which is commonly modeled as an NLG problem. The latest studies have shown that the summary generated by NLG models suffers from factual inconsistency issues, demanding more robust solutions.

Aiming at resolving the factual inconsistency issue, Cao et al. [Cao et al., 2018] extract fact descriptions from source sentences in the form of *(subject, predicate, object)* using the Open Information Extraction (OpenIE) tool and a dependency parser. The fact descriptions are provided as additional input to a neural model composed of two encoders and a dual attention decoder. A similar approach was carried out by [Wang et al., 2022a], where the authors extract knowledge graph triplets related to the input text, map them into low dimensional vector space, and train a graph embedding classifier to determine whether the triplet should be included in the summary or not. The embedding of triplets classified as key information is fed into a decoder along with the output of the input encoder for summary generation. Different from these objectives, [Narayan et al., 2018] attempt to enforce the generation of topic-aware summaries by integrating the topic models with neural approaches. The authors first apply the topic model to the source document and input the topic distribution as an additional input of an attention-based convolutional encoder-decoder model. This enables the model to associate each word in the document with key topics and condition the output words on the topic distribution of the document.

Similar to abstractive summarization, extraction summarization techniques have also adopted hybrid approaches to effectively model cross-sentence relations prior to the selection of summary-worthy sentences from the source. [Wang et al., 2020a] propose a heterogeneous graph-based neural network to model the inter-sentence relationships. The authors construct a heterogeneous graph with word and sentence nodes and model semantic features of the nodes and edges using various techniques including, CNN and BiLSTM-based sentence representation and TF-IDF-based edge weights. Graph attention networks combined with transformers are used to obtain the final representation of nodes. Finally, the authors choose sentence nodes in the heterogeneous graph for summary generation via node classification.

4.2.4 Machine Translation

Machine translation involves the automated conversion of text from one language to another. Initially, rule-based approaches and statistical approaches were prevalent in this field and later neural machine translation (NMT) turned out to be a key milestone in this era. Compared to other NLG tasks, machine translation requires less information from external sources as it is enforced to preserve the content during the conversion from the source language to the target language. However, enhancing the input to NMT with linguistic features such as morphological analysis, part of speech tags, and dependency labels is shown to improve the quality of the task [Sennrich and Haddow, 2016, Chen et al., 2018]. [Bastings et al., 2017] extend this idea to apply a graph convolution network on the dependency trees to obtain a dense vector representation for the sentence structure. Apart from utilizing the linguistic features, Chen et al. [Chen et al., 2018] aid the translation using search engines by extracting similar source sentences and their corresponding translation. Among the retrieved sentence pairs, top K sentences are chosen using edit distance and provided as additional input to an attention-based NMT model, enabling it to carefully attend to relevant sentence pair examples.

4.3 Hybrid Approaches to Natural Language Reasoning

Natural language reasoning (NLR) aims to integrate diverse knowledge sources such as encyclopedic and commonsense knowledge, to draw new logical conclusions about the actual or hypothetical world [Yu et al., 2023]. The knowledge can be integrated from both implicit and explicit sources. Reasoning plays a vital role in various NLU and NLG tasks, where neither memorizing the knowledge in the training data nor understanding the context is sufficient for deriving conclusions and requiring the integration of knowledge. This section presents hybrid approaches used for NLR in tasks specific to HYBRIDS research projects.

4.3.1 Argument Mining

Argument mining involves the identification of structured argument data from unstructured text, including the identification of premise and conclusion [Lawrence and Reed, 2019]. This enables understanding of the individual components of the arguments and their relationships used to convey the overall message. Argument mining is widely used for the development of qualitative assessment tools for online content grasping the attention of both policymakers and social media researchers.

Integration of knowledge graphs, Wikipedia, search engines, and pre-trained language models are often used as a solution for argument mining to infuse external knowledge and commence required for the inference [Fromm et al., 2019, Abels et al., 2021, Saadat-Yazdi et al., 2022, Saadat-Yazdi et al., 2023]. [Fromm et al., 2019] integrates three knowledge sources, Word2Vec, DBpedia knowledge graph, and a pre-trained BERT model for classifying a sentence as an argument or not for the given topic. The authors obtain vector representations of the sentence and the topic using the Word2Vec model and input them into a BiLSTM to encode them. Triplets of the entities present in the argument are extracted from the knowledge graph and converted to embedding vectors using a graph embedding technique TransE [Bordes et al., 2013]. Encoded topic and argument vectors and entity embeddings are used to fine-tune a BERT classifier. Similarly, [Abels et al., 2021] integrate topic modeling, Wikipedia, Knowledge graph, and search engine for argument mining. The authors learn topics using Latent Dirichlet Allocation (LDA) from Wikipedia pages linked using the entities present in the input sentence and obtain the subgraph related to the topic words from the existing knowledge graph, Wikidata. To resolve the incompleteness issue in Wikipedia, the authors constructed another knowledge graph using content that resulted in a Google search for the topic words. Finally, the authors extract evident paths from both knowledge graphs using a breadth-first search, convert each path into a vector representation using a Bi-LSTM, and use it as additional input for argument mining.

4.3.2 Automated Fact-Checking

Automated fact-checking is an essential task for detecting and mitigating the impact of misinformation. This is generally composed of three stages: 1. claim detection to identify sentences with check-worthy or verifiable claims; 2. evidence retrieval to extract supporting statements of the claim; and 3. claim verification to validate whether the retrieved claim is true or not based on the evidence. The evidence retrieval and claim verification tasks are often combined and handled as fact verification [Guo et al., 2022]. It is very evident that leveraging hybrid knowledge sources and techniques can significantly enhance fact-verification tasks for precise inference.

Fact-checking using knowledge sources was often solved by constructing knowledge graphs using the evidence gathered and executing path detection algorithms in knowledge graphs for claim verification [Ciampaglia et al., 2015, Shi and Weninger, 2016, Shiralkar et al., 2017] until the field began integrating of large language models. Addressing this aspect, [Zhou et al., 2019] integrate Wikipedia data, the BERT model, and a graph neural network for claim verification. The authors retrieve related sentences to the claim from Wikipedia using MediaWiki API and choose the top 5 relevant sentences using the hinge loss function. Both the claim and relevant sentences are encoded using the BERT model and input to a graph neural network for veracity detection of the claim. [Zhong et al., 2020] extend this approach by explicitly modeling the relationship between the evidence sentences by constructing an evidence graph. The authors apply the AllenNLP tool for semantic role labeling and model arguments and links between arguments as nodes and edges in the evidence graph. Graph-enhanced contextual representation of the words in the evidence graph is extracted by the pre-trained model XL-net and input to graph neural network for veracity classification of claims. Adopting this methodology, numerous inference techniques using graph neural networks and evidence graphs for fact-checking have been employed in the literature [Liu et al., 2020b, Xu et al., 2022].

Apart from the graph-based techniques,[Si et al., 2021] integrate topic models and neural networks for retrieving topic-constrained evidence information. Given a claim and set of evidence sentences, the authors apply Latent Dirichlet Allocation to extract the topics from the evidence sentences, and the topic distributions learned are used to obtain a topic representation of the evidence via a co-attention mechanism. This enables the authors to incorporate topic consistency between the evidence and topic consistency between the claim and evidence into a dense representation. This topic-aware evidence representation and claim are input to a capsule network for determining the stance of the evidence towards the claim.

4.4 Challenges

While hybrid approaches eliminate weaknesses of symbolic approaches and deep learning approaches, they pose certain challenges that would significantly impact practical usage. Following are some of the key challenges in the implementation of hybrid approaches.

- Generalization of knowledge - Although the existing powerful models are infused with external knowledge sources for accurate inference, deep learning models tend to remember the knowledge provided during the learning and fail to update their internal memory according to the changes in the real world. This requires more generalized solutions without the need to retrain the model with changes in the real world.
- Generalization across tasks - Hybrid models are often trained for a specific task with the integration of symbolic representation required to accomplish inference for the given task, making them incompatible across other NLP tasks.
- Reliability of knowledge sources - Another key concern of hybrid models is the reliability of external sources infused during the training which are often curated using automated tools and search engines. This questions the reliability of the factual inferences obtained by hybrid models [Yin et al., 2022].
- Human-level reasoning - Hybrid approaches are relatively powerful in natural language reasoning. However, human-level reasoning remains an open research problem [Yin et al., 2022], requiring more robust reasoning models simulating human thoughts.

4.5 Conclusion

The hybrid approach is a promising direction of combining rich knowledge in symbolic approaches with machine learning and deep learning to enhance their inference capabilities. Moreover, it facilitates the generation of more credible and factually grounded inferences by incorporating external knowledge and common sense, ultimately improving the reliability and adaptability of hybrid solutions across a broad spectrum of NLP tasks.

Hybrid approaches for NLP are generally decomposed into two types of techniques, integration of symbolic knowledge into the input of statistical or deep learning models and modifying the deep learning models with symbolic structures resulting in architectures such as graph neural networks and hybrid data representation such as graph embedding. The first approach enables a wide range of adoption of symbolic knowledge sources such as external databases, knowledge graphs, and topic models and encourages the model to utilize this knowledge as additional information during inference to make accurate decisions. Similarly, hybrid architectures empower powerful representations of data and their relationships via hybrid structures, improving the scalability and explainability of the task. Possible future directions in hybrid techniques include the introduction of pre-trained hybrid models, few-shot-reasoning using hybrid models for learning with limited training data, dynamic reasoning for inference of new logic over time, answering more complex questions [Zhang et al., 2021], and automatic construction of symbolic knowledge using hybrid techniques [YU2, 2023]. There is no doubt, that a clear understanding of both deep learning approaches and symbolic approaches will lead to innovative deep learning architectures with substantial advancements in the future.

5 Hybrid approaches applied to text classification

Text classification plays an important role in the development of the Hybrids project, since it is related to the detection of toxic language, or more specific hate speech issues such as harassment, xenophobia, or specific type of news such as hyper-partisan political news. Therefore, this section will explore the task of text classification in some more detail.

The problem of classification is defined as follows: “it involves a collection of training data *D* = $\{X_1, ..., X_N\}$ where each data point is assigned a class label from a set of *k* distinct discrete values indexed by *1...k*”. The goal is to use this training data to create a *classification model* that establishes a connection between the features of each data point and one of the available class labels. Subsequently, when faced with a *test instance* with an unknown class label, this trained model is utilized to forecast a suitable class label for that instance [Aggarwal and Zhai, 2012].

The problem of text classification shares a strong connection with the general classification problem. However, in the context of text classification, the approach primarily relies on detecting whether textual features are present or absent within a document (i.e. a unit of text). Text classification is a fundamental task in NLP that plays a pivotal role in a wide range of applications spanning various domains. This task involves categorizing text data into predefined classes or categories based on its content and context.

Some of the key areas where text classification is essential include email filtering [Adnan et al., 2023], news categorization [Daud et al., 2023] and political analysis [Osnabrügge et al., 2021]. But also more sensitive domains like medical diagnosis [Mahdi and Yuhaniz, 2023], toxicity and hate speech [Saleh et al., 2023] or mental health disorders [Parapar et al., 2023].

Text classification is typically achieved through machine learning and NLP techniques, where algorithms are trained on labelled datasets to learn patterns and features in the text. These models can then make predictions or classifications on new, unseen text data. Common approaches include traditional machine learning algorithms like Naive Bayes and SVM, and more advanced techniques like deep learning with neural networks, such as CNNs or RNNs.

5.1 Traditional Text Classification Models

Nowadays, popular text classification approaches are based on methods like **Large Language Models** (LLMs), which excel at capturing complex linguistic patterns and can generalize well to unseen data; **Convolutional Neural Networks** (CNNs) [Zhou et al., 2022], which are effective for tasks where the arrangement of words and phrases in a sentence is crucial for classification; **Recurrent Neural Networks** (RNNs) [Cai et al., 2018], which are well-suited for tasks involving sequences, however, they tend to struggle with long-range dependencies due to vanishing gradient problems; and **Transformer-Based Models** [Pilicita and Barra, 2023], which have a deep architecture that can effectively model relationships between words in a sentence and often outperform other architectures on various text classification tasks. But also widely used are more traditional models like **Word Embeddings** [Costa et al., 2023], which map words to continuous vector representations, allowing models to capture semantic relationships between words.

5.1.1 Decision Trees

Decision trees are hierarchical structures that divide the data space using conditions on attribute values. In text data, these conditions are often based on the presence or absence of specific words in documents. The division continues recursively until leaf nodes have a minimum number of records or meet certain class purity conditions. The majority class label in a leaf node is used for classification.

[Alfina et al., 2017] have addressed the problem of hate speech detection in the Indonesian language using different classification methods, and have got an F-measure of 93.5% when using word n-gram features with Random Forest Decision Tree algorithm.

5.1.2 Rule-based Classifiers

Rule-based classifiers model the data space using a set of rules, where the left-hand side of each rule represents a condition on the features, and the right-hand side is the class label. These rules are generated from the training data and are used to predict class labels for test instances.

In rule-based classifiers, the left-hand side of a rule typically consists of a set of terms that must be present in a document for the rule to be satisfied. The absence of terms is rarely used because it is not informative for sparse text data. Rules can also be expressed as a simple conjunction of conditions on term presence. Decision trees and decision rules encode rules on the feature space, with decision trees being hierarchical and rules allowing for overlaps.

The training phase constructs rules based on support (the number of instances relevant to the rule) and confidence (the strength of the rule). For a given test instance, all relevant rules are determined, and conflicts between rules are resolved using various methods, such as ranking rules by confidence.

[Weegar and Dalianis, 2015] have developed a rule-based system for information extraction in the domain of breast cancer. They compared the rule-based system outcome with experts doing it manually and obtained an F-score of 0.86.

5.1.3 SVM Classifiers

SVMs are initially designed for numerical data and aim to find separators in the feature space that best separate different classes. The concept of maximizing the margin of separation is introduced, where the “margin” is the normal distance of data points from the separating hyperplane. SVMs determine the optimum direction of discrimination in the feature space, making them robust to high-dimensional data.

Text data is considered well-suited for SVM classification due to its sparse, high-dimensional nature. SVMs can construct nonlinear decision surfaces using the kernel trick, although linear SVMs are commonly used for their simplicity and interpretability.

For automatic document classification in Tagalog, Bation et al. [Bation et al., 2017] have used an SVM classifier with a stemmed dataset, yielding an F-score of 0.9199.

5.1.4 Neural Network Classifiers

Inspired by the human brain's interconnected neurons, neural networks consist of layers of nodes, or artificial neurons, that process and transform input data. Through a training process, where the model learns from labelled examples, neural network classifiers can generalize and make predictions on new, unseen data. In the context of text classification, neural network classifiers excel at capturing intricate relationships and representations within textual information, enabling them to categorize and assign labels to diverse pieces of text with a high degree of accuracy.

1. **Convolutional Neural Networks (CNNs)**: CNNs are deep artificial neural networks designed to identify complex patterns within data, including the extraction of features from both image and text data. Historically, CNNs have predominantly found application in computer vision tasks such as image classification, object detection, and image segmentation. Nevertheless, there has been a recent expansion of CNNs into text-related challenges. Prakhya et al. [Prakhya et al., 2017] have investigated the suitability of CNNs for text classification, using datasets with a large number of classes, and concluded that CNNs outperformed state-of-the-art approaches.

2. **Recurrent Neural Networks (RNNs)**: RNNs are a specific type of artificial neural network designed for processing sequential or time series data. These deep learning algorithms are commonly employed for tasks involving order or time, such as language translation, NLP, speech recognition, and image captioning. In contrast to feedforward and CNNs, RNNs rely on training data for learning. What sets them apart from previous architectures is their "memory" capability, as they incorporate information from previous inputs to influence the current input and output. While traditional deep neural networks assume input and output independence, recurrent neural networks' outputs are dependent on preceding elements within the sequence.

   (a) **Long Short-Term Memory Networks (LSTMs)**: LSTMs posses the capacity to acquire and retain patterns across time. LSTMs prove highly valuable in tackling sequence prediction tasks, such as text classification, where preserving the sequential order of words is crucial. Zhou et al. [Zhou et al., 2016] have conducted an experiment on six different classification tasks, proposing an improved model by integrating bidirectional LSTM with two-dimensional max pooling, and improved the performance on 4 out of 6 tasks.

3. **Transformer-Based Models**: The prevalent design for a large language model is the transformer architecture [Vaswani et al., 2017b], comprised of both an encoder and a decoder. In its operation, a transformer model starts by breaking down the input data into tokens and then carries out mathematical operations simultaneously to uncover connections between these tokens. This approach grants the computer the ability to discern patterns akin to those a human would perceive when presented with the same query.

   Transformer models leverage self-attention mechanisms, a feature that accelerates the learning process compared to conventional models such as long short-term memory models. Self-attention is responsible for enabling the transformer model to take into account various segments of the sequence or the complete context of a sentence when making predictions. Raza [Raza, 2021] proposes a model based on transformers for automatic fake news detection in political platforms, detecting fake news with higher accuracy and much earlier, compared to baselines.

4. **Large Language Models (LLMs)**: LLMs are deep learning algorithms capable of executing various natural language processing assignments. These sizable language models utilize transformer architectures and undergo training on extensive datasets, thus gaining their "large" designation. Consequently, they possess the ability to identify, translate, forecast, or create text and other forms of content. Although LLMs' performance is still significantly inferior to fine-tuned models in the task of text classification, some novel approaches like [Sun et al., 2023] have developed models, based on LLMs, that achieve comparable performances to supervised models.

5. **Word Embedding**: Word Embedding are a form of word representation designed to make words with comparable meanings share a similar encoding. They serve as a distributed text representation and are considered a pivotal advancement contributing to the remarkable success of deep learning techniques in addressing complex natural language processing tasks. Works like [Liu et al., 2018] use word embedding to capture semantic and task-specific features of words, applying it to the text classification task, showing that their method significantly outperforms the state-of-the-art methods.

   The models that produce word embeddings have limitations when it comes to representing world knowledge. They rely heavily on the text data they were trained on and may struggle with tasks that require broader context and external knowledge.

5.2 Text Classification Using Hybrid Approaches

Some ways researchers and practitioners are addressing the limitations of pure machine learning approaches is by using **Knowledge Graph Integration**, aiming to incorporate external knowledge from sources like knowledge graphs into text classification models. This helps models access structured information about entities, relationships, and concepts beyond what's available in text data. Efforts are currently in progress to enhance models with commonsense reasoning abilities within the domain of **Commonsense Reasoning** [Sap et al., 2020]. **Pre-training on Diverse Data Sources**, as training models on diverse and multimodal data sources, including text, images, and structured data, can help them gain a broader perspective of the world. And **External Knowledge Injection**, which some techniques involve explicitly injecting external knowledge into text data before feeding it to models. For example, augmenting text with relevant facts or concepts from knowledge bases can improve model performance.

Most research on hybrid approaches focuses on text summarization [Muresan et al., 2001] and text generation or are focused on very specific domains, like the medical domain [Li et al., 2021]. There are some works about knowledge-augmented methods for natural language processing [Zhu et al., 2022], active learning for reducing labelling effort in text classification tasks [Jacobs et al., 2021], and some other state-of-the-art works include hybrid solutions combining deep learning and multilingual lexicon applied to text classification [Pamungkas and Patti, 2019]. But overall, we have found that this is an area with little research for text classification tasks.

The fundamental advantage of hybrid techniques is the incorporation of structured knowledge from the social and human sciences into deep learning algorithms and tools for natural language processing. To address the limitations of current artificial intelligence techniques, hybrid systems help integrate machine and human intellect.

5.2.1 Active Learning

Active Learning (AL) aligns with the concept of Hybrid Intelligence by combining the strengths of both machine and human intelligence. Active Learning allows the model to leverage its learning capabilities while also incorporating the nuanced understanding and expertise of human annotators. It's a collaborative approach to training models that aims to overcome the limitations of relying solely on traditional machine learning or artificial intelligence methods.

Active Learning seeks to minimize the amount of data that requires annotation by human experts [Schröder and Niekler, 2020]. It follows an iterative and cyclic procedure involving a guide, typically the human annotator, and an active learner. In contrast to passive learning, where data is passively provided to the algorithm, in active learning, the active learner selects which samples should be labelled next. However, the actual labelling is carried out by a human expert, often referred to as the "human in the loop." After acquiring new labels, the active learner proceeds to train a fresh model, and this cycle repeats. When we mention the term "active learner", we are referring to the combination of a model, a query strategy, and a stopping criterion.

5.2.2 Applications of Text Classification Using Hybrid Methods

In the dynamic landscape of NLP, the quest for optimal text classification methods has led to the development and exploration of hybrid approaches that seamlessly integrate diverse techniques. In this subsection, we showcase various domains where endeavours in hybrid text classification have been undertaken.

1. **Medical Text Classification**. Medical text classification is crucial for healthcare professionals and researchers to manage and extract valuable insights from the vast amount of medical data generated daily. By accurately categorizing medical texts, such as clinical notes, research papers, and patient records, it streamlines information retrieval, diagnosis, treatment planning, and medical research. This automation saves time, reduces errors, and ultimately improves patient care, enabling healthcare providers to make more informed decisions and advance medical knowledge. Experiments carried out by Li et al. [Li et al., 2021] aim to improve the quality and transparency of medical text classification solutions by proposing a hybrid method for medical text classification that combines a bi-directional Long Short-Term Memory architecture with an attention layer and a regular expression-based classifier.

2. **Hate Speech and Toxicity Detection on Social Media**. Hate speech detection is a vital task in safeguarding online spaces from harmful content. Combining human judgment

and artificial intelligence enables more nuanced and accurate identification of offensive language and harmful rhetoric. This hybrid approach not only enhances content moderation efficiency but also helps mitigate biases, ensuring a safer and more inclusive online environment for all users. Works like [Pamungkas and Patti, 2019] address the challenges of detecting abusive language in social media across different domains and languages by proposing a hybrid approach that combines deep learning and a multilingual lexicon. There are also some proposed solutions to narrow the challenge of detecting toxicity by [J et al., 2023]. J et al. use a symbolic artificial intelligence approach for toxicity identification in combination with a BERT fine-tuning model for toxicity category detection.

3. **Disinformation detection**. Detecting disinformation on social media is of paramount importance in the modern information age. False or misleading information can have profound societal impacts, from influencing public opinion to undermining trust in institutions. Hybrid approaches, combining AI algorithms for identifying suspicious content with human fact-checkers, offer a robust defence against disinformation. They leverage the speed and scalability of AI and the nuanced judgment of humans, curbing the spread of false narratives, and safeguarding the integrity of information dissemination and the democratic process in the digital era. Different works like [Okunoye and Ibor, 2022] and [Davoudi et al., 2022] have already raised concerns about this issue and proposed different hybrid approaches. The first shares a fake news detection technique that uses genetic search for neural architecture selection and deep learning to classify news; the second, proposes propagation-based and graph-based features in combination with a deep model for fake news detection.

In summary, the application of hybrid text classification methods represents a cutting-edge approach to natural language processing and machine learning. This methodology leverages the complementary strengths of AI and human expertise to tackle intricate problems such as hate speech detection, medical text classification, and disinformation identification.

The incorporation of human reviewers adds a vital layer of contextual understanding, addressing nuances and cultural subtleties that pure AI systems may overlook. Meanwhile, AI algorithms provide scalability and the ability to process vast amounts of data at high speeds. This collaborative synergy significantly enhances the precision and adaptability of the classification process.

5.3 Evaluation of Hybrid Text Classification Methods

In this subsection, we turn our focus to the critical aspect of evaluating text classification methods. Assessing the performance and effectiveness is paramount in understanding their practical applicability and fine-tuning their capabilities for specific use cases. Besides the evaluation strategies discussed in Section 2, we can also consider the following when addressing hybrid text classification tasks:

Specificity focuses on the negative class. It measures the proportion of correctly identified true negative cases. **Fall-out** calculates the probability of determining a positive value when there is none. It is the proportion of falsely classified actual negative cases as positive. Miss rate quantifies the proportion of positive values incorrectly classified as negative.

ROC curves, or receiver-operator curves, depicts the relationship between sensitivity and fall-out. They summarize the model's performance by combining confusion matrices at various threshold values. ROC curves help identify an optimal probability threshold, which is used by the model for classification. These thresholds determine the minimum predicted probability for a positive class prediction.

When working with neural networks, **cross-entropy** is employed as an evaluation metric to measure the effectiveness of a classification model's predictions. It quantifies the dissimilarity between the predicted probability distribution (the model's classification probabilities) and the actual distribution of the observed labels (the ground truth). Lower cross-entropy values suggest that the model's predictions align more closely with the true labels, indicating superior classification performance. As a standard evaluation metric for text classification, cross-entropy provides a valuable measure of the model's ability to correctly categorize text data.

Besides performance metrics, when working with hybrid methods, it is important to evaluate the **human agreement**. Measuring the agreement level among human annotators to evaluate their consistency and reliability of them. Metrics for inter-annotator agreement, for example, Cohen's Kappa, can provide insights into human reviewer consensus. In addition, it is interesting to conduct a **bias analysis** to identify and mitigate any potential biases introduced by human reviewers or the AI model.

When working with humans and AI models, we need to bear in mind the cost-effectiveness of the hybrid approach by considering **resource allocation**, assessing the amount of time and resources required for human reviewers and model refinement. Evaluating hybrid text classification methods is an ongoing process that requires a multidimensional approach, considering both technical and ethical aspects. It involves continuous monitoring, feedback incorporation, and fine-tuning to ensure that the hybrid approach aligns with its intended goals and responsibilities. As we delve further into this research domain, there arises a necessity for innovative approaches to seamlessly integrate human judgments with quantitative performance metrics. Exploring new methodologies becomes imperative to enhance our understanding of this field.

5.4 Domain-Specific Advantages of Hybrid Approaches Compared to Non-Hybrid Methods

Hybrid methods offer enhanced accuracy by combining human judgment and AI algorithms. They are adaptable to changing datasets and nuanced contexts, thanks to human annotators' domain expertise. These approaches are effective in reducing bias and ensuring fair content moderation. They excel in handling complex tasks requiring context and nuance, such as disinformation detection or medical diagnosis.

Hybrid approaches harness the scalability of machine learning models for efficiently handling large volumes of data. Machine learning automates routine classification tasks, reducing the need for human intervention in straightforward cases. These models generalize well, enabling them to tackle diverse text classification tasks without the need for manual rule specification.

Human annotators in hybrid approaches bring domain-specific knowledge, which is vital for applications like medical text analysis or legal document classification. They excel in understanding context and cultural nuances, ensuring more accurate classifications. Additionally, human reviewers are effective in handling anomalies or exceptional cases that can be challenging for purely machine learning-based models.

In summary, hybrid approaches offer a balanced fusion of the advantages of both machine learning and knowledge-based methods, resulting in improved accuracy, adaptability, and bias mitigation. This approach leverages machine learning for automation and scalability while tapping into human expertise to navigate complexities and context-specific challenges, making it a powerful paradigm for text classification tasks.

5.5 Challenges in Hybrid Approaches

While text classification models have come a long way and continue to advance, the challenge of endowing them with comprehensive world knowledge remains an active area of research. Bridging this knowledge gap is essential to make these models even more powerful and reliable in real-world applications.

The adoption of hybrid approaches for text classification is not without its challenges. These methods, which integrate human expertise with artificial intelligence, face several complex issues. The cost and scalability of human annotation can be prohibitive, particularly when dealing with large datasets. Subjectivity and inconsistency may emerge due to human involvement, potentially affecting the quality of training data. Additionally, there are concerns surrounding data privacy and security, as sensitive information is handled during annotation.

Model explainability poses another challenge, as hybrid models may produce results that are challenging to interpret. Addressing bias and fairness is a crucial consideration, and striking a balance between human involvement and automation remains a challenge. Efficiently managing the feedback loop between humans and AI models requires well-defined processes and tools. Allocating limited resources, ensuring regulatory compliance, training human annotators, and navigating ethical concerns add further complexity to the implementation of hybrid approaches for text classification. These challenges underscore the need for a holistic, well-documented approach that carefully considers the strengths and limitations of both human and AI components while maintaining a strong ethical focus.

5.6 Conclusion

We delved into the multifaceted landscape of text classification, exploring both machine learning and hybrid approaches that combine human intelligence with artificial intelligence. Text classification, a fundamental task in natural language processing, has witnessed a remarkable evolution driven by the relentless advancement of machine learning techniques. We've seen how traditional supervised models, such as Naïve Bayes and Support Vector Machines, have paved the way for state-of-the-art deep learning models like Transformers, significantly elevating the accuracy and efficiency of text classification.

Hybrid techniques in NLP hold great promise for overcoming the limitations of current AI approaches. By combining the structured knowledge and expertise of human sciences with the power of deep learning and other AI technologies, these hybrid systems have the potential to create more context-aware, reliable, and adaptable NLP tools. This convergence of human and machine intellect represents a significant step forward in the development of AI systems that can better serve a wide range of applications and industries.

The emergence of hybrid approaches, however, has further transformed the field by leveraging the strengths of both AI and human judgment. These methods are not only more versatile in handling complex tasks like hate speech detection, medical text classification, and disinformation identification, but they also serve as a testament to the marriage of technological prowess with human insight. As we've discussed, it's evident that text classification, whether through machine learning or hybrid methods, plays an indispensable role in shaping our digital world. As we move forward, the challenge lies in striking a balance between automation and human involvement, mitigating bias, ensuring data privacy, and addressing ethical concerns, all while harnessing the potential of these methods to foster a safer, more informed, and inclusive digital society.

6 Final Remarks

This report has provided an overview of the historical advances in various approaches to Natural Language Processing leading up to the present day, with a focus on text classification tasks. Section 2 reviewed approaches based on machine learning techniques that have been applied to text classification and generation. These approaches can be divided into two categories: statistical machine learning and deep learning. Starting from classic approaches to encoding and representation of textual input, it traced the developments through deep learning approaches up to the current landscape, marked by the advent of Transformer models and Large Language Models. Section 3 reviewed work in symbolic approaches to NLP, that are driven by human knowledge. The section paid special attention to the most prominent knowledge structures that are used in symbolic approaches, starting from controlled vocabularies, knowledge graphs, linked data, and ontologies. It also detailed two dominant theoretical frameworks, namely, Abstract Meaning Representation and Rhetorical Structure Theory. Section 4 provided an overview of the more recent hybrid approaches to NLP, NLG and NLR as they have been applied in various tasks such as text classification, language modeling, argument mining and automated fact-checking. Generally speaking, hybrid approaches fall into two categories: those that combine symbolic knowledge with statistical representations on the one hand, and those that modify deep-learning based models with structural symbolic information such as graphs and ontologies on the other hand. Finally, section 5 deals with hybrid approaches to text classification, showing how these overcome the challenges faced by traditional approaches to text classification problems.

All the approaches explored in this report face different challenges and limitations. The goal of hybrid approaches to NLP and text classification is to overcome the challenges and limitations of the approaches that preceded them historically. As a final remark, we will briefly summarize the challenges outlined in the previous sections and facing each approaches that precede hybrid methodologies. One of the biggest challenges facing machine learning approaches in the context of text classification has been the issue of capturing contextual relationships in linguistic input. In classic, statistical machine learning approaches, this problem is reflected in the limitations of the statistical representation of the text-based input. Statistical encoding strategies like TF-IDF, BOW, and Word Embeddings only capture a limited amount of linguistic context. While this has improved drastically with the current state of the art in many tasks that use deep learning architectures like Transformer and Attention-based models, these models still face the challenge of model explainability because of their black-box characteristics. Additionally, deep learning models are often challenged by the fact that it is difficult to update their internal memory to match with the real world without retraining them.

In symbolic approaches to NLP, the hallmark challenge is the symbol grounding problem. In the context of NLP, this is the problem of whether representational knowledge (e.g. semantic information) is intrinsic in intelligent systems, and if not, how they can be acquired from data via learning. The various subsets of symbolic approaches to NLP have different limitations. Controlled vocabularies are limited by the size of their lexica and the fact that these resources (like thesauri) only capture limited sets of relations between terms. While Knowledge Graphs overcome this limitation by encoding relations that are wider in scope, neural techniques employed in current research on KGs are vulnerable to noise in addition to being black-boxes, depending on the nature of the system. Although Linked Data approaches facilitate integrating, reusing and diffusing knowledge representations in a system, the main challenge for this technique is the complexity of the task of integrating different knowledge sources, which is arduous. Challenges for Abstract Meaning Representations systems in semantic parsing include relation matching and robustness to low-level variance and noise, such as creative tokenization and misspelling. Rhetorical Structure Theory and systems based on extracting and parsing text at the level of rhetorical units are challenged by the following: limited (scarce) training data, how to establish standardized evaluation metrics, and the complexity of the knowledge structures that can be modeled. Finally, a general challenge to symbolic approaches is their reliance on manually and labor-intensively crafted features, in addition to the generalizability and scalability of symbolic AI systems.

Although Hybrid Approaches to NLP broadly speaking overcome challenges faced by Deep Learning and Symbolic approaches, they are themselves faced with challenges including the generalizability across tasks, since hybrid models tend to be highly specialized and 'vertical'. Hybrid systems are also limited by the reliability of their, often external, knowledge sources used during training. Finally, while hybrid models perform well on natural language reasoning tasks, they yet to achieve human-level reasoning capabilities. In the context text classification specifically, hybrid approaches are challenged by issues of scalability and cost-efficiency of labor-intensive human annotations that are required for the symbolic and knowledge-based components of hybrid systems. Because of the partial reliance on human annotation in this context, hybrid approaches also face the challenge of subjectivity in annotation, depending on the classification task, and privacy concerns. Since hybrid models often include deep, neural components, there are still unresolved issues of model explainability that affect hybrid systems that incorporate these components. While there are many opportunities for future research in the area of hybrid A.I. the general challenge is the question of how to further bridge the gap between human-centered world knowledge representations and structures with deep learning based information.

References

[YU2, 2023] (2023). A survey on neural-symbolic learning systems. *Neural Networks*, 166:105–126.

[Abburu, 2012] Abburu, S. (2012). A survey on ontology reasoners and comparison. *International Journal of Computer Applications*, 57(17).

[Abels et al., 2021] Abels, P., Ahmadi, Z., Burkhardt, S., Schiller, B., Gurevych, I., and Kramer, S. (2021). Focusing knowledge-based graph argument mining via topic modeling. *arXiv preprint arXiv:2102.02086*.

[Adnan et al., 2023] Adnan, M., Imam, M. O., Javed, M. F., and Murtza, I. (2023). Improving spam email classification accuracy using ensemble techniques: a stacking approach. *International Journal of Information Security*.

[Aggarwal and Zhai, 2012] Aggarwal, C. C. and Zhai, C. (2012). *A Survey of Text Classification Algorithms*, pages 163–222. Springer US, Boston, MA.

[Alfina et al., 2017] Alfina, I., Mulia, R., Fanany, M. I., and Ekanata, Y. (2017). Hate speech detection in the indonesian language: A dataset and preliminary study. In *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 233–238.

[Auer et al., 2007] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *international semantic web conference*, pages 722–735. Springer.

[Bahdanau et al., 2016] Bahdanau, D., Cho, K., and Bengio, Y. (2016). Neural machine translation by jointly learning to align and translate.

[Banarescu et al.,] Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Gri, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. Abstract Meaning Representation for Sembanking.

[Barth et al., 2022] Barth, F., Varachkina, H., Dönicke, T., and Gödeke, L. (2022). Levels of non-fictionality in fictional texts. In *Proceedings of the 18th Joint ACL - ISO Workshop on Interoperable Semantic Annotation within LREC2022*, pages 27–32, Marseille, France. European Language Resources Association.

[Bastings et al., 2017] Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., and Sima'an, K. (2017). Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark. Association for Computational Linguistics.

[Bation et al., 2017] Bation, A. D., Vicente, A. J., and Manguilimotan, E. (2017). Automatic categorization of tagalog documents using support vector machines. In *Pacific Asia Conference on Language, Information and Computation*.

[Bender et al., 2021] Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

[Bengio et al., 2003] Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155.

[Bevilacqua et al., 2021] Bevilacqua, M., Blloshmi, R., and Navigli, R. (2021). One SPRING to Rule Them Both: Symmetric AMR Semantic Parsing and Generation without a Complex Pipeline. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12564–12573.

[Bhaskaran and Bhallamudi, 2019] Bhaskaran, J. and Bhallamudi, I. (2019). Good secretaries, bad truck drivers? occupational gender stereotypes in sentiment analysis. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 62–68, Florence, Italy. Association for Computational Linguistics.

[Bhatia et al., 2015] Bhatia, P., Ji, Y., and Eisenstein, J. (2015). Better document-level sentiment analysis from rst discourse parsing. *ArXiv*, abs/1509.01599.

[Blodgett et al., 2020] Blodgett, S. L., Barocas, S., Daumé III, H., and Wallach, H. (2020). Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

[Bordes et al., 2013] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

[Boros et al., 2022] Boros, E., González-Gallardo, C.-E., Giamphy, E., Hamdi, A., Moreno, J. G., and Doucet, A. (2022). Knowledge-based contexts for historical named entity recognition & linking. *Experimental IR Meets Multilinguality, Multimodality, and Interaction*.

[Breiman,] Breiman, L. Random forests. 45(1):5–32.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. *CoRR*, abs/2005.14165.

[Cai and Lam, 2020] Cai, D. and Lam, W. (2020). Graph transformer for graph-to-sequence learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7464–7471.

[Cai et al., 2018] Cai, J., Li, J., Li, W., and Wang, J. (2018). Deeplearning model used in text classification. In *2018 15th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pages 123–126.

[Cao et al., 2018] Cao, Z., Wei, F., Li, W., and Li, S. (2018). Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Chen et al., 2018] Chen, K., Wang, R., Utiyama, M., Sumita, E., and Zhao, T. (2018). Syntax-directed attention for neural machine translation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

[Chen and Guestrin, 2016] Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754.

[Chen et al., 2020] Chen, X., Jia, S., and Xiang, Y. (2020). A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, 141:112948.

[Child et al., 2019] Child, R., Gray, S., Radford, A., and Sutskever, I. (2019). Generating long sequences with sparse transformers. *CoRR*, abs/1904.10509.

[Cho et al., 2014] Cho, K., van Merrienboer, B., Gülçehre, Ç., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

[Christiano et al., 2023] Christiano, P., Leike, J., Brown, T. B., Martic, M., Legg, S., and Amodei, D. (2023). Deep reinforcement learning from human preferences.

[Ciampaglia et al., 2015] Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., and Flammini, A. (2015). Computational fact checking from knowledge networks. *PloS one*, 10(6):e0128193.

[Costa et al., 2023] Costa, L., Oliveira, I., and Fileto, R. (2023). Text classification using embeddings: a survey. *Knowledge and Information Systems*, 65:1–43.

[Damonte and Monti, 2021] Damonte, M. and Monti, E. (2021). One semantic parser to parse them all: Sequence to sequence multi-task learning on semantic parsing datasets. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 173–184, Online. Association for Computational Linguistics.

[Dao et al., 2022] Dao, T., Fu, D. Y., Ermon, S., Rudra, A., and Ré, C. (2022). Flashattention: Fast and memory-efficient exact attention with io-awareness.

[Daud et al., 2023] Daud, S., Ullah, M., Rehman, A., Saba, T., Damaševičius, R., and Sattar, A. (2023). Topic classification of online news articles using optimized machine learning models. *Computers*, 12(1).

[Davoudi et al., 2022] Davoudi, M., Moosavi, M. R., and Sadreddini, M. H. (2022). Dss: A hybrid deep model for fake news detection using propagation tree and stance network. *Expert Systems with Applications*, 198:116635.

[Decker et al., 2000] Decker, S., Melnik, S., Van Harmelen, F., Fensel, D., Klein, M., Broekstra, J., Erdmann, M., and Horrocks, I. (2000). The semantic web: The roles of xml and rdf. *IEEE Internet computing*, 4(5):63–73.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[Dinan et al., 2018] Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., and Weston, J. (2018). Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.

[Ding et al., 2016] Ding, X., Zhang, Y., Liu, T., and Duan, J. (2016). Knowledge-driven event embedding for stock prediction. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers*, pages 2133–2142.

[Fellbaum, 2010] Fellbaum, C. (2010). Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

[Feng and Hirst, 2014] Feng, V. W. and Hirst, G. (2014). A Linear-Time Bottom-Up Discourse Parser with Constraints and Post-Editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 511–521, Baltimore, Maryland. Association for Computational Linguistics.

[Feng et al., 2020] Feng, Y., Chen, X., Lin, B. Y., Wang, P., Yan, J., and Ren, X. (2020). Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1295–1309, Online. Association for Computational Linguistics.

[Flanigan et al., 2014] Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., and Smith, N. A. (2014). A Discriminative Graph-Based Parser for the Abstract Meaning Representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1426–1436, Baltimore, Maryland. Association for Computational Linguistics.

[Fonseca and Egenhofer, 1999] Fonseca, F. T. and Egenhofer, M. J. (1999). Ontology-driven geographic information systems. In *Proceedings of the 7th ACM international symposium on Advances in geographic information systems*, pages 14–19.

[Fromm et al., 2019] Fromm, M., Faerman, E., and Seidl, T. (2019). Tacam: topic and context aware argument mining. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 99–106.

[Gao et al., 2020] Gao, Y., Wu, C.-S., Li, J., Joty, S. R., Hoi, S. C. H., Xiong, C., King, I., and Lyu, M. R. (2020). Discern: Discourse-aware entailment reasoning network for conversational machine reading. *ArXiv*, abs/2010.01838.

[Garcez and Lamb, 2020] Garcez, A. d. and Lamb, L. C. (2020). Neurosymbolic AI: The 3rd Wave. arXiv:2012.05876 [cs].

[Garnelo and Shanahan, 2019] Garnelo, M. and Shanahan, M. (2019). Reconciling deep learning with symbolic artificial intelligence: representing objects and relations. *Current Opinion in Behavioral Sciences*, 29:17–23.

[Gerani et al., 2014] Gerani, S., Mehdad, Y., Carenini, G., Ng, R. T., and Nejat, B. (2014). Abstractive summarization of product reviews using discourse structure. In *Conference on Empirical Methods in Natural Language Processing*.

[Ghazvininejad et al., 2018] Ghazvininejad, M., Brockett, C., Chang, M.-W., Dolan, B., Gao, J., Yih, W.-t., and Galley, M. (2018). A knowledge-grounded neural conversation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Guan et al., 2020] Guan, J., Huang, F., Zhao, Z., Zhu, X., and Huang, M. (2020). A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

[Guha et al., 2016] Guha, R. V., Brickley, D., and Macbeth, S. (2016). Schema. org: evolution of structured data on the web. *Communications of the ACM*, 59(2):44–51.

[Gui et al., 2019] Gui, T., Zou, Y., Zhang, Q., Peng, M., Fu, J., Wei, Z., and Huang, X.-J. (2019). A lexicon-based graph neural network for chinese ner. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 1040–1050.

[Guo et al., 2022] Guo, Z., Schlichtkrull, M., and Vlachos, A. (2022). A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

[Hamilton et al., 2022] Hamilton, K., Nayak, A., Božić, B., and Longo, L. (2022). Is neuro-symbolic ai meeting its promises in natural language processing? a structured review. *Semantic Web*, (Preprint):1–42.

[Handschuh, 2007] Handschuh, S. (2007). Semantic web services: Concepts, technologies, and applications. *Semantic Annotation of Resources in the Semantic Web*, pages 135–155.

[Hardy and Vlachos, 2018] Hardy, H. and Vlachos, A. (2018). Guided Neural Language Generation for Abstractive Summarization using Abstract Meaning Representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 768–773, Brussels, Belgium. Association for Computational Linguistics.

[Harris, 1954] Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

[Hirao et al., 2013] Hirao, T., Yoshida, Y., Nishino, M., Yasuda, N., and Nagata, M. (2013). Single-document summarization as a tree knapsack problem. In *Conference on Empirical Methods in Natural Language Processing*.

[Hitzler, 2021] Hitzler, P. (2021). A review of the semantic web field. *Communications of the ACM*, 64(2):76–83.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

[Hu et al., 2022] Hu, W., He, L., Ma, H., Wang, K., and Xiao, J. (2022). Kgner: Improving chinese named entity recognition by bert infused with the knowledge graph. *Applied Sciences*, 12(15):7702.

[J et al., 2023] J, A., Trueman, T., Murugappan, A., Cambria, E., and Mitra, S. (2023). A hybrid ai framework for multi-label toxicity category detection.

[Jacobs et al., 2021] Jacobs, P. F., Wenniger, G. M. d. B., Wiering, M., and Schomaker, L. (2021). Active learning for reducing labeling effort in text classification tasks.

[Jana and Goyal, 2018] Jana, A. and Goyal, P. (2018). Can network embedding of distributional thesaurus be combined with word vectors for better representation? *arXiv preprint arXiv:1802.06196*.

[Jansen et al., 2014] Jansen, P. A., Surdeanu, M., and Clark, P. (2014). Discourse complements lexical semantics for non-factoid answer reranking. In *Annual Meeting of the Association for Computational Linguistics*.

[Jatnika et al., 2019] Jatnika, D., Bijaksana, M. A., and Suryani, A. A. (2019). Word2vec model analysis for semantic similarities in english words. *Procedia Computer Science*, 157:160–167. The 4th International Conference on Computer Science and Computational Intelligence (ICCSCI 2019) : Enabling Collaboration to Escalate Impact of Research Results for Society.

[Ji et al., 2020] Ji, H., Ke, P., Huang, S., Wei, F., Zhu, X., and Huang, M. (2020). Language generation with multi-hop reasoning on commonsense knowledge graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 725–736, Online. Association for Computational Linguistics.

[Ji and Smith, 2017] Ji, Y. and Smith, N. A. (2017). Neural discourse structure for text categorization. In *Annual Meeting of the Association for Computational Linguistics*.

[Jiang et al., 2023] Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. (2023). Mistral 7b.

[Kapetanakis et al., 2012] Kapetanakis, S., Samakovitis, G., Gunasekera, P., and Petridis, M. (2012). Monitoring financial transaction fraud with the use of case-based reasoning.

[Karimi et al., 2021] Karimi, A., Rossi, L., and Prati, A. (2021). Aeda: an easier data augmentation technique for text classification. *arXiv preprint arXiv:2108.13230*.

[Karpukhin et al., 2020] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. In

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6769–6781, Online. Association for Computational Linguistics.

[Kasper, 1989] Kasper, R. T. (1989). A flexible interface for linking applications to Penman's sentence generator. In *Proceedings of the workshop on Speech and Natural Language - HLT '89*, pages 153–158, Philadelphia, Pennsylvania. Association for Computational Linguistics.

[Kilgarriff, 2003] Kilgarriff, A. (2003). Thesauruses for natural language processing. In *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003*, pages 5–13. IEEE.

[Kingma and Ba, 2017] Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization.

[Knight et al., 2021] Knight, K., Badarau, B., Baranescu, L., Bonial, C., Bardocz, M., Griffitt, K., Hermjakob, U., Marcu, D., Palmer, M., O'Gorman, T., et al. (2021). Abstract meaning representation (amr) annotation release 3.0.

[Kobayashi et al., 2021] Kobayashi, N., Hirao, T., Kamigaito, H., Okumura, M., and Nagata, M. (2021). Improving Neural RST Parsing Model with Silver Agreement Subtrees. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1600–1612, Online. Association for Computational Linguistics.

[Kobayashi et al., 2022] Kobayashi, N., Hirao, T., Kamigaito, H., Okumura, M., and Nagata, M. (2022). A Simple and Strong Baseline for End-to-End Neural RST-style Discourse Parsing. arXiv:2210.08355 [cs] version: 2.

[Koncel-Kedziorski et al., 2019] Koncel-Kedziorski, R., Bekal, D., Luan, Y., Lapata, M., and Hajishirzi, H. (2019). Text Generation from Knowledge Graphs with Graph Transformers. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2284–2293, Minneapolis, Minnesota. Association for Computational Linguistics.

[Krishnan and Amarthaluri, 2019] Krishnan, A. and Amarthaluri, A. (2019). Large scale product categorization using structured and unstructured attributes. *ArXiv*, abs/1903.04254.

[Kumar and Solanki, 2023] Kumar, S. and Solanki, A. (2023). An abstractive text summarization technique using transformer model with self-attention mechanism. *Neural Computing and Applications*, 35(25):18603–18622.

[Langkilde and Knight, 1998] Langkilde, I. and Knight, K. (1998). Generation that Exploits Corpus-Based Statistical Knowledge. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

[Lawrence and Reed, 2019] Lawrence, J. and Reed, C. (2019). Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

[LeCun et al., 1989] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2.

[Li et al., 2016] Li, Q., Li, T., and Chang, B. (2016). Discourse parsing with attention-based hierarchical neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 362–371.

[Li et al., 2021] Li, X., Cui, M., Li, J., Bai, R., Lu, Z., and Aickelin, U. (2021). A hybrid medical text classification framework: Integrating attentive rule construction and neural network. *Neurocomputing*, 443:345–355.

[Lin et al., 2019] Lin, X., Joty, S. R., Jwalapuram, P., and Bari, M. S. (2019). A unified linear-time framework for sentence-level discourse parsing. In *Annual Meeting of the Association for Computational Linguistics*.

[Liu et al., 2016] Liu, P., Qiu, X., and Huang, X. (2016). Recurrent neural network for text classification with multi-task learning. *CoRR*, abs/1605.05101.

[Liu et al., 2018] Liu, Q., Huang, H., Gao, Y., Wei, X., Tian, Y., and Liu, L. (2018). Task-oriented word embedding for text classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2023–2032, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

[Liu et al., 2020a] Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., and Wang, P. (2020a). K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2901–2908.

[Liu et al., 2022] Liu, Y., Li, P., and Hu, X. (2022). Combining context-relevant features with multi-stage attention network for short text classification. *Computer Speech & Language*, 71:101268.

[Liu et al., 2019] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

[Liu et al., 2021a] Liu, Y., Wan, Y., He, L., Peng, H., and Philip, S. Y. (2021a). Kg-bart: Knowledge graph-augmented bart for generative commonsense reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6418–6425.

[Liu et al., 2021b] Liu, Z., Shi, K., and Chen, N. F. (2021b). Dmrst: A joint framework for document-level multilingual rst discourse segmentation and parsing. *arXiv preprint arXiv:2110.04518*.

[Liu et al., 2020b] Liu, Z., Xiong, C., Sun, M., and Liu, Z. (2020b). Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7342–7351, Online. Association for Computational Linguistics.

[Loshchilov and Hutter, 2019] Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization.

[Lv et al., 2020] Lv, S., Guo, D., Xu, J., Tang, D., Duan, N., Gong, M., Shou, L., Jiang, D., Cao, G., and Hu, S. (2020). Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8449–8456.

[Mahdi and Yuhaniz, 2023] Mahdi, A. Y. and Yuhaniz, S. S. (2023). Optimal feature selection using novel flamingo search algorithm for classification of covid-19 patients from clinical text. *Math. Biosci. Eng*, 20:5268–5297.

[Mann and Thompson, 1988] Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text & Talk*, 8:243 – 281.

[Marcu, 1998] Marcu, D. (1998). Improving summarization through rhetorical parsing tuning. In *VLC@COLING/ACL*.

[McCarthy, 1960] McCarthy, J. (1960). Programs with common sense.

[Mikolov et al., 2013a] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

[Mitra et al., 2019] Mitra, A., Banerjee, P., Pal, K. K., Mishra, S., and Baral, C. (2019). How additional knowledge can improve natural language commonsense question answering? *arXiv preprint arXiv:1909.08855*.

[Morey et al., 2017] Morey, M., Muller, P., and Asher, N. (2017). How much progress have we made on RST discourse parsing? A replication study of recent results on the RST-DT. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1319–1324, Copenhagen, Denmark. Association for Computational Linguistics.

[Mozer, 1995] Mozer, M. C. (1995). *A Focused Backpropagation Algorithm for Temporal Pattern Recognition*, page 137–169. L. Erlbaum Associates Inc., USA.

[Muresan et al., 2001] Muresan, S., Tzoukermann, E., and Klavans, J. L. (2001). Combining linguistic and machine learning techniques for email summarization. In *Proceedings of the ACL 2001 Workshop on Computational Natural Language Learning (ConLL)*.

[Narayan et al., 2018] Narayan, S., Cohen, S. B., and Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

[Naseem et al., 2019] Naseem, T., Shah, A., Wan, H., Florian, R., Roukos, S., and Ballesteros, M. (2019). Rewarding Smatch: Transition-Based AMR Parsing with Reinforcement Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4586–4592, Florence, Italy. Association for Computational Linguistics.

[Neumann, 2021] Neumann, A. (2021). Using and comparing Rhetorical Structure Theory parsers with rst-workbench. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Online. Association for Computational Linguistics.

[Nguyen et al., 2021] Nguyen, T.-T., Nguyen, X.-P., Joty, S., and Li, X. (2021). RST Parsing from Scratch. arXiv:2105.10861 [cs] version: 1.

[Noy et al., 2001] Noy, N. F., McGuinness, D. L., et al. (2001). Ontology development 101: A guide to creating your first ontology.

[Okunoye and Ibor, 2022] Okunoye, O. B. and Ibor, A. E. (2022). Hybrid fake news detection technique with genetic search and deep learning. *Computers and Electrical Engineering*, 103:108344.

[Osnabrügge et al., 2021] Osnabrügge, M., Ash, E., and Morelli, M. (2021). Cross-domain topic classification for political texts. *Political Analysis*, 31(1):59–80.

[Ouyang et al., 2022] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., and Lowe, R. (2022). Training language models to follow instructions with human feedback.

[Pamungkas and Patti, 2019] Pamungkas, E. W. and Patti, V. (2019). Cross-domain and cross-lingual abusive language detection: A hybrid approach with deep learning and a multilingual lexicon. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 363–370, Florence, Italy. Association for Computational Linguistics.

[Pan et al., 2023] Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., and Wu, X. (2023). Unifying large language models and knowledge graphs: A roadmap. *arXiv preprint arXiv:2306.08302*.

[Parapar et al., 2023] Parapar, J., Martín-Rodilla, P., Losada, D. E., and Crestani, F. (2023). Overview of erisk 2023: Early risk prediction on the internet. In Arampatzis, A., Kanoulas, E., Tsikrika, T., Vrochidis, S., Giachanou, A., Li, D., Aliannejadi, M., Vlachos, M., Faggioli, G., and Ferro, N., editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 294–315, Cham. Springer Nature Switzerland.

[Pedraza-Jimenez and Codina, 2009] Pedraza-Jimenez, R. and Codina, L. (2009). Metadatos en la web semántica: lenguajes de marcado para la organización de sistemas de información. *Trea. Web semántica y Sistemas de Información Documental*.

[Pennington et al., 2014] Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

[Peters et al., 2019] Peters, M. E., Neumann, M., Logan, R., Schwartz, R., Joshi, V., Singh, S., and Smith, N. A. (2019). Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.

[Petroni et al., 2019] Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., and Miller, A. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

[Pilicita and Barra, 2023] Pilicita, A. and Barra, E. (2023). Using of transformers models for text classification to mobile educational applications. *IEEE Latin America Transactions*, 21(6):730–736.

[Pittaras et al., 2023] Pittaras, N., Giannakopoulos, G., Stamatopoulos, P., and Karkaletsis, V. (2023). Content-based and knowledge-enriched representations for classification across modalities: a survey. *ACM Computing Surveys*.

[Prakhya et al., 2017] Prakhya, S., Venkataram, V., and Kalita, J. (2017). Open set text classification using CNNs. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 466–475, Kolkata, India. NLP Association of India.

[Radford et al., 2019] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners.

[Raimond and Sandler, 2012] Raimond, Y. and Sandler, M. (2012). Evaluation of the music ontology framework. In *Extended Semantic Web Conference*, pages 255–269. Springer.

[Raza, 2021] Raza, S. (2021). Automatic fake news detection in political platforms - a transformer-based approach. In *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, pages 68–78, Online. Association for Computational Linguistics.

[Riedl and Biemann, 2013] Riedl, M. and Biemann, C. (2013). Scaling to large^3 data: An efficient and effective method to compute distributional thesauri. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 884–890.

[Rumelhart et al., 1986a] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986a). *Learning Internal Representations by Error Propagation*, page 318–362. MIT Press, Cambridge, MA, USA.

[Rumelhart et al., 1986b] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986b). Learning representations by back-propagating errors. *Nature*, 323:533–536.

[Saadat-Yazdi et al., 2022] Saadat-Yazdi, A., Li, X., Chausson, S., Belle, V., Ross, B., Pan, J. Z., and Kökciyan, N. (2022). KEViN: A knowledge enhanced validity and novelty classifier for arguments. In *Proceedings of the 9th Workshop on Argument Mining*, pages 104–110, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

[Saadat-Yazdi et al., 2023] Saadat-Yazdi, A., Pan, J. Z., and Kokciyan, N. (2023). Uncovering implicit inferences for improved relational argument mining. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2484–2495, Dubrovnik, Croatia. Association for Computational Linguistics.

[Saleh et al., 2023] Saleh, H., Alhothali, A., and Moria, K. (2023). Detection of hate speech using BERT and hate speech word embedding with deep model. *Applied Artificial Intelligence*, 37(1).

[Sap et al., 2019] Sap, M., Le Bras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Roof, B., Smith, N. A., and Choi, Y. (2019). Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3027–3035.

[Sap et al., 2020] Sap, M., Shwartz, V., Bosselut, A., Choi, Y., and Roth, D. (2020). Commonsense reasoning for natural language processing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 27–33, Online. Association for Computational Linguistics.

[Sarker et al., 2021] Sarker, M. K., Zhou, L., Eberhart, A., and Hitzler, P. (2021). Neuro-symbolic artificial intelligence. *AI Communications*, 34(3):197–209.

[Schmidhuber, 2014] Schmidhuber, J. (2014). Deep learning in neural networks: An overview. *CoRR*, abs/1404.7828.

[Schröder and Niekler, 2020] Schröder, C. and Niekler, A. (2020). A survey of active learning for text classification using deep neural networks.

[Schütze et al., 1995] Schütze, H., Hull, D. A., and Pedersen, J. O. (1995). A comparison of classifiers and document representations for the routing problem. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, page 229–237, New York, NY, USA. Association for Computing Machinery.

[Sennrich and Haddow, 2016] Sennrich, R. and Haddow, B. (2016). Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

[Shi and Weninger, 2016] Shi, B. and Weninger, T. (2016). Fact checking in heterogeneous information networks. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 101–102.

[Shi and Weninger, 2018] Shi, B. and Weninger, T. (2018). Open-world knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

[Shiralkar et al., 2017] Shiralkar, P., Flammini, A., Menczer, F., and Ciampaglia, G. L. (2017). Finding streams in knowledge graphs to support fact checking. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 859–864. IEEE.

[Shou et al., 2022] Shou, Z., Jiang, Y., and Lin, F. (2022). AMR-DA: Data Augmentation by Abstract Meaning Representation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3082–3098, Dublin, Ireland. Association for Computational Linguistics.

[Si et al., 2021] Si, J., Zhou, D., Li, T., Shi, X., and He, Y. (2021). Topic-aware evidence reasoning and stance-aware aggregation for fact verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1612–1622, Online. Association for Computational Linguistics.

[Škrlj et al., 2021] Škrlj, B., Martinc, M., Kralj, J., Lavrač, N., and Pollak, S. (2021). tax2vec: Constructing interpretable features from taxonomies for short text classification. *Computer Speech & Language*, 65:101104.

[Smith, 2021] Smith, C. (2021). Controlled vocabularies: Past, present and future of subject access. *Cataloging & Classification Quarterly*, 59:186 – 202.

[Speer et al., 2017] Speer, R., Chin, J., and Havasi, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

[Steiner et al., 2001] Steiner, K., Essmayr, W., and Wagner, R. (2001). Topic maps-an enabling technology for knowledge management. In *12th International Workshop on Database and Expert Systems Applications*, pages 472–476. IEEE.

[Stevens et al., 2000] Stevens, R., Goble, C. A., and Bechhofer, S. (2000). Ontology-based knowledge representation for bioinformatics. *Briefings in bioinformatics*, 1(4):398–414.

[Stiennon et al., 2020] Stiennon, N., Ouyang, L., Wu, J., Ziegler, D. M., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. F. (2020). Learning to summarize from human feedback. *CoRR*, abs/2009.01325.

[Sun et al., 2023] Sun, X., Li, X., Li, J., Wu, F., Guo, S., Zhang, T., and Wang, G. (2023). Text classification via large language models.

[Sutskever et al., 2011] Sutskever, I., Martens, J., and Hinton, G. (2011). Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 1017–1024, Madison, WI, USA. Omnipress.

[Szymański and Ciota, 2002] Szymański, G. and Ciota, Z. (2002). Hidden markov models suitable for text generation.

[Tedeschi et al., 2021] Tedeschi, S., Maiorca, V., Campolungo, N., Cecconi, F., and Navigli, R. (2021). Wikineural: Combined neural and knowledge-based silver data creation for multilingual ner. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2521–2533.

[Thoppilan et al., 2022] Thoppilan, R., Freitas, D. D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H., Jin, A., Bos, T., Baker, L., Du, Y., Li, Y., Lee, H., Zheng, H. S., Ghafouri, A., Menegali, M., Huang, Y., Krikun, M., Lepikhin, D., Qin, J., Chen, D., Xu, Y., Chen, Z., Roberts, A., Bosma, M., Zhou, Y., Chang, C., Krivokon, I., Rusch, W., Pickett, M., Meier-Hellstern, K. S., Morris, M. R., Doshi, T., Santos, R. D., Duke, T., Soraker, J., Zevenbergen, B., Prabhakaran, V., Diaz, M., Hutchinson, B., Olson, K., Molina, A., Hoffman-John, E., Lee, J., Aroyo, L., Rajakumar, R., Butryna, A., Lamm, M., Kuzmina, V., Fenton, J., Cohen, A., Bernstein, R., Kurzweil, R., y Arcas, B. A., Cui, C., Croak, M., Chi, E. H., and Le, Q. (2022). Lamda: Language models for dialog applications. *CoRR*, abs/2201.08239.

[Vallez et al., 2015] Vallez, M., Pedraza-Jimenez, R., Codina, L., Blanco, S., and Rovira, C. (2015). Updating controlled vocabularies by analysing query logs. *Online Information Review*, 39:870–884.

[Vaswani et al., 2017a] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017a). Attention is all you need. *Advances in neural information processing systems*, 30.

[Vaswani et al., 2017b] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017b). Attention is all you need.

[Venkatesh et al., 2022] Venkatesh, G., Jana, A., Remus, S., Sevgili, Ö., Srinivasaraghavan, G., and Biemann, C. (2022). Using distributional thesaurus to enhance transformer-based contextualized representations for low resource languages. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pages 845–852.

[Wang et al., 2020a] Wang, D., Liu, P., Zheng, Y., Qiu, X., and Huang, X. (2020a). Heterogeneous graph neural networks for extractive document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6209–6219, Online. Association for Computational Linguistics.

[Wang et al., 2022a] Wang, G.-H., Li, W., Lai, E. M.-K., and Jiang, J. (2022a). Katsum: Knowledge-aware abstractive text summarization. *ArXiv*, abs/2212.03371.

[Wang et al., 2023] Wang, J., Wang, W., Meng, F., Zhou, L., and Guo, S. (2023). A review of knowledge reasoning based on neural network. In *2023 IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT)*, pages 580–584. IEEE.

[Wang et al., 2021] Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., and Tang, J. (2021). Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

[Wang et al., 2022b] Wang, X., Shen, Y., Cai, J., Wang, T., Wang, X., Xie, P., Huang, F., Lu, W., Zhuang, Y., Tu, K., Lu, W., and Jiang, Y. (2022b). DAMO-NLP at SemEval-2022 task 11: A knowledge-based system for multilingual named entity recognition. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1457–1468, Seattle, United States. Association for Computational Linguistics.

[Wang et al., 2017] Wang, Y., Li, S., and Wang, H. (2017). A Two-Stage Parsing Method for Text-Level Discourse Analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 184–188, Vancouver, Canada. Association for Computational Linguistics.

[Wang et al., 2020b] Wang, Y., Sun, Y., Ma, Z., Gao, L., and Xu, Y. (2020b). An ernie-based joint model for chinese named entity recognition. *Applied Sciences*, 10(16):5711.

[Wang et al., 2010] Wang, Z., Huang, J., Li, H., Liu, B., Shao, B., Wang, H., Wang, J., Wang, Y., Wu, W., Xiao, J., et al. (2010). Probase: a universal knowledge base for semantic search. *Microsoft Research Asia*.

[Weegar and Dalianis, 2015] Weegar, R. and Dalianis, H. (2015). Creating a rule based system for text mining of norwegian breast cancer pathology reports. In *Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis*. Association for Computational Linguistics.

[Wei and Zou, 2019] Wei, J. and Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

[Weizenbaum, 1966a] Weizenbaum, J. (1966a). Eliza—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45.

[Weizenbaum, 1966b] Weizenbaum, J. (1966b). ELIZA—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

[Winograd, 1971] Winograd, T. (1971). Procedures as a representation for data in a computer program for understanding natural language.

[Workshop et al., 2023] Workshop, B., :, Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., Biderman, S., Webson, A., Ammanamanchi, P. S., Wang, T., Sagot, B., Muennighoff, N., del Moral, A. V., Ruwase, O., Bawden, R., Bekman, S., McMillan-Major, A., Beltagy, I., Nguyen, H., Saulnier, L., Tan, S., Suarez, P. O., Sanh, V., Laurençon, H., Jernite, Y., Launay, J., Mitchell, M., Raffel, C., Gokaslan, A., Simhi, A., Soroa, A., Aji, A. F., Alfassy, A., Rogers, A., Nitzav, A. K., Xu, C., Mou, C., Emezue, C., Klamm, C., Leong, C., van Strien, D., Adelani, D. I., Radev, D., Ponferrada, E. G., Levkovizh, E., Kim, E., Natan, E. B., Toni, F. D., Dupont, G., Kruszewski, G., Pistilli, G., Elsahar, H., Benyamina, H., Tran, H., Yu, I., Abdulmumin, I., Johnson, I., Gonzalez-Dios, I., de la Rosa, J., Chim, J., Dodge, J., Zhu, J., Chang, J., Frohberg, J., Tobing, J., Bhattacharjee, J., Almubarak, K., Chen, K., Lo, K., Werra, L. V., Weber, L., Phan, L., allal, L. B., Tanguy, L., Dey, M., Muñoz, M. R., Masoud, M., Grandury, M., Šaško, M., Huang, M., Coavoux, M., Singh, M., Jiang, M. T.-J., Vu, M. C., Jauhar, M. A., Ghaleb, M., Subramani, N., Kassner, N., Khamis, N., Nguyen, O., Espejel, O., de Gibert, O., Villegas, P., Henderson, P., Colombo, P., Amuok, P., Lhoest, Q., Harliman, R., Bommasani, R., López, R. L., Ribeiro, R., Osei, S., Pyysalo, S., Nagel, S., Bose, S., Muhammad, S. H., Sharma, S., Longpre, S., Nikpoor, S., Silberberg, S., Pai, S., Zink, S., Torrent, T. T., Schick, T., Thrush, T., Danchev, V., Nikoulina, V., Laippala, V., Lepercq, V., Prabhu, V., Alyafeai, Z., Talat, Z., Raja, A., Heinzerling, B., Si, C., Taşar, D. E., Salesky, E., Mielke, S. J., Lee, W. Y., Sharma, A., Santilli, A., Chaffin, A., Stiegler, A., Datta, D., Szczechla, E., Chhablani, G., Wang, H., Pandey, H., Strobelt, H., Fries, J. A., Rozen, J., Gao, L., Sutawika, L., Bari, M. S., Al-shaibani, M. S., Manica, M., Nayak, N., Teehan, R., Albanie, S., Shen, S., Ben-David, S., Bach, S. H., Kim, T., Bers, T., Fevry, T., Neeraj, T., Thakker, U., Raunak, V., Tang, X., Yong, Z.-X., Sun, Z., Brody, S., Uri, Y., Tojarieh, H., Roberts, A., Chung, H. W., Tae, J., Phang, J., Press, O., Li, C., Narayanan, D., Bourfoune, H., Casper, J., Rasley, J., Ryabinin, M., Mishra, M., Zhang, M., Shoeybi, M., Peyrounette, M., Patry, N., Tazi, N., Sanseviero, O., von Platen, P., Cornette, P., Lavallée, P. F., Lacroix, R., Rajbhandari, S., Gandhi, S., Smith, S., Requena, S., Patil, S., Dettmers, T., Baruwa, A., Singh, A., Cheveleva, A., Ligozat, A.-L., Subramonian, A., Névéol, A., Lovering, C., Garrette, D., Tunuguntla, D., Reiter, E., Taktasheva, E., Voloshina, E., Bogdanov, E., Winata, G. I., Schoelkopf, H., Kalo, J.-C., Novikova, J., Forde, J. Z., Clive, J., Kasai, J., Kawamura, K., Hazan, L., Carpuat, M., Clinciu, M., Kim, N., Cheng, N., Serikov, O., Antverg, O., van der Wal, O., Zhang, R., Zhang, R., Gehrmann, S., Mirkin, S., Pais, S., Shavrina, T., Scialom, T., Yun, T., Limisiewicz, T., Rieser, V., Protasov, V., Mikhailov, V., Pruksachatkun, Y., Belinkov, Y., Bamberger, Z., Kasner, Z., Rueda, A., Pestana, A., Feizpour, A., Khan, A., Faranak, A., Santos, A., Hevia, A., Unldreaj, A., Aghagol, A., Abdollahi, A., Tammour, A., HajiHosseini, A., Behroozi, B., Ajibade, B., Saxena, B., Ferrandis, C. M., McDuff, D., Contractor, D., Lansky, D., David, D., Kiela, D., Nguyen, D. A., Tan, E., Baylor, E., Ozoani, E., Mirza, F., Ononiwu, F., Rezanejad, H., Jones, H., Bhattacharya, I., Solaiman, I., Sedenko, I., Nejadgholi, I., Passmore, J., Seltzer, J., Sanz, J. B., Dutra, L., Samagaio, M., Elbadri, M., Mieskes, M., Gerchick, M., Akinlolu, M., McKenna, M., Qiu, M., Ghauri, M., Burynok, M., Abrar, N., Rajani, N., Elkott, N., Fahmy, N., Samuel, O., An, R., Kromann, R., Hao, R., Alizadeh, S., Shubber, S., Wang, S., Roy, S., Viguier, S., Le, T., Oyebade, T., Le, T., Yang, Y., Nguyen, Z., Kashyap, A. R., Palasciano, A., Callahan, A., Shukla, A., Miranda-Escalada, A., Singh, A., Beilharz, B., Wang, B., Brito, C., Zhou, C., Jain, C., Xu, C., Fourrier, C., Periñán, D. L., Molano, D., Yu, D., Manjavacas, E., Barth, F., Fuhrimann, F., Altay, G., Bayrak, G., Burns, G., Vrabec, H. U., Bello, I., Dash, I., Kang, J., Giorgi, J., Golde, J., Posada, J. D., Sivaraman, K. R., Bulchandani, L., Liu, L., Shinzato, L., de Bykhovetz, M. H., Takeuchi, M., Pàmies, M., Castillo, M. A., Nezhurina, M., Sänger, M., Samwald, M., Cullan, M., Weinberg, M., Wolf, M. D., Mihaljcic, M., Liu, M., Freidank, M., Kang, M., Seelam, N., Dahlberg, N., Broad, N. M., Muellner, N., Fung, P., Haller, P., Chandrasekhar, R., Eisenberg, R., Martin, R., Canalli, R., Su, R., Su, R., Cahyawijaya, S., Garda, S., Deshmukh, S. S., Mishra, S., Kiblawi, S., Ott, S., Sangaroonsiri, S., Kumar, S., Schweter, S., Bharati, S., Laud, T., Gigant, T., Kainuma, T., Kusa, W., Labrak, Y., Bajaj, Y. S., Venkatraman, Y., Xu, Y., Xu, Y., Xu, Y., Tan, Z., Xie, Z., Ye, Z., Bras, M., Belkada, Y., and Wolf, T. (2023). Bloom: A 176b-parameter open-access multilingual language model.

[Xiong et al., 2019] Xiong, W., Du, J., Wang, W. Y., and Stoyanov, V. (2019). Pretrained encyclopedia: Weakly supervised knowledge-pretrained language model. *arXiv preprint arXiv:1912.09637*.

[Xu et al., 2022] Xu, W., Wu, J., Liu, Q., Wu, S., and Wang, L. (2022). Evidence-aware fake news detection with graph neural networks. In *Proceedings of the ACM Web Conference 2022*, pages 2501–2510.

[Yao et al., 2019] Yao, L., Mao, C., and Luo, Y. (2019). Graph convolutional networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7370–7377.

[Yasunaga et al., 2021] Yasunaga, M., Ren, H., Bosselut, A., Liang, P., and Leskovec, J. (2021). QA-GNN: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 535–546, Online. Association for Computational Linguistics.

[Yin et al., 2022] Yin, D., Dong, L., Cheng, H., Liu, X., Chang, K.-W., Wei, F., and Gao, J. (2022). A survey of knowledge-intensive nlp with pre-trained language models. *arXiv preprint arXiv:2202.08772*.

[Yu et al., 2022a] Yu, D., Zhu, C., Yang, Y., and Zeng, M. (2022a). Jaket: Joint pre-training of knowledge graph and language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11630–11638.

[Yu et al., 2023] Yu, F., Zhang, H., and Wang, B. (2023). Nature language reasoning, a survey. *arXiv preprint arXiv:2303.14725*.

[Yu et al., 2022b] Yu, W., Zhu, C., Li, Z., Hu, Z., Wang, Q., Ji, H., and Jiang, M. (2022b). A survey of knowledge-enhanced text generation. *ACM Computing Surveys*, 54(11s):1–38.

[Yuan et al., 2018] Yuan, K., Deng, Y., Chen, D., Zhang, B., Lei, K., and Shen, Y. (2018). Construction techniques and research development of medical knowledge graph. *Application Research of Computers*, 35(7):1929–1936.

[Zhang et al., 2020] Zhang, H., Liu, Z., Xiong, C., and Liu, Z. (2020). Grounded conversation generation as guided traverses in commonsense knowledge graphs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2031–2043, Online. Association for Computational Linguistics.

[Zhang et al., 2021] Zhang, J., Chen, B., Zhang, L., Ke, X., and Ding, H. (2021). Neural, symbolic and neural-symbolic reasoning on knowledge graphs. *AI Open*, 2:14–35.

[Zhang et al., 2022a] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, P. S., Sridhar, A., Wang, T., and Zettlemoyer, L. (2022a). Opt: Open pre-trained transformer language models.

[Zhang et al., 2022b] Zhang, X., Bosselut, A., Yasunaga, M., Ren, H., Liang, P., Manning, C. D., and Leskovec, J. (2022b). Greaselm: Graph reasoning enhanced language models for question answering. *arXiv preprint arXiv:2201.08860*.

[Zhang et al., 2019] Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. (2019). ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

[Zhong et al., 2020] Zhong, W., Xu, J., Tang, D., Xu, Z., Duan, N., Zhou, M., Wang, J., and Yin, J. (2020). Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6170–6180, Online. Association for Computational Linguistics.

[Zhou et al., 2018] Zhou, H., Young, T., Huang, M., Zhao, H., Xu, J., and Zhu, X. (2018). Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*, pages 4623–4629.

[Zhou et al., 2019] Zhou, J., Han, X., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2019). GEAR: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 892–901, Florence, Italy. Association for Computational Linguistics.

[Zhou et al., 2016] Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., and Xu, B. (2016). Text classification improved by integrating bidirectional lstm with two-dimensional max pooling.

[Zhou et al., 2020] Zhou, W., Lee, D.-H., Selvam, R. K., Lee, S., Lin, B. Y., and Ren, X. (2020). Pre-training text-to-text transformers for concept-centric common sense. *arXiv preprint arXiv:2011.07956*.

[Zhou et al., 2022] Zhou, Y., Li, J., Chi, J., Tang, W., and Zheng, Y. (2022). Set-CNN: A text convolutional neural network based on semantic extension for short text classification. *Knowledge-Based Systems*, 257:109948.

[Zhu et al., 2022] Zhu, C., Xu, Y., Ren, X., Lin, B., Jiang, M., and Yu, W. (2022). Knowledge-augmented methods for natural language processing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 12–20.

[Zhu et al., 2023] Zhu, C., Xu, Y., Ren, X., Lin, B. Y., Jiang, M., and Yu, W. (2023). Knowledge-augmented methods for natural language processing. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 1228–1231.

Author's contributions

Michele Joshua Maggini was responsible for the introduction of section 3 and sections 3.1, 3.5 and 3.6.
Rafael Martins Frade was responsible for sections 3.2, 3.3 and 3.4.
Rrubaa Panchendrarajan was responsible for section 4.
Paloma Piot was responsible for the section 5.
Rabiraj Bandyopadhyay was responsible for section 2.
Søren Fomsgaard was responsible for sections 1 and 6.